N° d'ordre : 155 /2018

Université Sultan Moulay Slimane
Faculté des Sciences et Techniques
Département d'Informatique
Béni Mellal

Centre d'Études Doctorales « Sciences et Techniques »
Formation Doctorale « Mathématique et Physique Appliquées »

THESE

Présentée par

Said NOURI

Pour l'obtention du grade

Doctorat

Option: Informatique

Reconnaissance automatique de l'écriture imprimée Arabe

Soutenu publiquement le 24 Février 2018 devant les membres du jury:

Pr. Mohamed OUKESSOU	Professeur à la Faculté des Sciences et Techniques, Béni Mellal	Président
Pr. Lalla Saadia CHADLI	Professeur à la Faculté des Sciences et Techniques, Béni Mellal	Rapporteur
Pr. Brahim MINAOUI	Professeur à la Faculté des Sciences et Techniques, Béni Mellal	Rapporteur
Pr. Rachid EL AYACHI	Professeur à la Faculté des Sciences et Techniques, Béni Mellal	Examinateur
Pr Ali RACHIDI	Professeur à l'École Nationale de Commerce et de Gestion, Agadir.	Examinateur
Pr. Khalid NAFIL	Professeur à l'École Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Rabat.	Examinateur
Pr. Mohamed FAKIR	Professeur à la Faculté des Sciences et Techniques, Béni Mellal.	Encadrant

Remerciements

Je tiens à remercier vivement mon directeur de thèse monsieur Mohammed Fakir, pour les précieux conseils, directives, patience, disponibilité et sa contribution générale à l'élaboration de ce travail de thèse.

Je remercie vivement tous les membres de mon jury de thèse qui ont pris de leurs temps pour lire et juger ce travail ainsi que pour leur efforts jusqu'au jour de la soutenance.

Je remercie les rapporteurs de ma thèse pour le temps consacré à la lecture de ce travail ainsi que pour leurs commentaires et remarques appropriées.

Je remercie mes parents pour leur soutien qui m'a été bien utile durant ma thèse.

Je remercie également tous les thésards et les autres membres du laboratoire de « traitement d'information et aide à la décision » de la faculté des sciences et techniques de Beni Mellal, notamment ceux avec qui j'ai eu l'occasion de travailler et les autres simplement pour les bons moments partagés.

Je remercie mes anciens professeurs depuis la première année scolaire jusqu'à ce point, qu'ils sont participés chacun de son côté afin d'avoir arrivé à ce niveau.

En fin, je remercie tous ceux qu'ils sont participés de loin ou de près de la réussite de ce travail de thèse.

Résumé

Les travaux présentés dans cette thèse se situent dans le cadre de la reconnaissance de l'écriture imprimée Arabe, en utilisant un ensemble d'approches au niveau de la phase de classification, à savoir l'algorithme Support Vecteur Machine (SVM) et K-Plus-Proche-Voisin (KPPV). L'objectif consiste à reconnaitre le contenu d'une image d'entrée. Ce dernier est réalisé via un système de reconnaissance constitué principalement de trois phases : prétraitement, extraction et classification.

Dans la première phase, nous avons traité le problème de la segmentation du texte arabe imprimé en caractères. Le processus de la segmentation commence par la détection des paragraphes en calculant l'histogramme horizontal et les interlignes du document pour déterminer les espaces verticaux séparant les paragraphes. Chaque paragraphe est segmenté en lignes en utilisant la projection horizontale de l'histogramme, en calculant cette fois-ci l'espace interligne du même paragraphe. Chaque ligne du paragraphe est segmentée en mots ou pseudo-mots, cela est réalisé à l'aide de l'histogramme vertical pour estimer l'espace inter mots dans la même ligne. La segmentation des mots en caractères est basée sur la suppression de la ligne de base et la projection de l'histogramme verticale. Les tests réalisés montrent des résultats très encourageants pour la segmentation du texte en lignes et la ligne en mots, par contre la segmentation des mots en caractères présente quelques difficultés.

Une deuxième approche a été proposée pour la reconnaissance des caractères isolés arabes. En effet; nous avons proposé une méthode d'extraction des caractéristiques appelée « Cadre de Niveau » pour la discrimination des caractères arabes imprimés, cette technique adopte une approche statistique basée sur des positions pouvant donner quelques informations sur la morphologie de la forme. Le processus de cette technique divise l'image binaire normalisée en une matrice carré d'ordre 100 d'un caractère en 100 ou 64 zones. Chacune d'elle est subdivisée en 5 niveaux, pour chaque niveau, des calculs sont effectués pour décrire la distribution et la densité des pixels, la moyenne des 5 valeurs extraites (une valeur pour chaque niveau) est retenue pour représenter une zone, ce qui donne un vecteur de 100 ou 64 variables caractérisant un caractère. Cette technique a été appliquée sur une base de données locale composée de : 105 classes des caractères arabes, 33 classes de différents caractères. Pour prédire les classes d'appartenance des caractères, nous avons utilisé l'approche de K-plus-proche-voisin en se basant sur trois types de distance (Correlation, Citybloc et Spearman). Les résultats obtenus sont très encourageants vis-à-vis la simplicité d'implémentation et la capacité discriminante de la méthode.

Dans la troisième approche, nous présentons une nouvelle méthode appelée « zigzag de poids de densité » pour la reconnaissance des mots arabes imprimés. Cette technique s'opère en deux étapes:
- La première vise à réduire la taille de la matrice d'image normalisée 96x96 en 12x12, en utilisant la technique de poids de densité.

- Dans la deuxième étape, la dernière matrice (12x12) a été utilisée, pour extraire 144 séquences suivant le chemin zigzag.

Les 144 primitives calculées sont adoptées pour représenter chaque mot dans la base de données. Cette technique a été testée sur les noms des villes et villages du Maroc en utilisant KPPV avec la règle de vote majoritaire et le classificateur SVM. Les meilleurs résultats ont été obtenus avec KPPV (k = 9) et SVM (noyau linéaire).

La dernière approche s'est focalisée sur la reconnaissance de la fonte des mots ou pseudo-mots de différentes fontes arabes appliquées sur plusieurs familles de fontes, tailles et styles. L'algorithme d'extraction proposé est basé sur la continuité des pixels pour les quatre directions matricielles et huit paramètres statistiques de l'histogramme pour extraire en total 20 primitives des dix derniers pixels du mot. L'algorithme proposé a été testé sur la base des mots arabe imprimés de basse résolution APTI. Les meilleurs résultats sont obtenus avec l'algorithme de classification KPPV.

Mots clés : Histogramme vertical, Histogramme horizontal, Cadre de Niveau, K-Plus-Proche-Voisin, Correlation distance, Cityblock distance, Spearman distance, zigzag de poids de densité, support vecteur machine (SVM). APTI, reconnaissance de l'écriture imprimée, Continuité des pixels, caractères Arabes.

Abstract

This thesis deals with an offline printed Arabic characters, recognition, adopting Vector Support Machine (SVM) and K-Nearest-Neighbor (KPPV) Models in classification step. The goal is to recognize content image of text, character Arabic, based on recognition system of three phases: preprocessing, features extraction and classification.

In the first phase, we treated the segmentation problem in Arabic character recognition. The segmentation process begins with the detection of paragraphs by analyzing the horizontal histogram and document spacing to estimate the vertical spacing between the paragraphs. Each paragraph is segmented into lines using the horizontal projection of the histogram, but this time the interline spacing of the same paragraph was analyzed. Each line of the paragraph is segmented into words or sub words, based on the vertical histogram to estimate the word spacing in the same text line. Words segmentation into characters is based on baseline detection and removing based on vertical histogram projection. The tests show very encouraging results for text segmentation into lines and line into words, whereas words segmentation into characters presents some difficulties.

In the second approach, we present a new feature extraction technique for off-line printed Arabic, called “Cadre of Level”. This technique adopts a statistical approach based on positions that can give some information on shape morphology. The process of this technique divides the normalized binary image into a square matrix (100x100) of character in 100 or 64 areas, each one is divided into 5 levels; for each level, distribution and density of pixels are calculated. The average of the extracted values (one value for each level) is used to represent an area, to extract vector of 100 or 64 features. This technique was applied on 105 classes of Arabic characters, 33 classes of different characters. The K-nearest-neighbor method approach based on three types of distance (Correlation, Citybloc and Spearman) was adopted to predict characters. The results obtained are encouraging, towards the simplicity of implementation and discriminating capacity of this approach.

In the third approach, we present a technique called “zigzag of density weight” to recognize printed Arabic names. This technique was performed on two steps, the first aims to reduce normalized matrix size of 96x96 into 12x12 using density weight technic, in the second step, the last matrix (12x12) was used to extract 144 features following path zigzag, to represent each name in data set. This technique was tested on Morocco Town and village names using KNN with consensus rule and SVM classifiers. The maximum score was obtained using KNN (k=9) and SVM (linear kernel).

In the last approach, we present our method for words or sub-words font recognition of different Arabic fonts applied to different font family, sizes and styles. The features extraction algorithm is based on the continuity of pixels in the four directions and some histogram statistical to extract 20 features from the last ten pixels of word. The proposed algorithm has been tested on the dataset of Arabic words printed of low-resolution APTI. The best score

results obtained using KNN classification algorithm show the performance of our approach towards other approaches.

Keywords: Vertical histogram, Horizontal histogram, Cadre of Level, K–Nearest-Neighbour (KNN), Density Weight Zigzag, Support Vector Machine (SVM). APTI, Arabic characters, printed script, Pixels continuity, structural, statistical, Correlation, Cityblock, Spearman.

Table des matières

Liste des tableaux

Liste des figures

Liste des abréviations

APTI : Images du texte Arabe Imprimé

ROCAI : Reconnaissance Optique de Caractères Arabes Imprimés

ROFA : Reconnaissance Optique de Fonte Arabe

ROC : Reconnaissance Optique de Caractères

APTID/MF : Arabic Printed Texte Image Dynamique/multifonte

BP-RN : Rétro Propagation- Réseau de Neurone

AOFR: Arabic Optical Font Recognition

MMC : Modèle Markov Caché

PMC: Perceptrons Multicouches

KPPV : K Plus Proche Voisins

SVM : Support à Vaste Marge

PC: Personal Computer

2D: Bidimensionnel

1D: Unidimensionnel

IRCAM : Institut Royal de la Culture Amazighe

CL : Cadre of Level

DEP : Distance Euclidienne Pondérée

CP : Continuité de Pixels

Introduction générale

La communication entre le système d'information et l'utilisateur s'effectue dans les deux sens : l'utilisateur introduit l'information et le système récupère et traite les données saisies et renvoi les résultats de traitement à l'utilisateur, en fonction des données saisies et le traitement demandé. Les applications de traitement de données standard, nécessitent que l'utilisateur introduise les informations nécessaires à la main, via un matériel physique, qui peut être un clavier, une souris ou autre pour l'exécution du traitement demandé, ce qui nécessite un temps variant en fonction de la quantité de données à introduire au système, en plus le temps nécessaire pour que la machine traite et renvoi les résultats à l'utilisateur.

Aujourd'hui, le besoin des algorithmes et systèmes d'information permettant la communication entre l'Homme et la machine, peut s'exprimer par l'automatisation des tâches répétitives quotidiennes, l'offrent d'un service de détection ou la lecture automatique des libellées ou d'autres informations écrites sur un support physique.

Afin que la récupération des informations soit automatique, sans avoir les saisies à la main, les nouvelles applications font la récupération de l'information en utilisant un système de reconnaissance de forme, qui peut être un caractère, un mot ou autre objet via une image contenant les formes à reconnaître qui est capturée par une caméra ou le support physique est scanné sous forme d'une image.

Le domaine de la reconnaissance de forme, regroupe plusieurs gammes de systèmes soit de reconnaissance de l'écriture arabe ou latine, manuscrite ou imprimée, de la parole, de plaque routière ou de la détection de certaines formes contenues dans des images médicales ou de satellite, le tri automatique du courrier postal pour la poste et la lecture automatique du chèque pour la banque ou autre.

Les systèmes de reconnaissance automatique de l'écriture, se basent sur les techniques de traitement d'image. En général, dans la phase du prétraitement, un ensemble d'algorithmes de discrimination pour décrire les caractéristiques principales de forme à reconnaître sont utilisés. Les techniques de l'intelligence artificielle sont adoptées dans la reconnaissance et la prise de décision.

La reconnaissance automatique du texte imprimé ou manuscrit est un processus informatique complexe, vise à convertir l'image d'un support physique contenant du texte (par exemple un papier) en texte codé en format numérique par exemple .txt ou .doc .docx manipulable par la machine.

Le processus de la reconnaissance du texte a connu plusieurs niveaux de difficultés dès la première phase jusqu'à la dernière. En effet, au niveau de la phase d'acquisition qui vise à numériser le support physique du texte, plusieurs facteurs influençant sur la qualité d'image produite, soit par la présence de bruit à cause de la poussière ou de mauvaise qualité du document (par exemple cas d'ancien papier), la mauvaise qualité relative à la résolution ou la

configuration de la machine d'acquisition, ou bien un mauvais positionnement sur la machine de numérisation, ce qui nécessite une autre phase nommée prétraitement afin d'améliorer la qualité d'image et résoudre les problèmes liés à l'étape d'acquisition.

Durant la phase du prétraitement, l'image produite de la phase d'acquisition soumise à certaines opérations de filtrage, binarisation et recadrage pour représenter l'information en deux classes (blanches et noires) et supprimer les informations inutiles. Une autre opération de correction d'inclinaison du texte et/ou du document dans la même phase. Le processus de segmentation du texte est un problème extrêmement difficile à mettre en œuvre, la grande variabilité liée aux habitudes des scripteurs ainsi aux styles et formes d'écriture (manuscrite, cursive ou imprimé avec de nombreuses fontes), cette opération vise à segmenter le texte en lignes puis segmenter les lignes en mots ou pseudo-mots, et enfin segmenter les mots en caractères.

La phase d'extraction des caractéristiques utilise les formes résultantes de la phase du prétraitement. Les approches statistiques, structurelles, géométriques, et syntaxiques sont adoptées pour une discrimination optimale de la forme sous un vecteur afin d'optimiser l'exploitation des ressources de la machine. L'ensemble des vecteurs d'extraction sont regroupés sous forme d'une base d'apprentissage, pour prédire les exemples du test, en utilisant les algorithmes de classification. Une dernière phase de post traitement tente de compléter et corriger les erreurs de la phase de classification.

Les recherches menées dans le domaine de la reconnaissance de l'écriture sont nombreuses, depuis les années soixante à ce jour, ce qui a donné des progrès considérables surtout dans la dernière décennie, et cela grâce à l'amélioration des performances et qualité des machines. L'apparition et l'amélioration des techniques et systèmes de classification de l'intelligence artificielle telle que K-plus-proche-voisin, les réseaux de neurones, les machines à vecteurs de support ou les modèles de Markov cachés. La création d'une nouvelle base de données internationale standard relative à l'écriture manuscrite et imprimée permet encore aux chercheurs de rapporter de façon crédible les performances de leurs approches dans ce domaine, avec la possibilité de les comparer avec d'autres approches.

Malgré les efforts et les progrès réalisés dans le domaine grâce aux nombreuses années d'investigation consacrées au sujet, aujourd'hui à mes connaissances, il n'existe pas de système fiable capable de traiter l'écriture naturelle dans sa globalité et particulièrement l'écriture arabe imprimée, vu la grande variabilité liée aux habitudes des scripteurs ainsi aux styles et formes d'écriture imprimée avec de nombreuses fontes. La majorité des résultats des travaux publiés dans la littérature ne traite que des cas restreints à des domaines d'application bien déterminés (reconnaissance d'un nombre limité des mots, adresses postales) ou à des catégories d'écriture très contraintes ne représentant qu'un aspect particulier de l'écriture courante. De ce fait, la reconnaissance automatique de l'écriture arabe imprimée reste encore un sujet de recherche actif, vu sa nature calligraphique et structurelle complexe. Plusieurs tentatives sont consacrées à ce type d'écriture. Ces efforts, ont permis le développement et l'expérimentation de plusieurs approches de reconnaissance, qui pouvant être classées en quatre approches : statistiques, structurelles, géométriques, et syntaxiques. Malgré tous les

systèmes proposés, aucun d'entre eux n'est considéré comme fiable et ne répond pas aux besoins demandés, ce qui nécessite la continuation de recherche dans ce domaine.

Le présent travail de thèse porte sur la reconnaissance de caractères arabes imprimés en mode hors ligne. Ce sujet innovant massivement demandé dans différents secteurs traitant l'information. L'histoire du sujet reflète des difficultés de conception, modélisation et implémentation, ce qui présente un grand défi.

Les travaux menés dans ce travail de thèse, sont aboutis à la proposition de quatre approches : la première approche est destinée à la segmentation du texte arabe imprimé, la deuxième approche traite la reconnaissance des caractères isolés Arabe, la troisième approche est réservée à la reconnaissance des mots arabes. Tandis que la quatrième approche s'intéresse à la reconnaissance de fonte par famille, par taille et par style.

En effet, dans la première approche, nous nous sommes intéressés à la segmentation du texte arabe imprimé en caractères, le processus de segmentation commence par la détection des paragraphes, en se basant sur le calcul des projections d'histogramme horizontal et le calcul des interlignes dans tout le document pour déterminer les espaces existants entre les paragraphes. Chaque paragraphe est segmenté en lignes en utilisant aussi les projections d'histogramme horizontal et en calculant cette fois-ci l'espace interligne du même paragraphe. Le problème à ce niveau se pose lorsque l'espace d'interligne est trop petit ou négligeable ce qui produit le chevauchement entre la ligne du texte et la ligne suivante, du fait que les points diacritiques se trouvent en dessous de la première ligne et les points diacritiques existant en dessus de la ligne de base de la deuxième ligne du texte se situent dans le même niveau horizontal. Pour chaque ligne de texte extrait du paragraphe, on calcule les projections d'histogramme vertical, pour déterminer les points de segmentation de la ligne du texte en mots ou pseudo-mots, nous calculons l'espace inter-mots dans la même ligne de texte.

La position et l'épaisseur de la ligne de base sont estimées pour la suppression de la ligne de base, avant la segmentation de la ligne en mots pour déterminer les points de segmentation des mots en caractères. Nous montrons que l'approche proposée donne des bons résultats surtout pour la segmentation du texte en lignes puis en mots avec la prise en considération des cas des lignes d'interligne nulle du même paragraphe.

Notre deuxième approche a été proposée pour la reconnaissance des caractères isolés, Arabes. En effet, les techniques d'extraction de caractéristiques sont importantes dans le processus de reconnaissance de caractères, parce qu'ils peuvent améliorer l'efficacité de la reconnaissance.

Dans ce cadre, nous avons proposé une méthode d'extraction de caractéristique appelée «Cadre de Niveau», pour calculer les primitives des caractères Arabes imprimés. Cette technique est basée sur les deux approches statistique et structurelle, c'est-à-dire le calcul est basé sur une approche statistique, mais dans des positions qui peuvent donner quelques informations sur la morphologie de la forme. Le processus de cette technique divise l'image

binaire d'un caractère en 100 zones, chaque zone est constituée de 5 niveaux. Pour chaque niveau, un ensemble de calculs sont effectués afin de décrire la distribution et la densité des pixels, la moyenne des 5 valeurs extraites (une valeur pour chaque niveau) est retenue pour représenter une zone, ce qui donne un vecteur de 100 variables caractérisant un caractère. Cette technique a été appliquée sur une base locale constituée de 105 classes des caractères arabes.

Une troisième approche a été adoptée pour la reconnaissance des mots arabes imprimés appelée « zigzag de poids de densité ». Cette technique s'opère en deux étapes; la première étape consiste à réduire la taille matricielle d'image normalisée de 96x96 en 12x12 en utilisant une fonction conçue pour le calcul des poids de densité, dans la deuxième étape la matrice (12x12) résultante de l'étape précédente a été utilisée pour extraire 144 séquences suivant le parcours zigzag. Les 144 caractéristiques discriminantes sont utilisées pour identifier chaque nom dans la base de données. Cette technique a été appliquée pour la reconnaissance des noms des villes et villages du Maroc. La modélisation du système proposé a été basée sur l'approche distance du classifieur K-Plus-Proche-Voisin (KPPV) avec la règle de vote majoritaire pour sélectionner la classe du mot, l'algorithme de séparateur à vaste marge (SVM) a été aussi utilisé avec l'approche un contre tous.

Dans la quatrième approche, une méthode de reconnaissance optique des mots ou pseudo-mots de différentes fontes arabe est appliquée sur une famille de fontes, tailles et styles, afin de l'intégrer dans un système global de reconnaissance optique des caractères arabes imprimés (ROCAI).

La base de données APTI [1] est utilisée pour extraire les dix derniers pixels pour chaque mot ou pseudo-mot, ce traitement a comme objectif d'élaborer une nouvelle base de données contenant les dix derniers pixels pour chaque mot ; le système proposé de reconnaissance optique de fonte arabe (ROFA) est basé sur cette nouvelle base de données, et un algorithme de calcul de la longueur des pixels continus de différente direction matriciel, en plus de quelques statistiques de projections d'histogramme pour extraire un ensemble de primitives qui sont au nombre de 20 .

La performance du système proposé a été testée en utilisant le classificateur KPPV avec trois différentes distances : en utilisant la distance Cityblock, la distance Euclidien et la distance Corrélation en se basant sur le vote majoritaire pour sélectionner la prise de décision.

En plus d'une introduction générale et d'une conclusion et perceptives. Ce mémoire est organisé en deux parties : La première partie qui comporte le premier chapitre qui s'articule sur l'étude bibliographique, les aspects théoriques et les techniques de classification. Tandis que la deuxième partie est consacrée à la présentation des contributions élaborées, cette dernière est composée de trois chapitres.

Dans l'introduction générale, un ensemble de points sont évoqués, à savoir : la présentation des motivations et les objectifs tracés pour ce travail, la détermination de la

problématique à résoudre et finalement la description du concept générale d'un système de reconnaissance des caractères.

Le chapitre 1 traite deux axes. Le premier axe est sous forme d'un état de l'art sur les systèmes de reconnaissance des fontes, des mots et des caractères. Le deuxième axe présente les concepts théoriques et généraux liés aux systèmes de reconnaissance d'écriture. Dans ce chapitre, nous avons étudié les différents aspects liés aux systèmes de reconnaissance hors ligne d'écriture, en commençant par le mode d'acquisition et les techniques de prétraitement applicables sur les images des mots ou caractères, en vue de produire une version nettoyée pour les phases restantes, ensuite, en passant par les différents aspects de la phase de segmentation aussi que les différentes approches d'extraction de primitives, et finalement en montrant les différentes méthodes de classification utilisée et les résultats obtenus.

Le chapitre 2 présente en détail le système de reconnaissance concernant les caractères Arabes imprimés. Au début, il explique l'approche de la segmentation adoptée qui se base sur la détection et la suppression de la ligne de base d'écriture pour déterminer les points de segmentation finaux. Ensuite, il montre la technique d'extraction utilisée pour calculer les primitives, l'approche présentée est testée aux caractères Arabes imprimés, sans et avec l'utilisation des squelettes.

Le chapitre 3 présente une approche de reconnaissance des mots arabe imprimés, qui est appliquée aux noms des villes et villages du Maroc. Cette méthode, appelée « Zigzag de poids de densité », a comme objectif la transformation d'image binaire des mots normalisée dans un espace de dimension fixe, basée sur la fonction de poids de densité, afin d'extraire les séquences composant la matrice d'image suivant le parcours zigzag.

Le chapitre 4 est réservé à la reconnaissance de la famille de fonte, de taille et de style des mots arabe. Il propose une nouvelle technique d'extraction de primitives nommée « Continuité de pixels » basée sur la distribution des pixels dans les quatre directions : horizontal, vertical, diagonal et antidiagonal. La longueur maximale, minimale ou moyenne des chaînes de pixels est calculée sans prendre en considération les chaînes de longueur d'un seul pixel. De plus, huit paramètres caractérisant la taille et le style de la famille de fonte sont considérés. Nous achevons ce chapitre par la présentation des résultats expérimentaux et une conclusion,

Le rapport est achevé par une conclusion générale qui résume le travail réalisé répondant aux objectifs fixés au départ. La performance des approches proposées, par rapport aux techniques existantes dans la littérature, est approuvée par les résultats expérimentaux obtenus.

Liste des publications.

1. Segmentation and Recognition of Arabic Printed Script. IAES International Journal of Artificial Intelligence (IJ-AI) Vol.2, No.1, March 2013, pp. 20~26
2. Printed Arabic Character Classification Using Cadre of Level Feature Extraction Technique. IJACSA Special Issue on Selected Papers from Third international symposium on Automatic Amazigh processing (SITACAM' 13)
3. Classification of Printed Moroccan Town and Village Names. Journal of Information Technology Research, 7 (4), 1-11, October-December 2014

Liste des communications.

1. Printed Arabic Character Classification Using Cadre of Level Feature Extraction Technique. Third international symposium on Automatic Amazigh processing (SITACAM' 13)
2. Printed Character Classification Using Cadre of Level Feature Extraction Technique and 2-D correlation coefficient. Third international symposium on Automatic Amazigh processing (SITACAM' 13)
3. Classification of Printed Arabic Names by Density Weight and Zigzag Sequence Method: Application to Moroccan Town & Village Names. CBI'14
4. Word and sub word Arabic font, Size and styles recognition using majority vote of different classifiers. CBI'15
5. Word and sub word Arabic font, Size and styles recognition using majority vote of different classifiers. CBI'17

Chapitre 1: État de l'art sur les systèmes de reconnaissance

1.1 Introduction

Dans ce chapitre, nous présentons une étude bibliographie sur les approches de segmentation de texte arabe imprimé, la reconnaissance des caractères et mots Arabes, ainsi les approches de reconnaissance de fonte de l'écriture arabe. Dans une autre section, nous abordons les déférents aspects liés aux systèmes de reconnaissance d'écriture et particulièrement Arabe imprimée, puis nous allons illustrer l'architecture générale d'un système de reconnaissance ; en particulier, il va aborder deux points : le premier concerne la présentation des trois approches de classification qui sont : K-plus-proches-voisins (KPPV), réseaux de neurones multicouches et support à vaste marge (SVM). Le deuxième point se focalise sur l'étude de la segmentation d'un texte en lignes, en mots et en caractères, à ce niveau, plusieurs méthodes seront abordées. Ce chapitre est achevé par la présentation des différentes formules de mesure de performance.

1.2 Étude bibliographie.

Dans cette section, nous présentons un état de l'art sur les méthodes de segmentation du texte Arabe, en suite, un détail d'un ensemble d'approches de reconnaissance (mot et caractère) sera donné, et finalement, un certain nombre de travaux de reconnaissance de fonte seront énumérés.

1.2.1 *Segmentation de texte*

Dans [2], Zaki et al présentent un état de l'art sur les méthodes de segmentation qui sont classées en neuf différentes méthodes basées sur des techniques déjà utilisées. Les avantages et les inconvénients de chacune de ces méthodes sont abordés.

Dans [3], Belghith et al proposent une approche de segmentation de textes arabes non-voyelles basée sur une analyse contextuelle des signes de ponctuation et de certaines particules, tels que les conjonctions de coordination. Ils présentent aussi un système appelé STAr qui est un segmenteur de textes arabes basé sur l'approche proposée. STAr accepte en entrée un texte arabe en format (.txt) et génère en sortie un texte segmenté en paragraphes et en phrases.

Dans [4], Baccour et al ont proposé une méthode de segmentation de textes arabes non-voyelles en phrases et en paragraphes. Cette approche a été basée sur une étude d'un corpus pour extraire un ensemble de règles permettant de déterminer les frontières des phrases à travers l'analyse contextuelles gauche et droite des signes de ponctuation, des conjonctions de coordination et de certains mots connecteurs. La segmentation des textes en paragraphes a été basée sur les signes de ponctuation et les retours chariot.

Dans [5], Touj et al ont proposé un algorithme de segmentation du texte arabe imprimé en caractères basé sur l'histogramme vertical et certaines règles. Ces règles sont basées non seulement sur les caractéristiques structurelles entre les régions d'arrière-plan et les composants des caractères, mais aussi les caractéristiques des caractères arabes isolés sont utilisées pour

vérifier si le pseudo-mot ne comporte qu'un seul caractère. Ensuite, l'histogramme vertical et d'autres règles sont utilisés pour trouver les points réels de segmentation. Enfin, le pseudo-mot est segmenté au niveau des points de segmentation. Les résultats expérimentaux montrent que l'algorithme atteint environ un taux de 94 % de segmentation correcte.

Dans [6], les auteurs propose une architecture basée sur la Transformée en Ondelettes pour la segmentation des mots persans/arabes imprimés en caractères. L'algorithme utilise une nouvelle transformation en Ondelettes par laquelle les coefficients d'Ondelettes extraits sont exploités, dans la détection des conteurs sous-jacents horizontaux et la ligne de base. La Projection des bords horizontaux et leur emplacement sur la ligne de base fournissent des points de segmentation. Cette méthode de classification distingue les vrais points de segmentation. L'algorithme est robuste contre le bruit, niveau de gris, fonte et taille de caractères. Les résultats de la simulation permettent une comparaison entre le nouvel algorithme et trois approches, en termes de précision, vitesse et robustesse contre le bruit Gaussien. Les résultats expérimentaux montrent la supériorité du système en termes de précision et montrent que l'algorithme proposé améliore la vitesse de reconnaissance d'un facteur d'au moins 2,5 fois.

Dans [7], Alginahi et al présentent une description des caractéristiques de l'écriture arabe avec une vue générale sur les systèmes OCR et un état de l'art détaillé principalement sur les techniques de segmentation des caractères arabes imprimés hors ligne.

1.2.2 *Approche de reconnaissance de caractères et mots*

Le travail de [5] propose une méthode de reconnaissance de l'écriture arabe imprimée basée sur la Transformée de Hough Généralisée (THG). Cette méthode utilise un modèle de reconnaissance de caractères arabes préalablement établi qui permet d'identifier et localiser les caractères dans les pseudos-mot par une méthode de segmentation basée reconnaissance.

Dans [8], les auteurs ont proposé une technique de reconnaissance automatique hors ligne du texte arabe imprimé avec les modèles de classification de Markov cachés. Le travail est basé sur les fenêtres chevauchées et non chevauchées de différente taille afin de générer 16 primitives de chaque bande verticale. Huit différentes familles de fonte de caractères arabes ont été utilisées pour les tests (À savoir. Arial, Tahoma, Akhbar, Thuluth, Naskh, Simplifié Arabe, Andalus et traditionnelle arabe). Expérimentalement, il a été prouvé que l'utilisation d'un nombre d'états (5 ou 7) et une taille d'indice discret d'un dictionnaire de référence (codebook) (128 ou 256) donne un taux de reconnaissance le plus élevé pour les déférentes fontes. Expérimentalement, il a été considéré chaque forme comme une classe différente entraînant un total de 126 classes (par rapport à 28 lettres arabes). Les taux de reconnaissance moyens obtenus étaient entre 98,08% et 99,89% pour les huit fontes utilisées.

Le travail de [9], présente une nouvelle approche pour traiter le problème de la reconnaissance de textes arabes imprimés de la machine. Il considère qu'en raison de la difficulté de reconnaissance des mots arabes cursifs, le texte doit être normalisé et segmenté pour être prêt pour la phase de reconnaissance. Le système de reconnaissance de caractères

arabes proposé dépend de multiple classificateur parallèle des réseaux de neurones. Le classificateur comporte deux phases. La première phase catégorise le caractère d'entrée dans l'un des huit groupes. La deuxième phase classifie le caractère dans l'une des classes de caractères arabes dans le groupe. Le système proposé a été testé sur plus de 100 images du texte arabes. Le taux de reconnaissance atteint le (98%).

Dans [10], Zayene et al ont proposé une approche de reconnaissance de textes arabes imprimés hors-ligne à vocabulaire ouvert et à très basse résolution (72 dpi). La méthode est basée sur les Modèles de Markov Cachés avec la boîte à outils HTK. Le système proposé est évalué sur trois fontes de calligraphie complexe et présentant de fortes ligatures Diwani Letter, DecoTypeNaskh et DecoTypeThuluth de la base APTI (Arabic Printed Text Image) basée sur les primitives statistiques et structurelles, le meilleur taux moyen obtenu est de 99.7% pour la reconnaissance des caractères et 96.2% pour la reconnaissance les mots.

1.2.3 *Approche de reconnaissance de fonte*

Dans [11], Sliman et al ont présenté une approche pour la reconnaissance de fonte arabe, basée sur l'utilisation d'une fenêtre glissante de dimension fixe pour l'extraction des primitives et le modèle de distributions de primitives Gaussien mixte (Gaussian Mixture Models (GMMs)).

L'approche proposée ne demande pas une segmentation apriori en caractères. Chaque image de mot est conservée en niveau de gris et normalisée en une hauteur fixe de taille H de 45 pixels. L'image prétraitée est transformée en une séquence X de N vecteurs caractéristiques (x_1….. x_N). Chaque vecteur caractéristique x_i est calculé à partir d'une fenêtre de taille 45×8 glissante de droite à gauche est décalé de 1 pixel sur l'image de mot. Le système proposé a été testé sur la base de données APTI (Arabic Printed Text Image) en utilisant dix tailles de police (6, 7, 8, 9,10, 12, 14, 16, 18 et 24) et les dix familles de fontes différentes : Advertising Bold, Andalus, Arabic Transparent, MUnicodeSaraTahoma, Simplified Arabic, Traditional Arabic, DecoType Naskh, DecoType Thuluth, Diwani Letter. Le taux global obtenu après une amélioration est de 99,1%.

Dans [12], Slimani et al ont proposé une méthode d'identification de fonte et de la taille pour les images des mots arabes de basse résolution avec l'utilisation d'une approche stochastique. Le travail a proposé une approche stochastique pour traiter le problème de reconnaissance de fonte et de taille. La méthode proposée traite les images des mots avec les fenêtres de langueur fixe et les fenêtres chevauchées, chaque fenêtre est représentée avec 102 primitives leurs distributions est capturée par le Modèle Gaussian Mixte (GMMs). Il présente trois systèmes : le premier pour la reconnaissance de la famille de fonte, le deuxième pour la reconnaissance de taille et le troisième pour la reconnaissance de taille et de la famille de fonte. Les résultats expérimentaux montrent l'importance d'identification de fonte avant la reconnaissance des images des mots avec deux multi-fonte Arabe OCRs (global et en cascade). Le système en cascade est meilleur que le système globale multi-fonte en termes de taux de reconnaissance de mot avec un taux de gain de 23% testé sur la base de données Arabic Printed Text Image (APTI).

Dans [13], les auteurs ont présenté un système de reconnaissance de caractères latin multi-fontes utilisant multi classifieurs. Chaque classifieur fournit une réponse puis le résultat final est obtenu par vote majoritaire. Les classifieurs sont de deux types : stochastique et plus proche voisin. Les classifieurs stochastiques sont des modèles de Markov cachés du premier et du second ordre. La reconnaissance des caractères est suivie d'un module de vérification lexicale qui utilise un modèle de Markov caché pour les mots dont les paramètres sont déterminés à partir des statistiques sur la langue et d'un dictionnaire.

Le travail présenté dans [14], traite des difficultés liées à l'identification de la fonte dans un contexte de lecture optique de l'Arabe multifonte. Ces difficultés découlent principalement de la complexité morphologique de cette écriture et de la grande variabilité d'une fonte à une autre, des dessins de différents caractères. Une approche d'identification des fontes arabes est proposée basée sur l'exploration des Ondelettes en paramétrant les réseaux de neurones en classification. Les différents tests effectués sur un jeu de 114229 pseudo-mots (chaînes de caractères) de neuf fontes, considérées dans cinq corps différents, ont conduit à des taux de reconnaissance très encourageants.

Le travail proposé dans [15] présente la première version d'un logiciel d'analyse et de reconnaissance de fontes anciennes. La principale originalité de ce travail est l'intégration d'un générateur d'images synthétiques de textes anciens dans la chaîne d'analyse. Ces images sont générées selon des critères spécifiés par l'utilisateur en termes de fontes utilisées, de nombre de lignes par image et de type de fond sur lequel sont intégrés les caractères. Plusieurs expérimentations ont été effectuées, montrant la génération d'images de documents anciens, en mettant à disposition une base d'images conséquente et adaptable permet de tester avec précision la qualité de la chaîne d'analyse de fontes anciennes.

Dans [16] les auteurs ont proposé un algorithme de reconnaissance de fonte arabe basée sur l'approche apriori. Tout d'abord, les mots dans l'ensemble de documents d'apprentissage pour chaque fonte sont segmentés en symboles ou graphèmes qui sont redimensionnés. Ensuite, des modèles sont construits, où chaque nouveau symbole d'entraînement qui n'est pas similaire aux modèles existants est un nouveau modèle. Les modèles sont partageables entre les fontes. Pour classifier la fonte d'un mot constituée d'une séquence de symboles, l'ensemble de ces symboles sont mis en correspondant avec les séquences des fontes d'apprentissage, les fontes de toutes les séquences sont concaténées pour former une liste de fontes F. La fonte la plus fréquente apparaissant dans F est sélectionnée comme la fonte reconnue du mot. Si F est vide la fonte de mot est inconnue.

Dans [17] Ibrahim et al ont présenté un algorithme pour la reconnaissance optique des fontes arabe avec l'approche apriori, appliqué sur quelques mots arabes communs. Une fois que ces fontes sont connues, elles peuvent être généralisées à des lignes, des paragraphes, ou mots voisins non-communs, depuis ces composants d'un matériau textuel ont presque la même fonte. Les arbres de décisions sont adoptés pour la classification des fontes. Un ensemble de 48 caractéristiques sont utilisées pour l'apprentissage de l'arbre. Ces primitives inclure les projections horizontales, les coefficients de Walsh, les moments invariants, et les attributs géométriques. Un ensemble de 36 fontes sont étudiées. Le taux de réussite global est de

90,8%. Certaines fontes montrent le taux de réussite de 100%. Le temps moyen nécessaire pour reconnaître la fonte du mot est d'environ 0.30 secondes.

Dans [18], les auteurs ont présenté une étude bibliographie de la littérature de recherche de la reconnaissance de fonte Arabe et Farsi et les bases de données utilisées. Les principales phases de systèmes sont étudiées, en passant par des prétraitements, ensuite les primitives utilisées et finalement la phase de classification. Tous les travaux des systèmes publiés de la langue arabe et Farsi, les plus courants pour les auteurs sont fuselés. Les avantages et les limites des techniques présentées et les domaines de recherche qui ne sont pas loin abordant la longue arabe/Farsi ainsi que les perspectives d'amélioration possible.

Dans [19], Zaghden et al ont présenté une étude pour la reconnaissance des fontes arabe imprimées avec les dimensions fractales appliquées sur les images des différents blocs écrits de différentes fontes. Une étude comparative a été présentée avec les autres méthodes élaborées auparavant du domaine de reconnaissance des fontes. Une étude comparative entre les taux de reconnaissance obtenus par les dimensions fractales et les ondelettes a été détaillée. Pour déterminer la technique la plus robuste, un test a été effectué pour la reconnaissance de fonte, mais cette fois sur des images bruitées.

Dans le travail de [20], Khosravi et al présentent une approche de reconnaissance de la fonte et taille de police d'une image de document Farsie. La méthode est basée sur la binarisation du document et ensuite l'analyse de l'effet de la binarisation sur le document, y compris la taille et la forme des points et des traits interrompus, qui sont formés en étape de validation. Le système proposé a été évalué sur une base de données comprenant 10 * 49 images de texte de 7 différentes fontes et 7 différentes tailles de polices sont formées à l'aide d'un logiciel de Paint. Le taux de reconnaissance de 95,7% est atteint.

Le tableau ci-dessus résume certains travaux réalisé dans le domaine de la reconnaissance des caractères, mots et fonte :

Référence et auteurs	Méthode d'extraction	Méthode de classification	Type de caractères	Base de données	Taux de reconnaissance
[20]	**La taille et la forme des points et des traits interrompus**		**Persan (font)**	**-**	**95,7%**
[9]	**Les moments invariants et svd**	**RN parallèle**	**Arabe**	**Plus de 100 images de texte**	**98%**
[11]	**Fenêtre glissante de langueur fixe**	**GMMs**	**Arabe (fonte)**	**APTI**	**99.1%**
[6]	**Fenêtre glissante**	**HMM**	**-Fonte Arabe** **-Mots arabes** **-Caractères arabes**	**APTI**	**91.9%** **93.7%** **98.4%**

[10]	**Ondelettes**	**HMM**	**Arabe (Font and size)**	**114229Pseudo-mots**	**96.1%**
[17]	**Horizontal projections, Walsh coefficients, invariant moments, et géométrique attributs**	**Arber de decision**	**Arabe(Font)**	**top 100 mots**	**90.8%**

Tableau 1-1 Ensemble de travaux concernant la reconnaissance de caractères, mots et fonte arabes

1.2.4 *Base de données*

Parmi les contraintes de développement, des systèmes de reconnaissance optique de caractères arabes (OCRA) est le manque des bases de données libre et standard, pour tester la robustesse des approches et solutions proposées. Dans ce qui suit, on présente par la suite la base de données APTI que nous avons exploitée pour la mesure de performance de notre système de reconnaissance de fonte.

Dans [21], Slimane et al ont élaboré une base de données composée des images de mots arabes imprimés. Le but de cette base de données est l'évaluation comparative à grande échelle du texte à vocabulaire ouvert entre les systèmes de reconnaissance Arabe, multifonte, multi-taille et multi-style. La base de données est composée de plusieurs fontes, tailles, et styles. Les images à basse résolution où l'anti-aliasing est générateur de bruit sur les caractères à reconnaître. La base de données est générée synthétiquement à l'aide d'un lexique de 113'284 mots, 10 fontes arabes, 10 tailles de police et 4 styles de fonte. La base de données contient 45313600 mots et en total de plus de 250 millions de caractères. Les annotations sont fournies pour chaque image. La base de données est appelée APTI (Arabic Printed Text Image).

1.3 Généralités sur les systèmes de reconnaissance d'écriture

1.3.1 *Définition d'un système de reconnaissance*

Un système de reconnaissance est un ensemble de processus visant à reconnaître une forme (caractère, symbole, …) contenue dans une image d'un support physique (papier, boîtier d'un produit …), capturée par un scanner ou caméra ou autre outil d'acquisition.

L'ensemble des procédés informatiques pour la conversion d'images de textes (imprimés, dactylographiés ou manuscrits) en fichiers texte, constituant un système de reconnaissance optique de caractères (ROC), ce type de systèmes permet aux ordinateurs de récupérer le texte dans l'image d'un texte imprimé et de le sauvegarder dans un fichier pouvant être exploité dans le traitement du texte pour éditer, copier, puis modifier, et stocké dans une base de données ou sur un autre support exploitable par un système informatique.

Les systèmes de reconnaissance optique de caractères (ROC) sont utilisés par les bibliothèques pour numériser et conserver leur archive. ROC est également utilisé pour traiter les chèques et les bordereaux de carte de crédit et trier le courrier. Des nombres importants de

magazines et lettres sont triés chaque jour par des machines ROC, ce qui accélère considérablement la distribution du courrier, et par la suite améliore la qualité de service des établissements utilisant ce type de système. Deux types différents de système (ROC), ayant chacun ses outils propres d'acquisition et ses algorithmes correspondants de reconnaissance.

1.3.2 *Reconnaissance en-ligne*

Les systèmes de reconnaissance d'écriture en-ligne, sont des systèmes interactifs et dynamiques puisque la reconnaissance s'opère pendant l'écriture, les formes écrites sont reconnues au fur et à mesure qu'ils sont écrits à la main, l'écriture est saisie naturellement à la main en utilisant un stylet ou stylo digital pour écrire sur une ardoise ou un écran. Ces systèmes sont utilisés dans plusieurs équipements électroniques : smartphone, IPhone, IPad, ou Tablette PC. Pour la reconnaissance d'écriture, ce mode est réservé généralement à l'écriture manuscrite.

La reconnaissance d'écriture en ligne présente des avantages énormes. Par exemple, l'absence de bruit du fait que l'écriture s'effectue sur une tablette spéciale, le stylo utilisé pour l'écriture dynamique (en-ligne), garde le même épaisseur de trait qui est en général un pixel, et par la suite le problème de variation d'épaisseur d'écriture est négligée [22], d'une autre le tracé d'écriture est squelettique et les données se présentent sous la forme d'une séquence de points dont les coordonnées sont en fonction du temps [23,24], ce qui donne la possibilité de récupérer des informations temporelles en relation avec les différents aspects pour identifier l'écriture, par exemple la pression et le levé du stylo, la vitesse et l'accélération.

La réponse en continu et l'interactivité du système de reconnaissance en-ligne présente un avantage majeur qui permet la possibilité de correction et de modification de l'écriture de manière interactive vu les possibilités énormes offertes via l'utilisation des supports électroniques [25].

Les systèmes de reconnaissance d'écriture en ligne et les systèmes de reconnaissance de la parole, traitent les données sous forme signal, pour cette raison, les chercheurs appliquent des techniques utilisées pour la reconnaissance de la parole pour la reconnaissance en-ligne d'écriture [26].

Le mode d'acquisition en ligne est utilisé dans plusieurs travaux de reconnaissance de mots ou phrases cursives en se basant sur la classification avec les réseaux de neurones [27, 28, 29,30], d'autre opérant sur la reconnaissance de caractères isolés avec les Support Vecteur Machines (SVM) [31] et d'autre en exploitant des informations temporelles dans l'écriture en ligne des caractères [32]

Les systèmes de reconnaissance dynamique procèdent globalement par une comparaison de la forme à reconnaître avec ceux contenus dans une base de données. Cette base de données peut être créée de toutes formes où être l'objet d'une phase d'apprentissage. Les techniques de comparaison reposent généralement sur des méthodes statistiques simples pour gagner en vitesse de traitement. La conséquence est que le nombre de formes reconnaissables doit être limité.

Les travaux de [33, 34] présentent un état de l'art des principaux systèmes avec l'analyse des différentes techniques de reconnaissance de l'écriture en ligne.

1.3.3 *Reconnaissance hors-ligne*

Ce type d'acquisition est appelé aussi statique, puisque l'écriture introduite au système ne change pas en fonction du temps. En effet dans les systèmes de reconnaissance, hors-ligne, le document manuscrit ou imprimé est écrit indépendamment du système de reconnaissance. Le processus d'acquisition avec le mode statique commence par la numérisation du papier d'écriture en utilisant un scanner ou une caméra, qui permet de produire la version numérique du support papier sous forme d'une image.

Les systèmes de reconnaissance hors-ligne ne fournirent aucune information temporelle et dynamique du tracé, dons ce cas la variation d'épaisseur d'écriture devient un facteur supplémentaire à prendre en compte.

Dans ces systèmes, toutes les informations sont présentées sous forme d'une image qui est exposée aux différents problèmes d'acquisitions telle que le bruit, l'inclinaison du document, le changement de la taille, la résolution du scan qui peut engendrer une perte d'information, d'où vient la nécessité d'une phase des prétraitements afin d'améliorer la qualité d'image produite et la quantité d'informations en utilisant les techniques de seuillage, squelettisation, atténuation ou élimination de bruit, normalisation et segmentation qui peut être beaucoup difficile et plus complexe, surtout dans le cas de l'écriture cursive.

La présentation du document ou papier d'écriture sous forme d'une image couleur, en niveaux de gris ou binaire, présente une perte d'information disponible en mode en ligne telle que l'ordre des points d'écriture en fonction du temps, de plus à la variabilité du tracé en épaisseur et en connectivité.

Ce mode peut être considéré comme le cas le plus général de la reconnaissance de l'écriture. Il se rapproche du mode de la reconnaissance visuelle. L'interprétation de l'information est indépendante de la source de génération [35].

1.4 Stratégie et difficulté de reconnaissance de l'écriture

1.4.1 *Stratégie de reconnaissance globale*

La stratégie de reconnaissance globale, considère le mot complet sans segmentation en caractères préalable comme des entités de bases. Chaque mot est modélisé par un modèle spécifique et caractérisé par des descripteurs globaux décrivant le mot complet, en évitant les problèmes liés avec le processus de segmentation, les primitives décrivant le mot sont en général : les profils haut/bas, les boucles, ascendants, descendants, croisements, longueur, points de terminaux et bien d'autres.

Cette approche est robuste, efficace et simple, car il ne se base pas sur la reconnaissance des caractères et/ou graphèmes composant le mot. Cette stratégie s'adapte avec les problèmes

de vocabulaire limité composé de quelques dizaines de classes, comme le cas des applications de traitement et vérification des chèques.

Dans le cas d'un dictionnaire de taille grande, contenant des mots similaires au niveau calligraphique, l'élaboration des modèles de chaque classe des mots nécessite l'utilisation des primitives de capacité discriminante de très grande précision, les primitives globales ne permet pas de décrire les détaille et la particularité de chaque mot, le temps de classification et la prise de décision se multiplie en fonction de la taille de la base d'apprentissage et aussi la taille des échantillon de test. Ainsi, cette approche est plus coûteuse et non fiable.

Plus que la taille des exemples d'apprentissage est importante, plus que la capacité discriminante des primitives globales diminue surtout lorsque le taux de similarité entre les classes de la population est important, par conséquent la performance générale des systèmes adoptants la stratégie de reconnaissance globale est dégradé plus que le nombre des classes augments. Pratiquement, l'approche holistique est restreinte à des vocabulaires de quelques dizaines de classes de mots au maximum [36,37].

1.4.2 *Stratégie de reconnaissance analytique*

Dans la section précédente, nous avons vu que l'approche de reconnaissance globale reste valable pour les vocabulaires des mots limités, mais pour réaliser un système de reconnaissance indépendant de la taille du vocabulaire, nous devons segmenter les mots en caractères ou en graphèmes de caractères, en se basant sur le fait que le nombre des caractères et/ou graphèmes pouvant composés un mot ou texte est en général limité. Dans le même contexte, l'approche analytique [38, 39, 40, 41], décompose le mot en séquence de caractères ou de graphèmes qui font partie d'un caractère. Dans cette approche, la reconnaissance du mot s'effectuée en combinant la reconnaissance des caractères intermédiaires et/ou graphèmes qu'il le compose. Il est donc nécessaire de découper le mot à reconnaître en une séquence de symboles (caractères, graphèmes, points diacritiques, boucles, …). L'opération de découpage n'est pas toujours triviale, d'où la nécessité d'une étape de segmentation pour déterminer les extrémités et les limites entre les entités composant le mot, mais cette tâche est particulièrement délicate et pouvant générer déférents types d'erreurs [42,43]. Le processus de reconnaissance basé sur cette approche s'opère en deux phases alternatives, une première phase pour la segmentation du mot en caractères et une deuxième phase pour l'identification des segments. Deux types de segmentation sont alors la segmentation explicite (externe) ou la segmentation implicite (en graphème) [41]. De plus, l'extraction des primitives est plus aisée sur un caractère que sur une chaîne de caractères [44].

1.4.3 *Difficulté de reconnaissance de l'écriture*

Dans les systèmes de reconnaissance de l'écriture, on trouve plusieurs niveaux de difficultés qui peuvent influencer dans les différentes phases de reconnaissance, ce qui complique le processus de la ROC. Parmi ces difficultés, on peut citer :

- **Acquisition :** la qualité du document à convertir en format numérique peut être entachée par plusieurs types des taches ou de qualité dégradée à cause de l'ancienneté, poussière ou

lorsqu'il s'agit d'une copie de l'origine. De plus, la phase d'acquisition peut générer, le problème d'inclinaison et une mauvaise qualité de numérisation

- **Prétraitement :**

  - La préséance de bruit à cause d'une mauvaise qualité de numérisation ou du document.
  - Correction d'inclinaison nécessite la détection de l'angle d'inclinaison et la corriger.
  - Segmentation : la localisation du texte avec la présence de d'autres formes (graphes tableaux, …), les lignes très rapprochées ou d'interligne nul, le chevauchement horizontal et ligature verticale des caractères, la nature de la ligne de base, sur-segmentation et la fonte utilisée.

- **Extraction des caractéristiques et classification** : forte similarité morphologique entre certains caractères et famille de la fonte.

1.5 Architecture d'un système de reconnaissance hors-ligne

La figure 1-1 illustre les déférentes étapes constituant la phase de reconnaissance et d'apprentissage, cette architecture s'adapte pour la reconnaissance des caractères et mots. Dans la suite, nous détaillons les techniques et méthodes utilisées pour bien mener un système de reconnaissance.

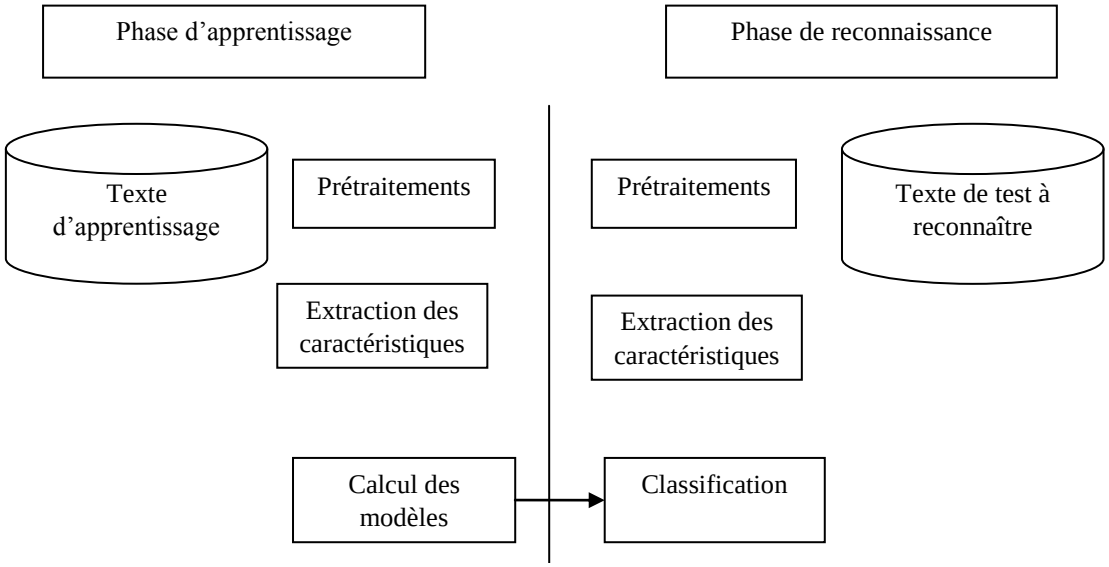


Figure 1-1 Schéma global d'un processus de reconnaissance des caractères.

1.5.1 *Description d'un système de reconnaissance*

Un système de reconnaissance de caractères hors-ligne, se compose en générale de quatre phases : Acquisition, prétraitements, extraction des caractéristiques et classification, que nous détaillons dans cette section.

1.6 Acquisition

L'acquisition est la première étape, dans le processus de reconnaissance de l'écriture. Dans cette étape, le document ou support physique contenant le texte est scanné par un scanner ou capturé par un appareil photo, puis enregistré sous forme d'une image numérique.

La phase d'acquisitions d'écriture est considérée comme une étape préliminaire pour les systèmes de reconnaissance d'écriture, l'objectif de cette technique est la numérisation des documents en format lisible par un ordinateur ou un système. Cette tâche, souffre de certaines difficultés en raison de la variabilité des formats et la qualité de présentation des formes dans les images. Dans le contexte de notre travail, on se restreint ces formes aux écritures des caractères et mots. En pratique, il y a deux modes principaux d'acquisition, le mode en ligne ou dynamique pour les écritures à reconnaître tout au cours de son écriture et le mode hors ligne ou statique pour les caractères et les mots déjà écrits sur des supports physiques par exemple un papier.

1.7 Prétraitements

L'image de la forme (caractère ou mot) acquise est représentée dans l'ordinateur par une dimension importante avec la présence de bruit et des problèmes d'alignement, cet ensemble de facteurs influence sur la qualité d'image [48]. Dans cette étape, une séquence d'opérations appartenant à la phase du prétraitement sont appliquées pour mettre l'image acquise dans un format adapté pour l'extraction des caractéristiques.

Dans cette phase, l'image acquise est soumise aux prétraitements suivants, pour produire une image bien nettoyée et de bonne qualité pour réussir les étapes restantes.

- ✓ Seuillage et réduction du bruit.
- ✓ Suppression des parties indésirables
- ✓ Détection et correction d'inclinaison
- ✓ Normalisation
- ✓ Segmentation

1.7.1 *Seuillage et réduction du bruit*

Le processus de seuillage consiste à représenter une image colore (RGB) au niveau de gris, dans laquelle un seuil est calculé ou fixer pour convertir l'image en binaire. Le choix ou le calcul d'un seuil approprié peut réduire le bruit entachant l'image.

Dans le processus du traitement d'écriture, pour distinguer les pixels de l'arrière plant et les pixels de l'écriture on adopte l'opération de seuillage. Cette opération se base sur le calcul d'un seuil, qui peut être divisé en deux types, le seuillage global qui s'adopte pour les documents d'arrière-plan simple et le seuillage adaptatif ou local qui s'adapte avec les images d'arrières plant complexe. Dans la suite, on présente les deux méthodes de seuillage global et local.

1.7.1.1 Seuillage global

Le seuillage global consiste à prendre un seuil pour toute l'image. Chaque pixel de l'image est comparé avec ce seuil et prend la valeur blanche ou noire selon s'il est supérieur ou inférieur. Cette méthode convient pour les documents simples et de bonne qualité et ne dépend alors que du niveau de gris du pixel considéré. En revanche, lorsque la qualité d'impression ou d'acquisitions du texte n'est pas constante dans toute la page, il n'est pas conseillé d'utiliser cette technique de seuillage. Dans [49], le choix arbitraire du seuil peut conduire à des pertes au niveau des informations utiles présentées dans l'image surtout si le fond est bruité et non-homogène ce qui présentes des taches parasites, la suppression des parties des caractères comme les points diacritiques, les parties des extrémités et voir aussi des impacts touchant les propriétés de la fonte des mots imprimés, ce qui touche la qualité totale des informations à traiter, par conséquence la performance générale du système de reconnaissance est aussi touchée. En analysant les histogrammes de la répartition des niveaux de gris des pixels de ces images contiennent deux pics nets, l'un présent le niveau de gris moyen pour le fond et l'autre présente le niveau de gris moyen de l'écriture, ce dernier niveau est retenu comme un seuil global d'image. Dans l'étape de binarisation les pixels de valeurs moins du seuil global sont remplacés par des pixels noirs et vice-versa [49].

1.7.1.2 Seuillage local

Dans le cas des documents de fond complexe, où l'intensité du fond et l'intensité de la forme varient dans le même document, il devient nécessaire de choisir le seuil de binarisation de manière local.

Dans ce cas, les étapes suivantes sont adoptées pour calculer l'image binaire d'mage de départ [50].

1- Calculer la valeur minimale de l'intensité dans le voisinage du pixel.
2- Calculer la valeur maximale de l'intensité dans le voisinage du pixel.
3- Calculer la valeur moyenne de l'intensité dans le voisinage du pixel.
4- Les pixels supérieurs à la moyenne de l'intensité sont remplacés par 1 (pixels blancs) et les pixels inférieurs à la moyenne de l'intensité sont remplacés par 0 (pixels noirs).

On calcule un seuil de binarisation pour chaque pixel d'image, en fonction de son voisinage.

1.7.2 *Suppression des parties indésirables*

À ce niveau du prétraitement, l'image binaire du texte comporte des marges haut, bas, gauche et droit, Le but c'est d'éliminer les marges indésirables dans l'image numérisée afin de minimiser l'espace utilisé de la mémoire et de réduire le temps de traitement.

1.7.3 *Normalisation de la taille*

La normalisation consiste à représenter les formes avec la même dimension. En particulier, la normalisation de la taille des caractères consiste à représenter tous les caractères selon une taille fixe, ce processus peut diminuer l'impact de changement de la taille.

La normalisation permet de ramener les images des mots à des tailles standard ou prédéfinies. Cette étape peut être indispensable pour certains types de systèmes qui traitent les images d'une taille prédéfinie. La différentielle pousse le principe de normalisation à un degré plus fin en essayant de normaliser localement les différentes parties du mot, de manière à augmenter la ressemblance d'une image à une autre. Les parasites, les hampes et les jambages provoquent des décalages verticaux des mots qui désynchronisent la présence des informations. Plusieurs techniques de normalisation de taille ont été proposées, mais aucune n'est considérée générale du fait que la performance générale varie suivant le type de graphie ou fonte de l'écriture. Dans [51] présente une méthode pour l'écriture latine.

Une technique de normalisation de la taille des caractères est présentée dans [52]. Cette technique est basée sur deux étapes : la première normalise le caractère en hauteur et la seconde en largeur en respectant l'ordre pour éviter que les caractères fins ne se déforment par rapport à des caractères épais.

Soit l'image de départ de dimension H_i, L_i. Nous voulons transformer cette image en une nouvelle image I' de dimension H et L, pour ce faire, nous utilisons une image intermédiaire de dimension H' et L' déterminée avec la formule suivante ?

$$P = H/H_i \text{ et } L' = P*L_i \qquad (1\text{-}1)$$

Première étape (normalisation en hauteur) : chaque pixel (x, y) noir d'image du caractère est transformé en (P*x, P*y).

Deuxième étape (normalisation en largeur) : la normalisation en largeur de l'image ainsi obtenue se fait par l'examen de deux cas.

Si L' > L, alors tout pixel de coordonnées ((x/L')*L, y) normalisée de l'image en hauteur est remplacé par un pixel noir.

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

1.7.4 *Détection et correction d'inclinaison des lignes (Skew correction)*

Dans la phase d'acquisition, le document à numérisé peut être mal positionné ce qui produit une image inclinée par rapport à l'axe horizontal, faisant un angle entre les lignes du texte et l'axe horizontal, c'est l'inclinaison.

Le processus de correction d'inclinaison consiste à calculer l'angle d'inclinaison puis appliquer une rotation dans le sens approprié suivant l'angle d'inclinaison.

L'opération de détection et de correction d'inclinaison est nécessaire dans le cas des documents qui présentent des écritures n'est pas bien aligné horizontalement, car il a un effet direct sur la fiabilité et l'efficacité des étapes de segmentation et d'extraction [53], cela est dû soit à un mauvais positionnement du document sur le scanner ou à une mise en page irrégulière

au moment de la rédaction. Plusieurs techniques sont utilisées pour la détection et la correction d'inclinaison de documents, nous citons par la suite les méthodes les plus utilisées.

La première est la détection d'inclinaison sur la base de la ligne de base ou la méthode de projections [54]. Dans cette méthode, l'image est projetée à plusieurs angles en général de -10° à +10° et la variance du nombre de pixels noirs par ligne de base projetée est déterminée. L'angle auquel la variance maximale se produit est l'angle d'inclinaison α du document. Cette méthode reste valable pour les documents à structure simple, mais il n'est pas convenable pour les documents contenant des graphes et de fond complexe.

Une seconde méthode de détection d'inclinaison basée sur la Transformée de Hough. La Transformation de Hough [55,56] est effectuée sur l'image du document numérisé et la variance des valeurs de ρ est calculée pour chaque valeur de θ, l'angle qui donne la variance maximale est l'angle d'inclinaison.

Le travail présenté dans [57] propose une méthode pour la détection de l'angle d'inclinaison de documents imprimés basée sur l'utilisation d'un polygone arbitraire et une dérivation de la ligne de base du centroïde du polygone. Cette méthode a été mise en œuvre sur 150 différents documents arabes numérisés, provenant de différentes sources, comme des revues, des manuels scolaires, des journaux et en plus du document manuscrit, avec différentes résolutions et différentes polices. Avec cette méthode, un taux d'exactitude de 87% a été obtenu.

Après avoir détecté et estimer l'angle d'inclinaison α (Par une des méthodes cités ci-dessus par exemple dans [54], l'image est tournée par l'angle estimé α dans la direction opposée pour faire la correction. Pour ce faire, tous les pixels de coordonnés (x, y) seront déplacés dans une nouvelle coordonnée (x', y') après la rotation de l'image entière autour de son origine par l'angle α, les nouvelles coordonnées (x', y') peuvent être calculées en fonction des anciennes coordonnées (x, y) par l'équation (1.2) :

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (1\text{-}2)$$

1.7.5 *Squelettisation*

Dans le processus de segmentation du texte et de reconnaissance des mots ou caractères, la variation d'épaisseur d'écriture influence sur les résultats obtenus. La technique de squelettisation vis à représenter l'écriture avec une épaisseur d'un pixel afin de rendre les formes d'écriture à traiter indépendante aux variations de trait d'écriture. Le calcul du squelette de la forme nous permet d'extraire plusieurs types de caractéristiques dont la majorité décrivant la structure, par exemple : les intersections, le nombre de tracés et leurs positions relatives, les propriétés métriques comme la hauteur et la longueur totale, distance entre les parties de la forme. En revanche l'utilisation du squelette simplifie la structure de la forme, ce qui rend les informations insuffisantes pour caractériser la fonte d'écriture. Pratiquement, l'extraction des primitives à partir du squelette de la forme minimise le temps d'exécution.

Le squelette de l'écriture est utilisé dans plusieurs travaux pour la segmentation ([58, 59, 60], normalisation [61, 62, 63] et extraction de primitives [64, 65,66, 67]).

Généralement, on peut classifier les types d'algorithme de squelettisation en deux catégories [47] :

- Les algorithmes de squelettisation séquentiels parcourant les pixels de forme les uns à la suite des autres pour extraire le squelette.

- Les algorithmes parallèles [68], où le traitement appliqué sur les pixels est indépendant, afin d'obtenir le squelette.

Une étude comparative de nombreux algorithmes de squelettisation présenté dans [69] sont appliqués sur différents caractères plus ou moins bruité en citant le nombre d'itérations et le temps de calcul pour chaque algorithme.

1.7.6 *Segmentation*

Dans le processus de la reconnaissance d'un texte imprimé ou manuscrit, la phase de la segmentation consiste à subdiviser le texte en trois étapes : premièrement, le partitionnement du texte en lignes, en suit, la subdivision de chaque ligne en pseudo-mots et finalement, la segmentation de chaque pseudo-mot en caractères(segmentation implicite) ou en graphème (segmentation explicite), les deux premières étapes de la segmentation sont suffisantes dans le cas d'une reconnaissance par approche globale et avec vocabulaire limité. En revanche, dans le cas d'une reconnaissance par approche analytique et avec vocabulaire ouvert, les trois étapes de la segmentation sont nécessaires. Les résultats des méthodes de la segmentation varient selon la langue traitée, la fonte utilisée, le chevauchement horizontal ou vertical des caractères, l'interligne et l'aspect morphologique de l'écriture. Dans ce qui suit, on considère que l'image binaire contienne un texte nettoyé.

1.7.6.1 Segmentation de texte en lignes

L'image binaire contenant le texte est exploitée dans cette section pour extraire les lignes du texte. La méthode la plus utilisée dans ce type de segmentation est la méthode de projection de l'histogramme horizontal [70] qui est sensible à la variation d'interligne et la fonte d'écriture.

1.7.6.2 Segmentation des lignes en mots

A ce niveau, chaque ligne du texte obtenue de l'étape précédente est découpée en mots ou parties des mots, l'espace inter-mot est exploité pour déterminer les points de segmentation, les espaces entre les parties des mots pour certaines fontes arabes provoquent une segmentation du même mot en plusieurs parties. La méthode la plus utilisée pour le découpage des lignes du texte arabe en mots est basée sur la projection de l'histogramme verticale. Pour éviter le problème de la segmentation du texte en mots, certains auteurs introduisent dans leurs systèmes, un seul mot à la fois [70, 71].

1.7.6.3 Segmentation des mots en caractères

La segmentation des lignes en mots suffit pour le cas d'un vocabulaire limité opérant avec l'approche globale, mais lorsque la taille du vocabulaire est illimitée ou dépasse un certain nombre ou même ouvert, cette fois-ci l'approche analytique est adoptée et qui considère que la segmentation des mots en lettres est nécessaire pour reconnaître la totalité des mots du texte. La segmentation des mots en caractères, consiste à découper les mots en lettres relativement prédéfinis et qui peut être dans différentes positions ou formes. La technique le plus utilisée se base sur l'exploitation de la projection verticale seule ou après la suppression de la ligne de base, mais les erreurs de segmentation varient suivant la complexité de la fonte de l'écriture, la présence des chevauchements verticaux et surtout horizontaux.

Cette étape de segmentation est le plus délicate dans le processus de la segmentation. Deux modes de segmentation sont adoptés pour mettre en place cette étape.

1.7.6.4 Segmentation explicite

La segmentation explicite s'intéresse au découpage du mot en plusieurs parties appelées graphèmes, et la reconnaissance consiste à regrouper un ensemble de graphèmes, pouvant composer une lettre. Le regroupement des lettres reconnues conduit à identifier le mot.

1.7.6.5 Segmentation implicite

La segmentation implicite concerne le partitionnement du mot en caractères et pas en graphèmes, avec cette approche, le processus de la segmentation commis plusieurs types d'erreurs :

- ✓ Sur-segmentation : la segmentation de certaines lettres en plusieurs petits graphèmes
- ✓ Fausse segmentation : pour les fontes présentant des chevauchements horizontaux, cette méthode découpe par erreur les caractères chevauchés en tant qu'un seul caractère.

1.8 Extraction des caractéristiques

Selon l'approche adoptée, globale ou analytique, la phase qui précède la reconnaissance des mots ou des lettres est la phase d'analyse de caractéristique pour décrire la forme (mot ou caractère). L'objectif de cette phase est d'estimer ou calculer un ensemble de paramètres liés aux aspects physique, synthétique, structurels et statistiques pour identifier la forme introduite au système de reconnaissance. Au niveau matriciel, également, l'objectif est de réduire le volume d'informations qui sera fourni au système par la transformation de l'image (caractère, mot) en un vecteur de primitives de taille fixe. Dans la suite, nous décrivons quatre catégories de caractéristiques : morphologiques, Statistiques, Structurelles et Transformations globales, ces caractéristiques peuvent être extraites à partir d'une image en plusieurs situations, à savoir : une image en niveau de gris, une image binaire, conteur et squelette.

1.8.1 *Caractéristiques statistiques*

Les caractéristiques statistiques fournissent certaines mesures statistiques extraites à partir de la forme [72, 73] décrivant la distribution des pixels, soit par zone ou dans les déférentes directions des matrices. Elles sont utilisées pour obtenir des informations locales décrivant l'écriture. Nous citons par la suite quelques caractéristiques statistiques [70,74, 75, 76].

- ✓ Le zonage, les caractéristiques de lieu et les moments géométriques [70].
- ✓ Les histogrammes (nombre de points noirs par colonne, par ligne, ou dans d'autres directions).
- ✓ La densité des pixels dans les parties chevauchantes ou non-chevauchantes de l'image.
- ✓ Le calcul de nombre et la longueur des segments blancs et des segments noirs le long d'une ligne verticale traversant la forme.
- ✓ La méthode de Leci, calcule le nombre de fois qu'un ensemble de lignes prédéterminées à différents angles (Ex. 16 lignes à 0°, 22.5°, 45°, etc.) traversent la forme, cette technique tolère des distorsions et des variations légères, et le calcul y facile [70].

1.8.2 *Transformations globales*

Ce type de caractéristiques vis à calculer une représentation alternative à la forme, par l'utilisation des algorithmes de transformation. Parmi ces transformations globales, on trouve :

- L'analyse de composante principale [77]
- La transformée de Karhunen-Loève [78]
- La transformée de Gabor [79]
- La transformée de Hough [80]
- La transformée de Fourier [80]
- Les moments [81]
- Les moments invariants sont indépendants au changement de la taille, la translation, et la rotation, ils sont utilisés comme des caractéristiques dans plusieurs travaux comme dans [82,83, 84]. Le temps demandé pour calculer les moments invariants limite la vitesse de lecture du système.
- Les ondelettes [85].
- La transformée de Walsh-Hadamard [86].
- La représentation du squelette ou du conteur de la forme comme une suite de code de différentes directions de Freeman [87, 71]
- La représentation de la forme par les projections horizontale et verticale [88].

1.8.3 *Analyse structurelle*

L'analyse structurelle consiste à extraire les caractéristiques et les propriétés topologiques et géométriques de la forme.

Parmi les caractéristiques structurelles, on trouve : concavités, convexités, occlusions, ascendants, descendants, composantes connexes, segments de droites et leurs attributs (position, orientation, langueur ...), mesures de déviation, arcs, boucles, croisements, jonctions des traits, paramètres de courbures, angularités, points extremums et points terminaux, longueur et épaisseur des traits, surfaces et les périmètres [89, 90, 91, 91, 92].

Ces primitives sont extraites à partir du squelette ou du contour et non de l'image brute de l'écriture. Les primitives structurelles ont une grande capacité de discrimination très forte, par contre la mauvaise détection de ces primitives génère automatiquement des résultats insatisfaisants pendant le processus de reconnaissance.

1.9 Techniques d'apprentissage

L'enfant avant d'être capable de parler la longue de sa famille ou de reconnaître les choses matérielles de déférentes formes, il apprend les noms de ses frères, sœurs et aussi les choses existantes dans la maison. Durant la phase d'apprentissage, l'enfant apprend en général à travers la répétition d'entendre ou de voir les choses des mêmes formes ou de formes déférentes, après la fin de la phase d'apprentissage, l'enfant devient capable de reconnaître ou deviner de quoi s'agit‘il; le même principe reste valable pour les systèmes intelligents de reconnaissance, mais dans ce cas, on distingue trois types d'apprentissage : apprentissage supervisé, non supervisé et par renforcement.

1.9.1 *Apprentissage supervisé*

Les formes à reconnaître sont connus a priori et le processus d'entraînement est guidé par le concepteur qui indique pour chaque forme une l'étiquette (nom, numéro, matricule, …) correspondante pour l'identifier en sortie. Ce type d'apprentissage est utilisé dans la majorité des systèmes de reconnaissance de l'écriture, et se termine par la génération d'un fichier ou structure ou modèle de référence à exploiter dans la phase de classification.

1.9.2 *Apprentissage non supervisé*

Dans ce type d'apprentissage, les formes ne sont pas connues a priori et le processus d'entraînement consiste à identifier les classes automatiquement en se basant sur les règles de modélisation ou de regroupement sans l'intervention du concepteur, les règles de regroupement doivent être précises et non contradictoires pour assurer une bonne séparation des classes.

1.9.3 *Apprentissage par renforcement*

Désigne l'ensemble des méthodes adaptatives qui permettent de résoudre un problème de décision en exploitant les expériences de l'utilisateur ; ce type d'apprentissage s'adapte avec les systèmes qui font la reconnaissance de l'écriture particulière d'un utilisateur [71], l'entraînement

s'effectue d'une manière dynamique, en interrogeant l'utilisateur pour désigner une nouvelle classe s'il n'est pas reconnaît préalablement pour reconnaître la forme (caractère ou mot) pour une autre fois.

1.10 Techniques de classification

Après l'extraction des caractéristiques de la forme à reconnaître et construction du modèle de références, les systèmes de la reconnaissance de caractères OCR procèdent automatiquement à la classification. Le processus de classification consiste à calculer la distance entre les primitives de références et celle de la forme en entrée à prédire, le résultat est une étiquette indiquant la classe d'appartenance de la forme introduite au système. Dans cette étape de prise de décision, il existe plusieurs méthodes pouvant être utilisées, à savoir : les méthodes connexionnistes, structurelles, syntaxique, méthode à noyau (Support à vaste marge (SVM) et la méthode de Markov Caché.

Pour prédire ou identifier la classe d'appartenance d'un caractère, mot ou d'une fonte, nous devons utiliser un algorithme de classification. Dans nos travaux de recherche, nous avons utilisé trois algorithmes d'apprentissage et classification qui s'inscrivent dans le cadre du type « supervisé : le réseau de neurones (RN), le support à vaste marge (SVM) et K-plus proche voisin (KPPV).

1.10.1 *K-plus proche voisin*

La méthode de classification des k-plus proches voisins est une technique de classification de deux types de données : données numériques et données nominaux (données binaires, données énumératives, données énumératives ordonnées). Cette méthode est basée sur l'apprentissage par analogie, en utilisant une fonction de distance pour estimer la vraisemblance d'un exemple en exploitant l'échantillon d'apprentissage, les distances entre l'exemple à prédire et les exemples d'apprentissage sont calculées, la décision est prise en utilisant une fonction de choix de la classe en fonction des classes des voisins les plus proches. La convergence de cette méthode est toujours assurée puisqu'elle ne construit aucun modèle.

1.10.1.1 Distance

Afin de trouver les K plus proches voisins d'une donnée à classer, l'utilisation de la notion de distance est nécessaire. Le choix de la distance est primordial au bon fonctionnement de la méthode. Les distances les plus simples permettent d'obtenir des résultats satisfaisants.

- ✓ **Propriétés de la distance:**

$$d(A, A) = 0 \quad (1.3)$$

$$d(A, B) = d(B, A) \quad (1.4)$$

$$d(A, B) \leq d(A, C) + d(B, C) \quad (1.5)$$

- ✓ **Distance entre numérique:**

$$d(x,y) = |x-y| \quad (1.6)$$

Où

$$d(x,y) = |x-y|/d_{max} \quad (1.7)$$

Où d_{max} est la distance maximale entre deux numériques du domaine considéré

- ✓ **Distance entre nominaux :**

Données binaires : 0 ou 1. On choisit : $d(0,0) = d(1,1) = 0$ et $d(0,1) = d(1,0) = 1$.

Données énumératives : la distance vaut 0 si les valeurs sont égales et 1 sinon 0.

Données énumératives ordonnées : elles peuvent être considérées comme des valeurs énumératives, mais on peut également définir une distance utilisant la relation d'ordre.

- ✓ **Distance Euclidienne entre 2 exemples**

Soit X = (x_1,..., x_n) et Y = (y_1,..., y_n) deux exemples, la distance euclidienne entre X et Y est:

$$D(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (1.8)$$

- ✓ **Sommation:**

$$D(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)} \quad (1.9)$$

- ✓ **Distance euclidienne pondérée:**

$$D(X,Y) = \sqrt{w_i \sum_{i=1}^{n}(x_i - y_i)^2} \quad (1.10)$$

- ✓ **Distance de Spearman**

La corrélation de Spearman consiste à trouver un coefficient de corrélation, non pas entre les valeurs prises par les deux variables mais entre les rangs de ces valeurs.

Étant donné une matrice X de taille (m×n) qui est considérée comme des vecteurs lignes m× (1 n) x_1, x_2... x_m, et une matrice Y, qui est traitée comme (1-par-n) (1×n) vecteurs lignes y_1, y_2,..., y_m, la distance de Spearman entre le vecteur x_s et y_s est définie comme suit :

$$d_{st} = 1 - \frac{(r_s - \bar{r_s})(r_t - \bar{r_t})^t}{\sqrt{(r_s - \bar{r_s})(r_s - \bar{r_s})^t}\sqrt{(r_t - \bar{r_t})(r_t - \bar{r_t})^t}} \quad (1.11)$$

Où r_{sj} est le rang de x_{sj} (x_{1j}, x_{2j}, ...x_{mj}), et r_{tj} est le rang de y_{tj}(y_{1j}, y_{2j}, ...y_{mj}), r_s et r_t sont les coordonnées des rangs vecteurs de x_s et y_t, c.-à-d., r_s = (r_{s1}, r_{s2}, ... r_{sn}) et r_t = (r_{t1}, r_{t2}, ... r_{tn})

$$\bar{r_s} = \frac{1}{n}\sum_j r_{sj} = \frac{(n+1)}{2} \quad (1.12)$$

$$\bar{r_t} = \frac{1}{n}\sum_j r_{tj} = \frac{(n+1)}{2} \quad (1.13)$$

- ✓ **Distance de Cityblock.**

$$d_{st} = \sum_{j=1}^{n} |x_{sj} - y_{tj}| \quad (1.14)$$

Notez que la distance de Citybloc est un cas particulier de la distance de Minkowski, où p = 1.

- ✓ **Distance de Corrélation.**

$$d_{st} = 1 - \frac{(x_s - \bar{x_s})(y_t - \bar{y_t})^t}{\sqrt{(x_s - \bar{x_s})(x_s - \bar{x_s})^t}\sqrt{(y_t - \bar{y_t})(y_t - \bar{y_t})^t}} \quad (1.15)$$

Où

$$\bar{x_s} = \frac{1}{n}\sum_j x_{sj} \quad (1.16)$$

Et

$$\bar{y_t} = \frac{1}{n}\sum_j y_{tj} \quad (1.17)$$

1.10.1.2Algorithme de K-PPV

L'algorithme de K-plus-proche-voisin est une approche très simple et directe. Elle ne nécessite pas d'apprentissage, mais simplement le stockage des données d'apprentissage. Son principe est de comparer une classe inconnue à toutes les données stockées. On choisit pour la

nouvelle donnée la classe majoritaire parmi ses K plus proches voisins (elle peut donc être lourde pour des grandes bases de données) au sens d'une distance choisie.

Algorithme :

- ✓ **Paramètre :** le nombre k de voisins.
- ✓ **Donnée :** un échantillon de m exemples et leurs classes.
  - o La classe d'un exemple X est c(X).
- ✓ **Entrée :** un enregistrement Y.
- ✓ 1. Déterminer les K plus proches exemples de Y en calculant les distances.
- ✓ 2. Combiner les classes de ces k exemples en une classe c.
- ✓ Sortie : la classe de Y est c(Y).

Pour le cas des K-PPV pour une valeur de K =1. L'algorithme est donné par le pseudo-code ci-après.

Algorithme 1 : Cas d'un seul plus proche voisin $k = 1$

Données :

✓Un échantillon d'apprentissage de n exemples et leurs classes $X^{train} = (x_1^{train}, \ldots, x_n^{train})$
Classes d'échantillon d'apprentissage $Z^{train} = (z_1^{train}, \ldots, z_n^{train})$.

✓Un échantillon de teste de m exemples et leurs classes $X^{test} = (x_1^{test}, \ldots, x_m^{test})$.

Algorithme 1PPV:

Pour $i \leftarrow$ de 1 à m
Pour $j \leftarrow$ de 1 à n
Calculer la distance entre x_i^{test} et x_j^{train}
$d_j \leftarrow d(x_i^{test}, x_j^{train})$
Fin
Calculer la classe z_i^{test} de l'ième exemple qui vaut la classe de son PPV:
Trouver l'indice du PPV de x_i^{test}:
$ind\_ppv_i \leftarrow arg\ min\ (d_1, \ldots, d_n)$
Trouver la classe le PPV de x_i^{tes} :
$z_i^{test} = z_{ind\_ppvi}^{train}$
Fin

Résultat: classes d'échantillons de test $z^{test} = (z_1^{test}, \ldots, z_n^{test})$

Pour le cas des K-PPV pour une valeur de K ≥ 1. L'algorithme est donné par le pseudo-code ci-après.

Algorithme 2 : Cas de plus d'un seul plus proche voisin $k \geq 1$

Données :

✓Un échantillon d'apprentissage de n exemples et leurs classes $X^{train} = (x_1^{train}, \ldots, x_n^{train})$ Classes d'échantillon d'apprentissage $Z^{train} = (z_1^{train}, \ldots, z_n^{train})$.

✓Un échantillon de test de m exemples et leurs classes $X^{test}= (x_1^{test}, \ldots, x_m^{test})$.

Algorithme KPPV:

Pour *i*← de 1 à *m*
Pour *j*← de 1 à *n*
Calculer la distance entre x_i^{test} et x_j^{train}

$$d_j = d\,(x_i^{test}, x_j^{train})$$

Fin

Calculer la classe z_i^{test} de l'ième exemple qui vaut la classe de son PPV:
Trouver l'indice des KPPV de x_i^{test}:

$$ind\_ppv_i = argmin\,(d_1, \ldots, d_n)$$

Trouver la classe du ppv de x_i^{test} : $z_i^{test} = z_{ind\_ppvi}^{train}$
Trier les distances d_j selon un ordre croissant pour $j = 1, \ldots, n$
Récupérer en même temps les indices IndVoisins avant le tri des dj
Récupérer les classes des K premiers ppv à partir des indices *IndVoisins* et en
Trouver la classe majoritaire :

$$C_k \leftarrow 0\,(k = 1, \ldots, K)$$

Pour k ← 1 to K do

$$ind\_voisin_k \leftarrow IndVoisins_k$$

$$h \leftarrow z_{ind\_voisin_k}^{train}$$

$$C_h = z_h + 1$$

Fin

/* trouver la classe du ppv de x_i^{test} :
(La classe majoritaire de celles de ses K-ppv) */ :
$z_i^{test} = argmax_{k=1}^{K}(C_k)$
Fin

Résultat: classes d'échantillons de test $z^{test} = (z_1^{test}, \ldots, z_n^{test})$

Dans le cas où la distance entre plusieurs classes différentes est la même, on utilise une fonction de choix aléatoire pour déterminer les K plus proches voisin.

1.10.2 *Perceptrons multicouches*

Les perceptrons multicouches (PMC) sont Apparus en 1985, aujourd'hui ces modèles sont les plus employés [93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104]. Plusieurs couches de traitement leur permettent de réaliser des associations non-linéaires entre l'entrée et la sortie [105]. Ce type de réseau est dans la famille générale des réseaux à « propagation vers l'avant », c'est-à-dire qu'en mode normal d'utilisation, l'information se propage dans un sens unique, des entrées vers les sorties sans aucune rétroaction. Son apprentissage est de type supervisé, par correction des erreurs. Dans ce cas, uniquement, le signal d'erreur est « rétro

propagé » des sorties vers les entrées pour mettre à jour les poids des neurones. Le PMC est un des RN les plus utilisés pour des problèmes d'approximation, de classification et de prédiction [106].

L'architecture du PMC se compose d'au moins trois couches de neurones. Chaque neurone de la première couche est relié à la suivante par une connectivité totale, et ce, jusqu'à la couche de sortie (figure 1-2.). La première couche est celle d'entrée. La dernière est la couche de sortie, tandis que les couches intermédiaires sont des couches cachées. La fonction d'activation des neurones des couches cachées et des neurones de la couche de sortie est la fonction sigmoïde. Cette différence par rapport au perceptron est essentielle à l'entraînement du PMC et permet le traitement de problèmes de façon non-linéaires.

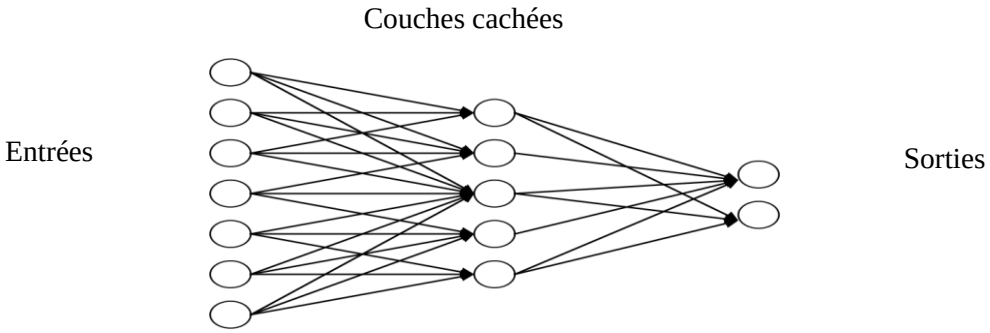


Figure 1-2 Le perceptron multicouches.

1.10.2.1 Algorithme d'apprentissage de rétropropagation

L'algorithme de rétropropagation est une méthode de calcul des poids pour un réseau à apprentissage supervisé qui consiste à minimiser l'erreur quadratique de sortie (somme des carrés de l'erreur de chaque composante entre la sortie réelle et la sortie désirée) [107].

L'algorithme de la rétropropagation de gradient, nécessite un certain savoir-faire pour une utilisation efficace. En effet, la convergence de l'algorithme n'est pas prouvée et de multiples variables sont à ajuster précisément en fonction du problème traité. Parmi ces variables à fixer, les paramètres apparaissant dans les différentes équations, la sélection des exemples pour l'apprentissage et le test, la structure du réseau (nombre de couches, taille de la couche cachée), la configuration initiale des poids et le nombre d'itérations d'apprentissage [108].

Le fonctionnement de l'algorithme de rétro-propagation à ce déroulé comme suite :
soient

$x= (x_1, x_2, ..., x_n)$, vecteur d'entrée
$o= (o_1, o_2, ..., o_m)$, vecteur de sortie désiré
$y= (y_1, y_2, \ldots, y_m)$ vecteur de sortie obtenu (réel)

1. Présenter un vecteur d'entrée aux nœuds d'entrées puis initialiser les poids du réseau.

2. Exécuter l'apprentissage d'échantillon à travers le réseau ;

3. Calculer les termes d'erreur du signal de la couche sortie et les couches cachées en utilisant respectivement les équations suivantes :

$$es = (o - x)f'(y) \tag{1.18}$$

$$ec = f'(y)\sum_{1}^{m} es.w \tag{1.19}$$

Avec :

es: erreur de la couche de sortie,

ec: erreur de la couche cachée.

x : Vecteur d'entré de la couche de sortie (signal de la couche cachée).

f: Fonction sigmoïde, tel que : $f(x) = 1/(1 + e^{-x})$.

f' : le dérivé de la fonction f.

4. Mise à jour des poids de la couche de sortie et des couches cachées en utilisant respectivement les équations suivantes :

$$w(t+1) = w(t) + \eta.\, es.x \tag{1.20}$$

$$w(t+1) = w(t) + \eta.\, ec.x \tag{1.21}$$

Avec

η : représente le taux d'apprentissage inférieur à 1.

5- Répéter ce processus jusqu'à ce que l'erreur E devienne acceptable (aller à 2)

$$E = \frac{1}{2}\sum_{k=1}^{m}(o_k - x_k)^2 \tag{1.22}$$

m : représente le nombre d'observations d'apprentissage.

o_k : représente la prévision (sortie du réseau).

x_k représente de la valeur cible pour la *k*-ième observation.

Plus la différence entre les prévisions du réseau et les valeurs cible sera importante, plus la valeur de l'erreur sera grande, ce qui nécessite alors un ajustement plus important des poids par l'algorithme d'apprentissage

La convergence du réseau par rétropropagation est un problème crucial qui requiert de nombreuses itérations. Pour remédier à ce problème, un paramètre est souvent rajouté pour accélérer la convergence. Ce paramètre est appelé « le fomentâmes ». C'est un terme d'inertie dont le rôle est de filtrer les oscillations dans la trajectoire de la descente du gradient. Donc le

fomentâme est un moyen efficace pour accélérer l'apprentissage et aussi pour pouvoir sortir des minimums locaux [105]. La règle de mise à jour des poids devient alors :

$$w(t+1) = w(t) + \eta . x + \alpha(w(t) - w(t-1)) \quad (1.23)$$

Où $0 <= \alpha < 1$ s'appelle le fomentâme : le terme du fomentâmes produit deux effets distincts selon la situation. Premièrement, lorsque la trajectoire du gradient a tendance à osciller, il contribue à la stabilisation en ralentissant les changements de direction. Par exemple, avec $\alpha = 0.8$, cela rajoute 80 % du changement précédent au changement courant. Deuxièmement, lorsque le gradient courant pointe dans la même direction que le gradient précédent, le terme d'inertie contribue à augmenter l'ampleur du pas.

1.10.2.2Évaluation du réseau multicouche

Le succès de la rétro-propagation témoigne de son efficacité. Mais l'algorithme le plus utilisé, il n'en souffre pas moins d'importantes limitations [108] :

- L'algorithme de rétropropagation est connu pour sa lourdeur et la lenteur de sa convergence dès lors que le problème devient un peu conséquent.

- Manque de bases analytiques : on ne dispose par exemple d'aucun outil fiable de dimensionnement du réseau plus particulièrement en ce qui concerne le nombre et la taille des couches cachées [108].

- Si le réseau est bien conçu, les RdN offrent des performances équivalentes à celles des meilleurs classifieurs en termes de taux de reconnaissance, mais ils se distinguent de ceux-ci par une plus grande facilité d'implantation sous forme de circuits spécialisés très rapides. Néanmoins, il faut souligner que l'on doit bien se garder d'utiliser systématiquement les RdN pour tout problème de classification. Il faut d'abord évaluer la difficulté du problème à traiter, et n'utiliser les RdN que lorsque c'est réellement nécessaire [106, 109, 1110].

1.10.3 *Support à vaste marge (SVM)*

Le SVM [111, 112] est un nouveau type de classifieur cherchant un hyperplan pour séparer les données en deux classes basé sur la théorie d'apprentissage statistique de Vapnik [113], dans le but de maximiser une marge géométrique de l'hyperplan, qui est lié à l'erreur de généralisation. La recherche de SVMs a connu un essor depuis les années 1990, et l'application de SVMs pour la reconnaissance de formes a donné une bonne performance d'après l'état de l'art. Généralement, le classifieur SVM est un classifieur linéaire et binaire (deux classes) dans l'espace induite par le noyau caractéristique et se présente sous forme d'une combinaison pondérée de fonctions noyau sur les exemples d'apprentissage. La fonction noyau représente le produit scalaire de deux vecteurs dans l'espace de fonction linéaires/non linéaires. Dans l'espace linéaire de fonction, la fonction de décision, principalement elle est comme une combinaison pondérée des fonctions du noyau, peut être convertie en une combinaison linéaire de fonctions de modèle. Ainsi, elle a la même forme que le réseau de neurone monocouche.

Pour illustrer le fonctionnement du classifieur SVM, Imaginons un plan (espace à deux dimensions) dans lequel sont répartis deux groupes de points. Ces points sont associés à un groupe : les points (+) pour *y>x* et les points (-) pour *y<x*. On peut trouver un séparateur linéaire évident dans cet exemple, la droite d'équation *y = x*. Le problème est dit linéairement séparable (Figure 1-3).

Pour des problèmes plus compliqués, il n'existe en général pas de séparateur linéaire. Imaginons par exemple un plan dans lequel les points (-) sont regroupés à l'intérieur d'un cercle, avec des points (+) tout autour : aucun séparateur linéaire ne peut correctement séparer les groupes : le problème n'est pas linéairement séparable. Il n'existe pas d'hyperplan séparateur.

1.10.3.1 Cas linéairement séparable

Dans le cas de deux classes linéairement séparables. Il existe une infinité d'hyperplan capable de séparer parfaitement ces deux classes. Pour toutes les formes x_i de classe y_i de la base d'apprentissage, on a :

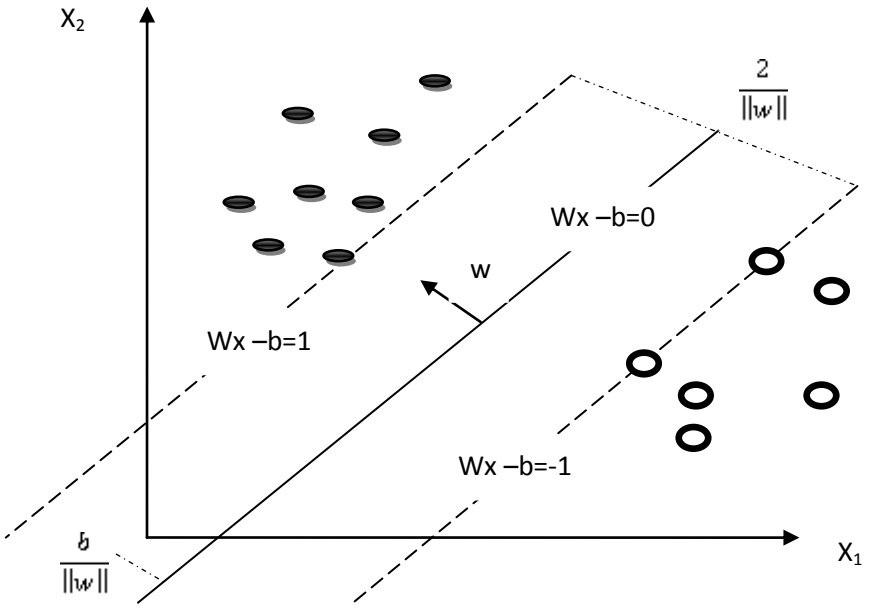


Figure 1-3 Hyperplan classifieur pour classification binaire et vaste marge (distance entre les deux classes).

Avec $\frac{1}{||w||}$ est la marge et (w,b) est le vecteur support à optimiser pour trouver un hyperplan séparateur.

Dans le cas séparable figure 1-3, on considère les points les plus près de l'hyperplan séparateur : vecteurs supports.

Pour tout point de l'espace des exemples, la distance à l'hyperplan séparateur est donnée par :

$$\frac{|w \cdot x + b|}{\|w\|} \tag{1.24}$$

De manière équivalente, le problème peut s'écrire plus simplement comme la minimisation de :

$$\frac{1}{2}\|w\|^2 \tag{1.25}$$

Sous les contraintes :

$$y_i(w \cdot x + b) \geq 1, \forall i \in [1, N] \tag{1.26}$$

La minimisation est possible sous les conditions dites de "Karush-Kuhn-Tucker (KKT)"

Soit le Lagrangien L :

$$L(w, b, \lambda) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \lambda_i [y_i(w.x_i + b) - 1] \tag{1.27}$$

Avec $\lambda \geq 0$.

Les conditions de KKT sont alors :

$$\frac{\partial L}{\partial w} = 0, \frac{\partial L}{\partial b} = 0, \frac{\partial L}{\partial \lambda_i} \geq 0. \lambda_i \geq 0 \tag{1.28}$$

$$\lambda_i [y_i(w.x_i + b) - 1] = 0 \tag{1.29}$$

Par ailleurs la dernière condition implique que pour tout point ne vérifiant pas $y_i(w \cdot x_i + b) = 1$ le λ_i est nul.

Les points qui vérifient $y_i(w \cdot x_i + b) = 1$, sont appelés "vecteurs supports". Ce sont les points les plus près de la marge. Ils sont censés être peu nombreux par rapport à l'ensemble des exemples.

Le problème s'exprime sous forme duale comme la minimisation de :

$$w(\lambda) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \lambda_i \lambda_j y_i y_j (x_i . x_j) \tag{1.30}$$

Ce problème fait partie des problèmes d'optimisation quadratique pour lesquels il existe de nombreux algorithmes de résolution ; dans la pratique, on utilisera les librairies SVM-Light de Joachim [114] ou la méthode SMO implémentée par Platt [115].

1.10.3.2Cas non linéairement séparable

Et si les données ne sont pas linéairement séparables ? L'idée est d'ajouter des variables d'ajustement ξ_i figure 1-3 dans la formulation pour prendre en compte les erreurs de classification ou le brui.

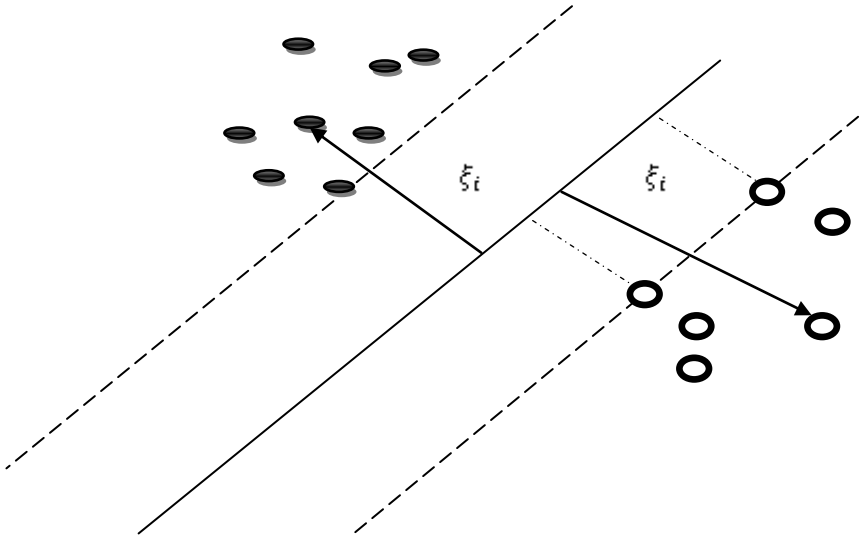


Figure 1-4 Hyperplan classifieur pour classification binaire avec variable d'ajustement et vaste marge [Arnaud Revel, Séparateurs à vaste marge]

Pour le cas de classes non linéairement séparables, on relâche la contrainte de bon classement, initialement pour toutes les formes $x_i$de classe $u_i$de la base d'apprentissage, on a :

$$u_i(w^t x_j + w_0) > 1 \tag{1.31}$$

Devient:

$$u_i(w^t x_j + w_0) \geq 1 - \xi_j \tag{1.32}$$

Et on ne doit plus minimiser

$$\frac{1}{2}\|w\|^2 \tag{1.33}$$

Mais

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \tag{1.34}$$

Avec C ≥ 0 est une constante permettant de contrôler le compromis entre le nombre d'erreurs de classement, et la largeur de la marge.

L'espace des données peut toujours être plongé dans un espace de plus grande dimension dans lequel les données peuvent être séparées linéairement.

Il existe plusieurs types de noyau :

- ✓ Noyau Linéaire (simple produit scalaire) :

$$K(x, x_i) = x * x_i \tag{1.35}$$

- ✓ Noyau Radial Basis Fonction (RBF) :

$$K(x, x_i) = \exp(-\gamma \|x * x_i\|^2) \quad (1.36)$$

- ✓ Noyau Polynomial :

$$K(x, x_i) = (x * x_i + c)^2 \quad (1.37)$$

- ✓ Noyau Sigmoïde:

$$K(x, x_i) = \tanh(x * x_i + c) \quad (1.38)$$

Les noyaux RBF semblent donner les meilleurs résultats, dans le cas de la reconnaissance de manuscrits [116].

1.10.3.3Stratège de reconnaissance

L'adaptation des SVM biclasses au cas multiclasse peut se faire de trois façons différentes. Le choix va dépendre de la taille du problème.

a. **L'approche un contre tous :** consiste à entraîner un SVM biclasse en utilisant les éléments d'une classe contre tous les autres. Il s'agit de résoudre c (avec, c représente le nombre des classes) problèmes SVM chacun de taille n.

b. **L'approche un contre un :** consiste à entrainer (c−1)/2 (avec, c représente le nombre des classes) SVM sur chacun des couples de classes, puis à décider la classe gagnante soit par un vote majoritaire soit en traitant les résultats grâce à l'estimation de probabilités à posteriori. Le nombre de classifieurs SVM à entrainer peut être réduit en utilisant un codage astucieux pour les classes à travers un code correcteur d'erreurs ou un graphe direct acyclique.

1.10.4 *Méthodes structurelles et syntaxiques*

Les méthodes de reconnaissance structurelles se basent sur la structure physique des mots ou des caractères. Elles décrivent les formes complexes par des primitives plus simples. Dans le cas de reconnaissance de l'écriture, les primitives sont de genre topologique et géométrique (traits, boucles, points, ...).

1.10.4.1Méthodes structurelles

Les méthodes de reconnaissance structurelle sont utilisées plus souvent dans la reconnaissance de caractères en ligne [117, 30] que dans la reconnaissance de caractères hors ligne. Les méthodes structurelles représentent une forme en tant que structure (chaîne, arbre ou graphique) de Taille flexible. Ces méthodes sont à base des arbres, graphes et chaînes :

Classification à base d'arbres et de graphes : dans les méthodes graphiques, les unités d'écriture (graphèmes, caractères, et mots) sont représentées par des arbres et des graphes dans la phase d'entraînement. Ces graphes permettent de décrire les caractéristiques des primitives de ces unités (traits, segments, points d'inflexion et de branchement…). Dans la phase de la

classification, les mesures de similitude d'arbres ou de graphes sont utilisées pour assigner les classes des unités d'écriture [118].

Classification à base de chaînes : dans ce cas, les unités d'écriture (graphèmes, caractères et mots) sont représentées par des chaînes de primitives. La méthode consiste à mesurer la similitude entre les chaînes des entités à reconnaître et un modèle de référence par un calcul de distance [119].

1.10.4.2 Méthodes syntaxiques

Les méthodes syntaxiques représentent chaque forme par une phrase dans un langage où le vocabulaire est constitué de primitives.

La reconnaissance syntaxique consiste alors à déterminer si la phrase de description de la forme peut être générée par la grammaire. Cette approche a été utilisée en reconnaissance de l'écriture persane [120].

L'absence des algorithmes d'apprentissage syntaxique efficaces permettant l'élaboration d'une grammaire à partir d'un ensemble fini de phrases, représente l'un des difficultés d'implémentation de cette approche.

1.10.4.3Post-traitements

La phase de post-traitement est la dernière phase implémentée dans les systèmes de reconnaissance, elle consiste à programmer un ensemble de techniques pour l'amélioration du taux de reconnaissance à travers le raffinement des décisions prises dans l'étape précédente et à confirmer l'hypothèse et les résultats obtenus par l'utilisation des méthodes grammaticales [45], lexicales, syntaxiques, sémantiques, pragmatiques, linguistiques. Cette étape peut être programmée pour corriger les erreurs de segmentation. L'utilisation d'un lexique ou d'un dictionnaire permet de valider aposteriori la reconnaissance effectuée, mais pour optimiser le temps de calcul, diverses techniques de pré-organisation et d'interrogation du lexique peuvent être utilisées [46], L'utilisation d'un modèle du langage permet de moduler le taux de confiance des hypothèses de mots reconnus [47].

1.11 Mesure de performance

Après la réalisation des tests sur le système développé, nous devons calculer la mesure de performance afin de s'assurer du degré d'efficacité. Plusieurs critères ont été proposés dans [44]. Parmi ces critères, on trouve :

- ✓ Le taux de reconnaissance : pourcentage de formes bien reconnues.

$$\textbf{Taux de reconnaissance} = \frac{Nombre\ des\ individus\ reconnus}{Taille\ d'échantillant}$$

- ✓ Le taux de confusion : pourcentage de formes pour lesquelles le système fait une erreur.

$$\textbf{Taux de confusion} = \frac{Nombre\ des\ individus\ \text{mal}\ reconnus}{Taille\ d'échantillant}$$

✓ Le taux de rejet : pourcentage de formes pour lesquels le système refuse de se prononcer.

$$\textbf{Taux de rejet} = \frac{Taille\ d'échantillant - (Nombre\ des\ individus(\ mal\ \ reconnus + reconnus))}{Taille\ d'échantillant}$$

✓ Le taux de confiance : pourcentage de formes bien reconnues par rapport à la somme des formes bien reconnues et des formes mal reconnues.

$$\textbf{Taux de confiance} = \frac{Nombre\ des\ individus\ \ reconnus}{(Nombre\ des\ individus\ (\text{mal}\ reconnus + reconnus))}$$

Pour juger la performance d'un système donné, nous devons comparer les résultats obtenus avec d'autres approches et d'autre système sous les mêmes conditions.

1.12 Conclusion

Dans ce chapitre, nous avons présenté les notions générales liées aux systèmes de reconnaissance de forme et particulièrement celle en relation avec l'écriture. Nous avons focalisé notre étude sur les principaux aspects de la reconnaissance de l'écriture, en touchant les déférentes parties constituant un système de reconnaissance de l'écriture imprimée ou manuscrit, à savoir : l'acquisition, le prétraitement, l'extraction des caractéristiques, la classification et le post-traitement.

Concernant les techniques de classification qui sont en nombre de quatre, elles sont les plus utilisées dans la littérature. Ces approches sont :

• La première approche repose sur l'utilisation des distances pour trouver les K voisins le plus proche (KPPV).

• La deuxième approche repose sur la séparation de données par la recherche d'un hyperplan optimal via les vecteurs à vaste marge(SVM).

• La troisième approche de classification présentée, repose sur l'approche connexionniste des perceptrons multicouches (PMC).

• La quatrième approche de classification repose sur les méthodes structurelles à base d'arbres, de graphes et chaîne. Ainsi, les méthodes syntaxiques des langages à base grammaticale d'un vocabulaire.

Dans les trois chapitres qui suivent, nous présenterons nos contributions dans le domaine de la reconnaissance en exploitent les perspectives de ces travaux cités.

Chapitre 2: Reconnaissance des caractères isolés Arabes imprimés

2.1 Introduction

Ce présent chapitre s'intéresse à la reconnaissance des caractères arabes, et imprimés. Au début, il cite les caractéristiques de l'écriture arabe ainsi que le principe de la segmentation adoptée pour le partitionnement d'un texte arabe en lignes, en mots et en caractères. Ensuite, il mentionne les problèmes rencontrés accompagnés des solutions proposées.

2.2 Segmentation d'un texte arabe imprimé

Dans cette section, nous exposons notre approche pour la segmentation du texte arabe imprimé en caractères (figure 2.1). En premier temps, l'image du texte soumise à une étape de prétraitement pour améliorer la qualité des informations présentées dans le document, pour les documents inclinés, nous proposons un module de traitement des problèmes d'inclinaison. Ensuite, le texte est segmenté en lignes, puis, chaque ligne du texte est segmentée en mots ou pseudo-mots, et finalement, on procède à la segmentation des mots en caractères ou graphèmes (partie d'un caractère). Une dernière étape de post-traitement tente de corriger les erreurs de segmentation commis par l'algorithme de segmentation suivant une approche syntaxique (vocabulaire) ou
structurelle.

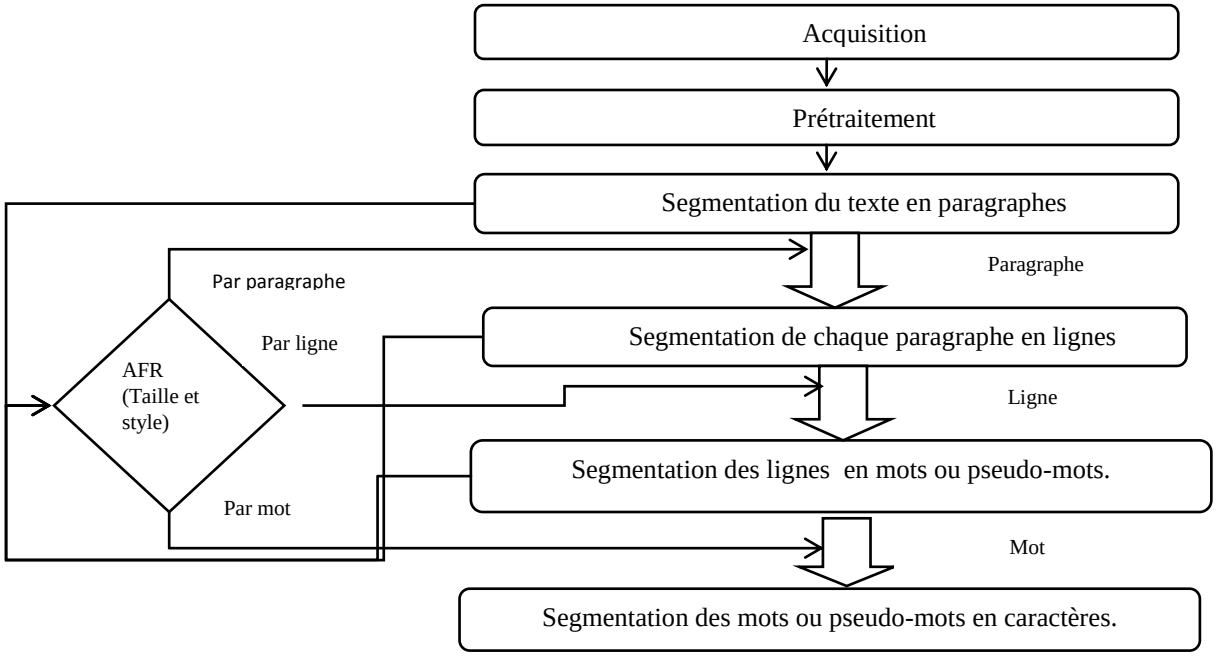


Figure 2-1 Processus de segmentation du texte en caractères.

Les algorithmes de segmentation de texte pouvant exploiter un module de reconnaissance de la fonte des paragraphes ou des lignes de texte ou mots, selon l'approche adoptée, pour améliorer le taux de segmentation.

2.2.1 *Caractéristiques de l'écriture arabe imprimée*

L'écriture arabe est classée parmi les écritures complexes, en effet la présence d'une ligne reliant les déférents caractères « la ligne de base » composant le mot ou le pseudo-mot. La nature cursive de l'écriture arabe rend le processus de segmentation du texte en caractères très délicat, en effet, la géométrie de la ligne de base (figure 2-2) variée en fonction de la fonte d'écriture, la taille et le style utilisé. Les points composants la ligne de base ne sont pas toujours rectilignes c.à.d.la ligne de base peut prendre deux formes ; segments ou courbures. Parmi les problèmes de segmentation du texte en caractères est la présence des chevauchements horizontaux et verticaux (figure 2-3).

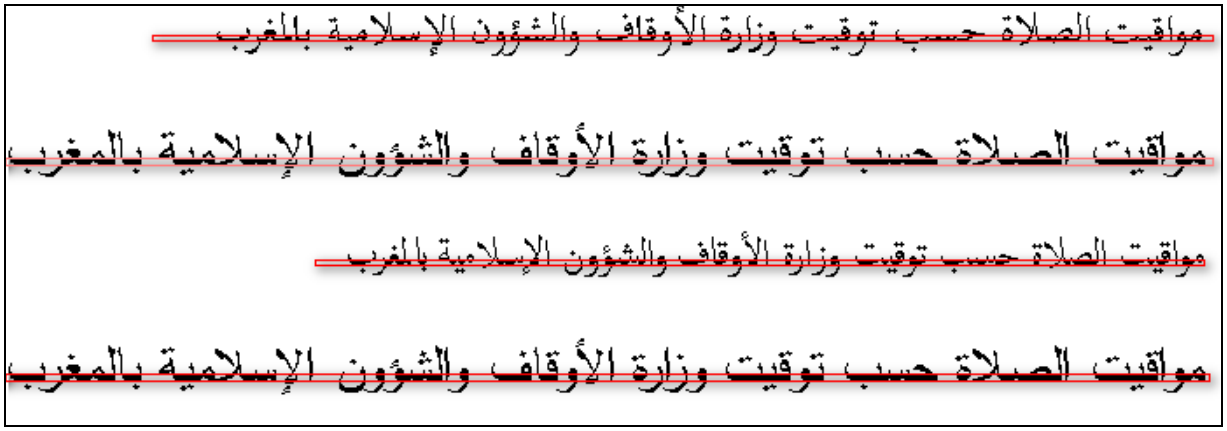


Figure 2-2 Exemple de ligne de base.

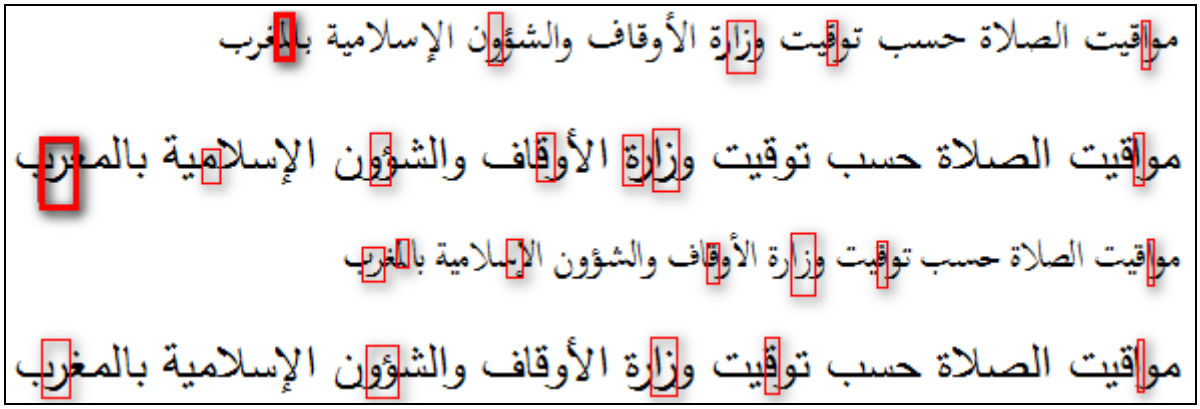


Figure 2-3 Exemple de chevauchements horizontaux et verticaux.

2.2.2 *Prétraitements*

Dans cette section, on met l'accent sur les méthodes de prétraitement que nous avons appliqué en vue de produire une version nettoyée d'image d'origine.

Les images des textes utilisés pour tester notre approche de segmentation, sont souvent entachées de différents types de bruit engendrés par différentes sources, qui peut réduit en choisissant un seuil de binarisation adéquat. L'intérêt des processus qui sont relatés dans ce qui suit, à savoir : binarisation, et correction d'inclinaison, peut être résumé essentiellement en

deux points : (1) réduction de la variabilité de l'écriture imprimée et (2) atténuation ou suppression des informations indésirables.

2.2.2.1 Binarisation

La binarisation est un cas particulier de seuillage, qui vise en principe à classer les pixels d'image traitée en deux classes (noire et blanche) : les pixels de premier plan associés au texte et les pixels de l'arrière-plan associés au fond. Les images que nous avons utilisées dans notre système peuvent être en différentes structures calligraphiques. À ce niveau, nous avons opté pour la méthode d'Otsu [121], qui consiste à déterminer un seuil unique globalement sur toute l'image. Ce seuil doit minimiser la variance intra-classe entre les pixels des deux classes précédentes (noire/blanche) (figure 2-4).

(a) (b)

Figure 2-4 (a) image avant la binarisation, (b) image binarisée

2.2.2.2 Correction d'inclinaison

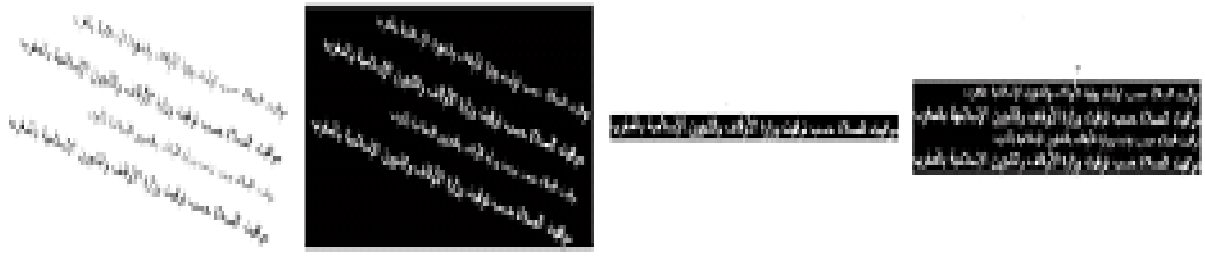
Le texte contenu dans le document, peut être incliné à cause d'un mauvais positionnement du document au moment de la numérisation. Une ligne du texte est inclinée si elle présente une distorsion de ces lignes par rapport l'axe horizontal. Les techniques utilisées de corrections d'inclinaison et déformation visent à ramener et mettre horizontalement chaque ligne de l'écriture inclinée.

Les algorithmes de correction d'inclinaison reposent tous sur le principe suivant : détection et estimation de l'angle d'inclinaison puis la correction d'inclinaison à l'aide d'une rotation isométrique par rapport à l'angle calculé.

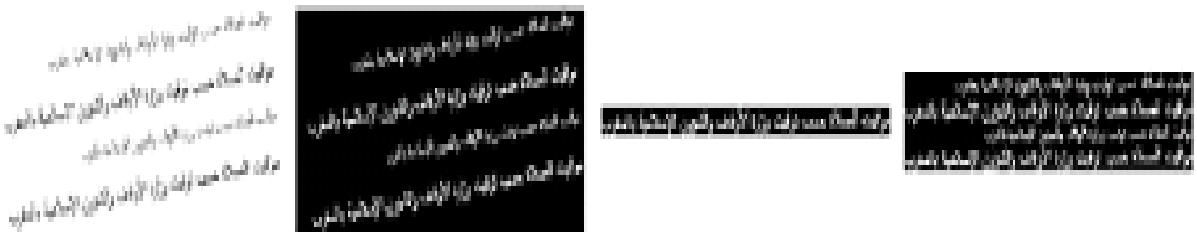
Dans notre système, nous avons utilisé la technique basée sur les histogrammes de projection horizontale [122]. Pour calculer l'angle d'inclinaison de la ligne ou paragraphe, nous avons calculé le nombre des pixels blancs selon différentes orientations qui sont proches de l'horizontale pour chaque ligne, pour chaque orientation l'histogramme présente une valeur

maximale, la direction la plus probable est celle qui maximise l'entropie. L'histogramme d'entropie maximale est celui dont les extremums sont les plus marqués. L'angle d'inclinaison de la ligne α est celui qui correspond à l'histogramme d'entropie maximale. Pour corriger cette inclinaison, il suffit d'appliquer une rotation isométrique d'image d'angle α.

La Figure ci-dessous illustre la correction d'inclinaison des lignes du texte arabe incliné à l'aide de la technique de l'histogramme de projection horizontale.



(a)



(b)

Figure 2-5 Exemple de correction d'inclinaison gauche (a) et droite (b) de texte arabe.

2.2.3 *Segmentation du texte en paragraphes*

Un paragraphe peut être composé d'une ou plusieurs lignes, après la localisation du texte, nous avons calculé les espaces interligne pour déterminer et distinguer entre les espaces inter-paragraphe et interlignes (figure 2-6 et 2-7).

Pour localiser le texte contenu dans l'image acquise, il suffit de calculer les projections horizontale et verticale d'histogramme. La localisation des paragraphes du texte, est réalisée en calculant les interlignes de toutes les lignes du texte pour déterminer l'inter-paragraphe.

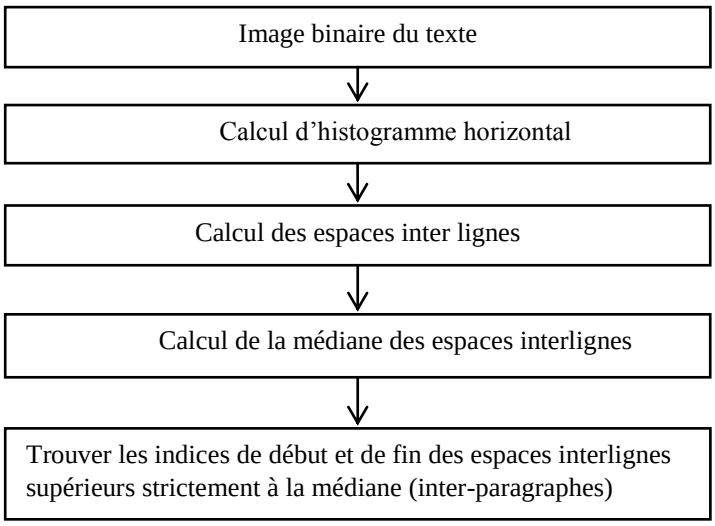


Figure 2-6 Processus de segmentation du texte en paragraphes.

Figure 2-7 Histogrammes de projections horizontales du texte.

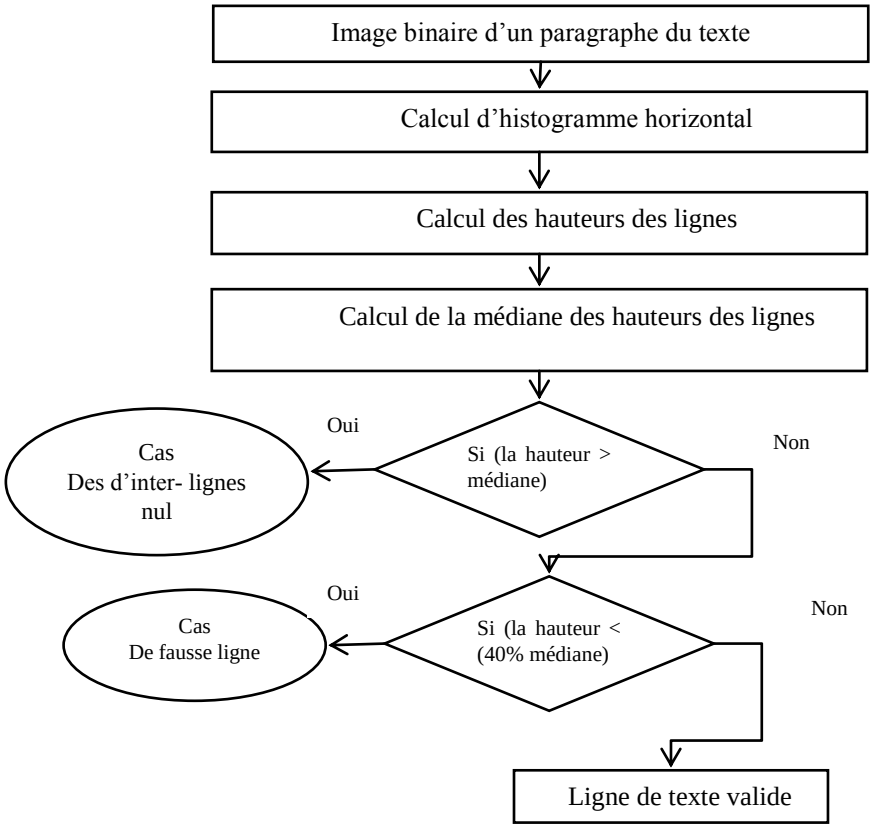
2.2.4 *Segmentation du paragraphe en lignes*

Le texte peut être composé d'une ou plusieurs paragraphes, dans le même paragraphe l'interligne est toujours le même, la segmentation du texte en lignes consiste à découper le paragraphe du texte en lignes. Dans cette étape, les lignes du texte sont extraites en utilisant l'histogramme horizontal (figure 2-8). En exploitant l'espace existant entre deux lignes en pixels noirs qui correspondent à un interligne. Pour résoudre le problème des fausses lignes de texte, nous estimons la valeur d'espace entre deux lignes suivant les interlignes du paragraphe.

Les points diacritiques au-dessus ou en dessous de la ligne de la base provoquent le problème de fausse ligne du texte qui est directement relié à la ligne la plus proche, ce processus traite le cas des lignes très approchées où l'interligne peut correspondre à un espace d'interligne nul ou correspond à un seul pixel.

Dans le cas des lignes approchées, la médiane des interlignes du même paragraphe est calculée pour déterminer les coordonnées de segmentation du bloc des lignes approchées en lignes.

Figure 2-8 Projections horizontale d'histogramme du texte



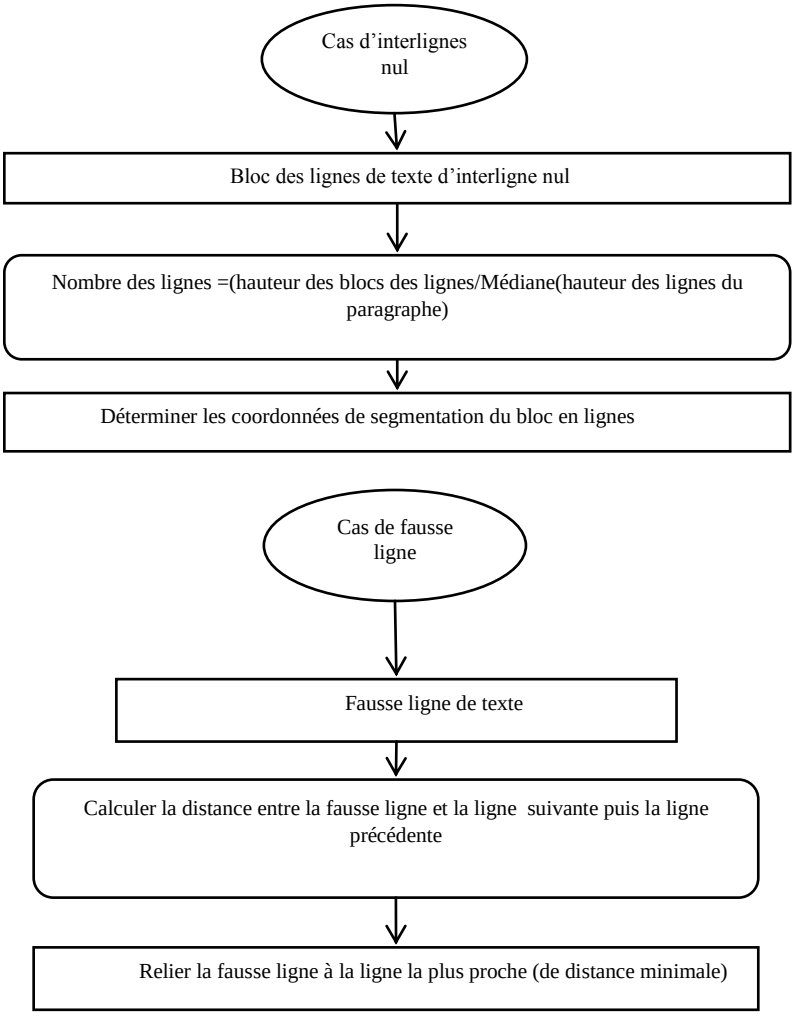


Figure 2-9 Processus de segmentation du paragraphe en lignes.

La ligne de base représente l'ensemble des points avec lesquelles les caractères des mots et pseudo-mot sont connectés. La ligne de base prend plusieurs structures qui peuvent être sous forme d'une courbure ou segment, la structure générale est déterminée suivant l'aspect calligraphique et morphologique de la famille de fonte.

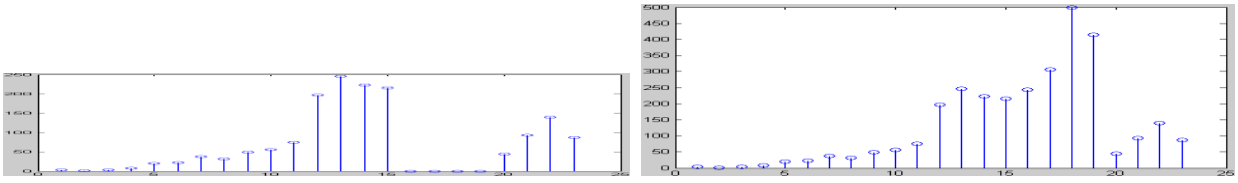
La ligne de base est détectée en utilisant l'histogramme horizontal. La ligne de base représente l'amplitude maximale de l'histogramme et d'épaisseur relativement variable en fonction de la famille de fonte, la taille et le style d'écriture (figure 2-10).

Pour estimer la position de la ligne de base, nous devons calculer son épaisseur.

Figure 2-10 Lignes du texte et ces histogrammes de projections horizontales montrant la ligne de base.

L'amplitude maximale d'histogramme horizontal de la ligne du texte est calculée, en analysant les valeurs d'histogramme vertical, au-dessus et au-dessous, supérieur ou égal à un taux (expérimentalement 60 % pour la Famille de fonte Tahoma) de l'amplitude maximale.

La suppression des amplitudes trouvée correspond à la suppression de la ligne de base figure 2.11.



(b) (a)

Figure 2-11 Histogramme horizontal d'une ligne avant (a) et après (b) la suppression de la ligne de base

La suppression de la ligne de base n'est pas toujours possible, en effet, certaine fonte présente une ligne de base de structure courbure où le pic d'histogramme ne correspond pas à la ligne de base (figure 2-12)

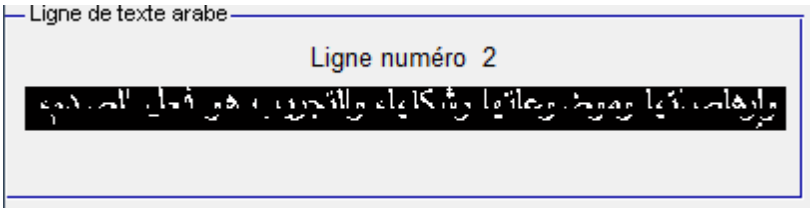


Figure 2-12 Résultat de suppression de la ligne de base d'une ligne

Les résultats obtenus de la suppression de la ligne de base dans le cas d'une ligne sont meilleurs par rapport au cas d'un mot ou pseudo-mot (figure 2-12, 2-13, 2-14 et 2-15).

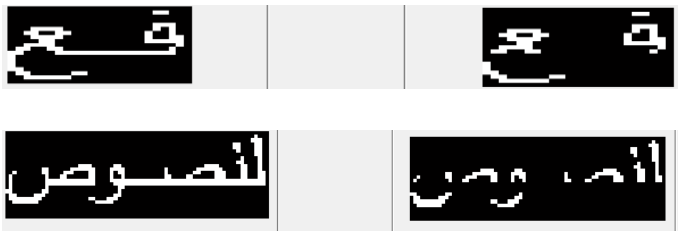


Figure 2-13 Résultat de suppression de la ligne de base d'un mot.

Figure 2-14 Résultat de suppression de la ligne de base d'une ligne du texte arabe.

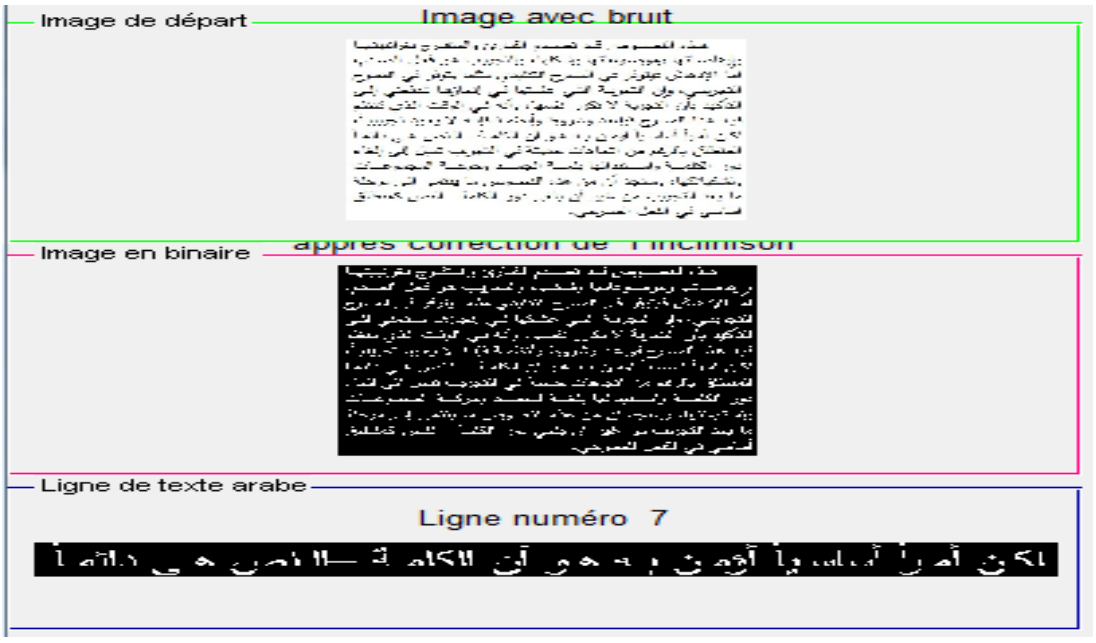
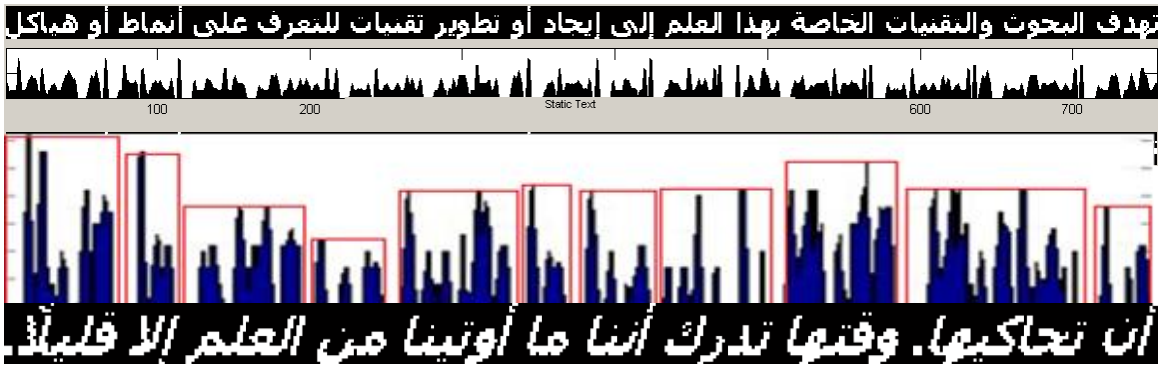


Figure 2-15 Résultat de suppression de la ligne de base d'une ligne du texte arabe avec la méthode de calcul d'épaisseur d'écriture.

2.2.5 *Segmentation des lignes du texte en mots*

Après le découpage du texte en lignes de l'étape précédente, nous avons obtenu des lignes séparées sous forme des images. Par conséquent, nous allons chercher à segmenter chaque ligne du texte en mots.

Dans le cadre de ce travail, nous avons utilisé la technique d'histogramme vertical pour segmenter chaque ligne du texte en mots, en exploitant les espaces séparant les mots entre eux (figure 2-16). La nature cursive de l'écriture arabe complique le processus de la segmentation du texte en mot ou pseudo-mots, en effet les chevauchements horizontaux et accolement des caractères des extrémités entre pseudo-mots restent parmi les problèmes major de la segmentation d'une ligne du texte en mots.



2.2.6 *Segmentation des mots en caractères*

Pour déterminer les points de segmentation des mots en caractères, l'image de chaque ligne du texte est exploitée, en effet, la ligne de base de chaque ligne du texte est supprimée, un histogramme vertical est établi sur la nouvelle image. Les espaces blancs trouvés dans l'histogramme contiennent les points préliminaires de segmentation (le point de segmentation correspond généralement à la fin de l'espace blanc).

Après la suppression de la ligne de base de l'étape précédente, nous avons obtenu des lignes séparées sans ligne de base sous forme des images. En conséquence, nous avons segmenté chaque ligne du texte en mots et chaque mot en caractères en se basant sur l'histogramme vertical.

Les Figures 2-17 et 2-18 ci-dessous présentes des exemples de segmentation en caractères.

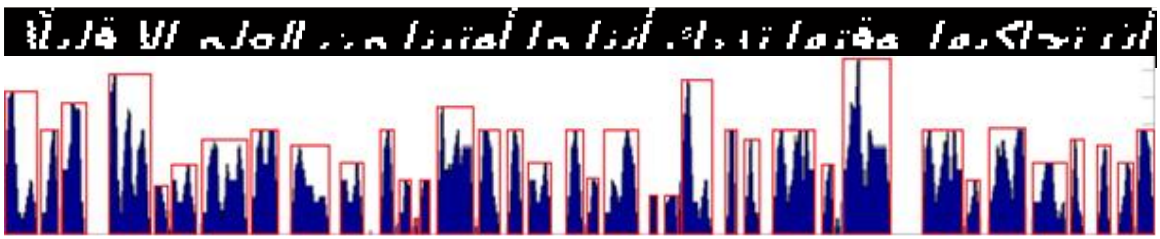


Figure 2-17 Ligne du texte sans ligne de base et son histogramme vertical pour le segmenter en caractères.

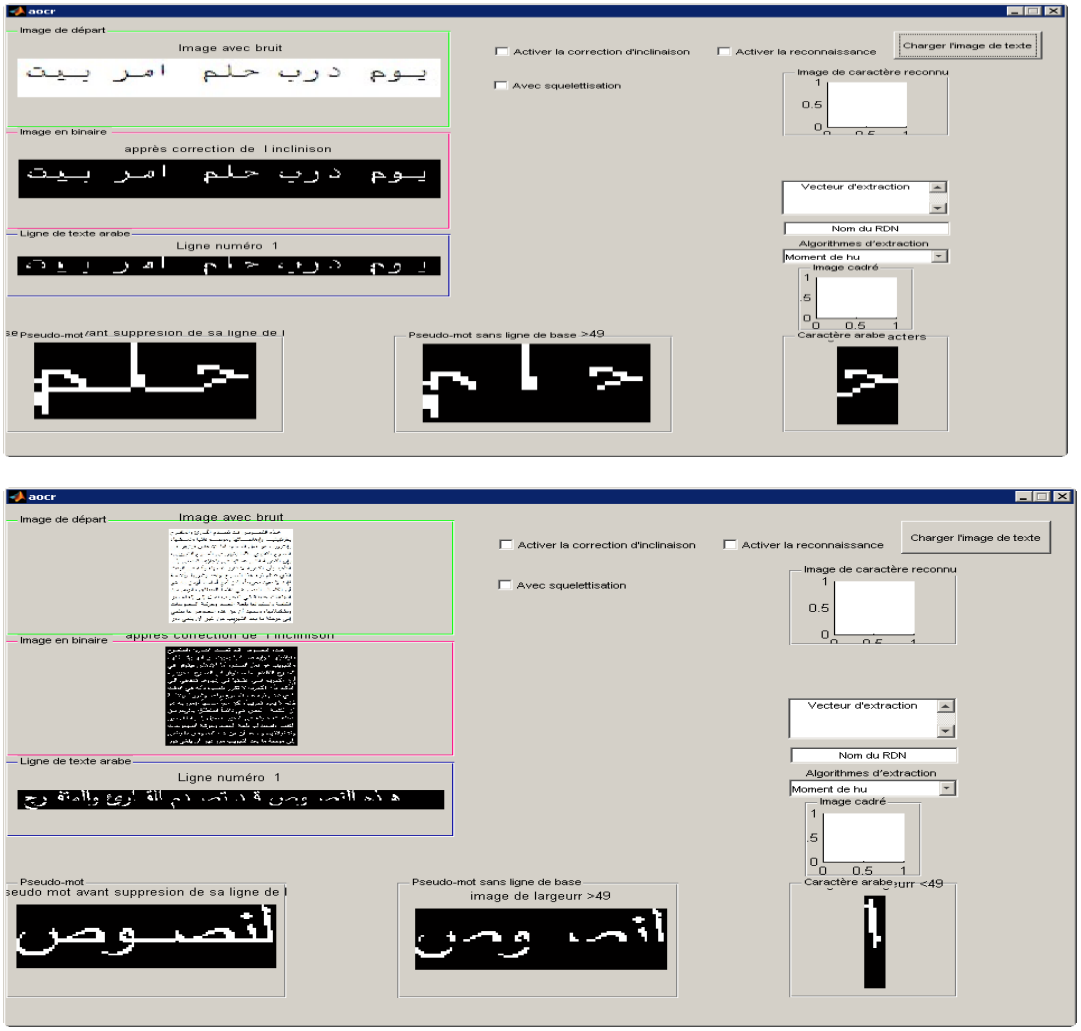


Figure 2-18 Exemple de segmentation d'un mot en caractères.

2.2.7 *Résultats expérimentaux*

Dans les tableaux suivant (2-1, 2-2 et 2-3), on présente les résultats expérimentaux de la segmentation du texte sur trois familles de fontes et quatre tailles et styles différents.

Familles de fonte	Tailles	Styles	Nombre des lignes	Taux de segmentation du texte en lignes
Segoe_UI	10, 12, 14,16	Gras, Italique, Italique_Gras, Simple	3501	100%
Andalus	10, 12, 14,16	Gras, Italique, Italique_Gras, Simple	2498	100%
Tahoma	10, 12, 14,16	Gras, Italique, Italique_Gras, Simple	3397	100%

Tableau 2-1 Taux de segmentation du texte arabe en lignes

Le tableau ci-dessus présente les résultats expérimentaux obtenus dans le cas de la méthode de projection horizontale et les techniques proposées permetant de traiter le cas des

lignes très approchées ou d'interligne nul et le cas des fausses lignes pour la segmentation du texte en paragraphes puis en lignes.

Les résultats obtenus illustrent la bonne performance du système proposé pour la segmentation du texte en lignes appliqués sur trois familles de fonte de déférentes tailles et quatre styles.

Famille de fonte	Taille	styles	Taux de segmentation des lignes en mots
Segoe_UI	10, 12, 14,16	Gras, Italique, Italique_Gras, Simple	97 %
Andalus	10, 12, 14,16	Gras, Italique, Italique_Gras, Simple	98 %
Tahoma	10, 12, 14,16	Gras, Italique, Italique_Gras, Simple	99 %

Tableau 2-2 Taux de segmentation des lignes du texte arabe en mots ou pseudo-mots.

L'approche adoptée dans notre système de segmentation des lignes en mots, repose sur le calcul des espaces séparant les mots et pseudo-mot dans la même ligne afin de déterminer les points de segmentation, en calculant la médiane des espaces inter-mot. Les résultats expérimentaux obtenus (tableau 2-2) sont très encourageant, les erreurs de segmentation sont dues à la présence des ponctuations et d'autres signes qui ne font pas partie du mot.

Famille de fonte	Taille	Styles	Taux de segmentation des mots en caractères
Segoe_UI	10, 12, 14,16	Gras, Italique, Italique_Gras, Simple	80 %
Andalus	10, 12, 14,16	Gras, Italique, Italique Gras, Simple	85 %
Tahoma	10, 12, 14,16	Gras, Italique, Italique Gras, Simple	97 %

Tableau 2-3 Taux de segmentation des mots en caractères ou graphèmes.

Les résultats expérimentaux concernant la segmentation des mots en caractères présentés dans le tableau ci-dessus varient d'une fonte à une autre, cela est dû à l'utilisation d'un seul algorithme qui n'est pas spécialisé pour une fonte bien déterminée, mais, il applique le même processus sur différentes fontes de différente structure. Les résultats expérimentaux pouvant être amélioré en adoptant des algorithmes spécialisés pour chaque fonte après la reconnaissance préalable de la fonte du texte à traiter.

2.2.8 *Analyse des résultats et discussions*

- **Analyse des résultats expérimentaux**

Le choix des paramètres, approche de prétraitements, la qualité d'image d'origine et la nature de l'écriture sont des facteurs influents directement sur les résultats expérimentaux présentés ci-dessus.

- **Segmentation du texte en lignes :** l'approche de segmentation du texte en lignes a prouvé des résultats très importants sur les échantillons testés, surtout pour le cas des lignes très approchées qui est rarement traité par les algorithmes de segmentation affichés dans la littérature.

- **Segmentation des lignes du texte en mots :** l'approche présente des résultats variant en fonction des caractéristiques calligraphiques, topologique et structurelle de la famille de fonte, l'approche adoptée est basée sur les composantes connectées de l'histogramme horizontal, les résultats expérimentaux obtenus sont en général encourageant.

- **Segmentation des mots en caractères :** les chevauchements horizontal et vertical de certains parties ou caractères composant le mot ou pseudo-mot, la qualité des images des mots sont parmi les problèmes empêchant la segmentation correcte de certains mots en caractères. En revanche, l'approche présente des résultats très encourageants.

Cette expérience nous a montré que l'utilisation d'un seul algorithme de segmentation pour segmenter tous les types des textes arabes, composés de différentes fontes, ne donne pas toujours des résultats fiables pour toutes les fontes, cependant, il présente toujours des problèmes particuliers liés à la nature de l'écriture dépend de la famille de fonte, la taille et le style.

La proposition de plusieurs méthodes de segmentation dont chacune spécialisée pour une fonte bien déterminée peut améliorer la performance générale du système, dans ce cas, nous nous sommes besoin d'un modèle pour identifier la fonte pour mieux choisir l'algorithme de segmentation convenable. Pour améliorer le taux de segmentation du texte en lignes, la famille de fonte et la taille du texte doivent être reconnues en premier temps pour déterminer l'interligne du texte et l'épaisseur du trait..

2.3 Reconnaissance des caractères isolés Arabes

2.3.1 *Introduction*

La reconnaissance des caractères arabes a bénéficié de plusieurs participations de recherche depuis 1970 [123], cette branche de recherche est basée en général sur l'approche analytique, en proposant des systèmes de reconnaissance de tous les types de caractères arabes, comme des contributions dans le domaine. Actuellement, il n'y a pas un système qui peut être considéré comme fiable pour la reconnaissance d'écriture arabe.

Dans cette étude, on propose un système pour reconnaître les caractères imprimés et les caractères d'une seule famille de fonte d'écriture arabe imprimée avec une nouvelle technique d'extraction de primitives. Afin d'améliorer le taux de reconnaissance, le système propose trois étapes pour le processus de reconnaissance des caractères isolés :

Dans la première étape, l'image est scannée et transformée en image binaire, le caractère est localisé en supprimant les parties additionnelles afin de la redimensionner en une image de 100×100 pixels, ensuite, deux opérations morphologiques sont appliquées; la première pour trouver le squelette du caractère et la deuxième pour boucher les trous dans les composantes connexes. Concernant la deuxième étape et pour extraire les primitives de chaque caractère, on utilise un algorithme nommé Cadre de Niveau, cet algorithme, divise la matrice binaire d'image en 100 zones dont chacune représente une matrice carrée d'ordre 10. Chaque zone est considérée en 5 niveaux, la densité des pixels est calculée pour chaque niveau pour obtenir 5 valeurs, la moyenne de ces 5 valeurs est calculée pour caractériser une zone, le même processus s'applique sur les autres zones afin d'obtenir un vecteur de 100 primitives représentant l'image du caractère arabe, l'approche proposée est appliquée sur 105 classes de caractères. Les échantillons de test sont classifiés par la méthode de K plus proche voisin en utilisant la distance de Spearman.

2.3.2 *Caractéristiques et problèmes des caractères arabes imprimés [124]*

Les caractéristiques des caractères arabes imprimés sont nombreuses, à savoir :

- ✓ Le texte arabe est écrit de droite à gauche et il est cursif.
- ✓ L'écriture arabe se base sur 28 caractères, mais, en écriture cursive connectés entre eux avec la ligne de base. Chaque caractère peut prendre 2,3 ou 4 formes différentes, et certains caractères isolés peuvent former des ligatures distinctes (par exemple ئ ,لا ,ؤ), donc au total il existe plus de 105 caractères différents.
- ✓ Il n'existe pas de concept de caractères majuscules ou minuscules.
- ✓ La même écriture arabe imprimée peut utiliser plus d'une seule fonte avec tailles et styles différents.
- ✓ Similitude de certains caractères comme : ب, ت, ث. ,ف, ق ,ح, خ, ع, غ
- ✓ Le bruit, le fond d'image, le chevauchement de l'écriture, et la forme des caractères nous donnent une petite quantité d'informations pour distinguer tous les caractères les uns des autres : par conséquent, nous trouverons certaines difficultés dans l'étape de classification.

Les caractères arabes sont normalement écrits de droite à gauche et verticalement de haut en bas, La figure 2.19 illustre les caractères arabes imprimés.

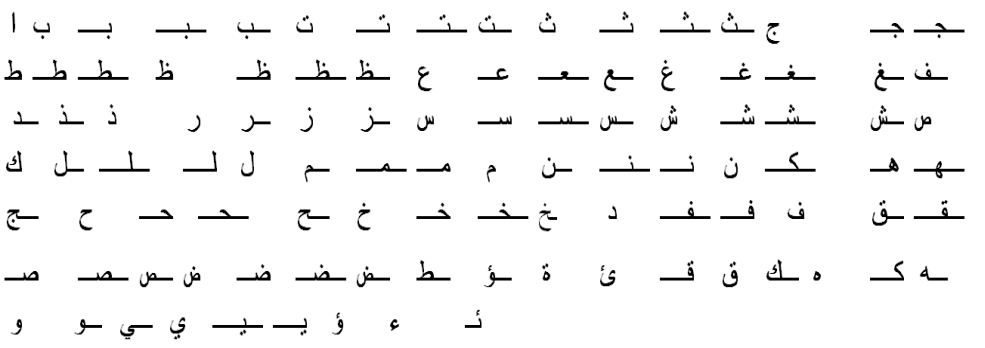


Figure 2-19 Caractères arabes imprimés de différente forme.

2.3.3 *Système de reconnaissance*

La figure (2.20) présente les phases implémentées dans le système proposé pour la reconnaissance des caractères arabes imprimés. Dans ce qui suit, nous détaillons chaque étape.

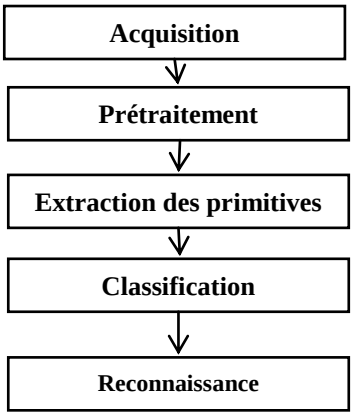


Figure 2-20 Étapes du système de reconnaissance

2.3.4 *Prétraitements*

L'étape de prétraitement prend la matrice d'image acquise, puis elle l'applique les opérations suivantes.

2.3.4.1 Binarisation et réduction de bruits

La matrice de l'image en couleurs ou gris est convertie en une image binaire en utilisant la technique de seuillage, suivi d'une opération de binarisation qui permet de convertir les pixels en deux valeurs 0 ou 1 selon le seuil fixé en adoptant la technique de seuillage globale. Pour la réduction de bruit dans les images utilisées, nous avons utilisé des techniques d'opérations morphologiques pour connecter les pixels non connectés, supprimer les pixels isolés (figure 2-21).

(a) (b)

Figure 2-21 (a) Image avant la binarisation, (b) Image après la binarization et la réduction du bruit.

2.3.4.2 Normalisation de taille

L'image du caractère est normalisée à une taille de 100×100 pour un premier test et 80×80 pour un deuxième test en utilisant la méthode plus proche présentée dans le chapitre 1 (Figure 2-22).

(a) (b)

Figure 2-22 (a) Image avant la normalisation, (b) Image après la normalisation de taille.

2.3.4.3 Calcul de Squelette

Pour réduire au minimum l'influence et la variation de trait d'écriture, l'image est normalisée puis le squelette du caractère est calculé à l'aide d'un algorithme de squelettisation (cf. chapitre 1) de [125], pour définir le squelette de forme pour la localisation d'un caractère, et l'utiliser pour extraire des caractéristiques en utilisant l'approche proposée appelée « Cadre de Niveau » (Figure 2-23).

(a) (b) (c)

Figure 2-23 (a) Image de caractère, (b) Squelette d'image,(c) Caractère localisé.

2.3.5 *Extraction de caractéristiques*

L'extraction des caractéristiques est une étape sensible qui influe directement sur la performance du système de reconnaissance. Pour fournir une quantité d'informations et de maximiser la possibilité de distinguer chaque caractère ou d'un de l'autre, nous avons proposé une nouvelle approche d'extraction pour calculer 100 paramètres caractéristiques qui ont été obtenus à partir de la forme et la distribution des pixels dans l'image. Pour cela, nous avons proposé une nouvelle approche nommée Cadre de Niveau. Cette technique pour la divise l'image du caractère en 100 zones, chacune représente une matrice d'ordre 10, chaque zone est utilisée pour calculer des valeurs statistiques comprises entre 1 et 0 comme suit :

Chaque matrice d'une zone est patronnée en 5 cadres, chacun représente un niveau (figure 2-24).

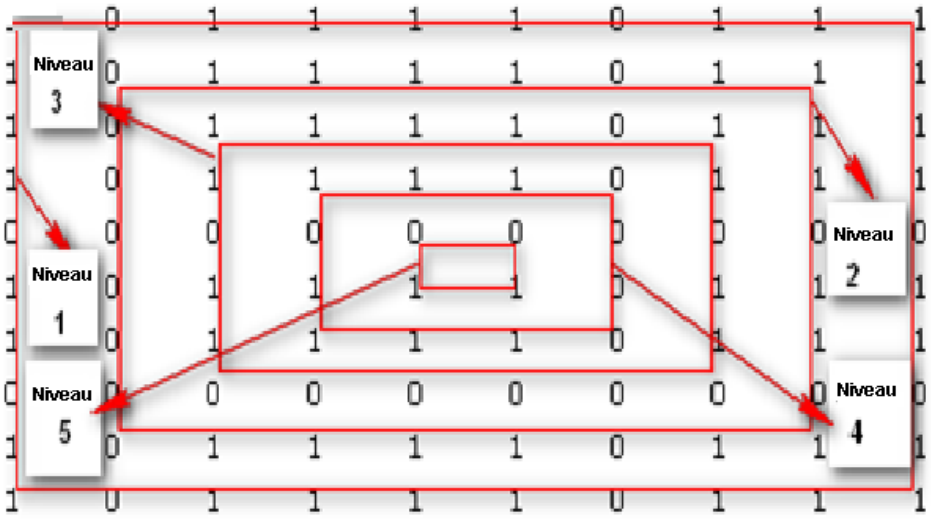


Figure 2-24 Exemple de Cadre de Niveau pour une zone.

Pour chaque niveau, le calcul suivant est effectué afin d'extraire les densités suivantes.

r1 = Densités de pixels dans la ligne supérieure.
r2 = Densités de pixels dans la ligne inférieure.
cl = Densité de pixels dans la colonne de gauche.
cr = Densité de pixels dans la colonne de droite.
Tel que :

Et $r = (r1+r2)/2$ (2-1)

Et $c = (cl+cr)/2$ (2-2)

Et $L_i = (r+c)/2$ (2-3)

L_i Représente la valeur caractérisant le niveau i.

Les valeurs caractéristiques des cinq niveaux ont été divisées par 5 et le résultat représente une primitive pour une zone. Finalement, nous avons obtenu 100 caractéristiques pour représenter chaque caractère arabe.

Pour étudier l'influence de changement de taille, nous avons applique le même processus sur les images des caractères arabes normalisés en une matrice carrée d'ordre 64, dans ce cas le vecteur caractéristique est de taille 64

2.3.6 *Classification*

Dans la phase de classification, pour classifier les échantillons de données qui représentent les caractères arabes imprimés, nous avons utilisé l'algorithme de K- plus-proche-voisin avec trois distances distinctes, Citybloc, Spearman et de corrélation, le nombre de voisin k est fixé à 1.

La technique du coefficient de corrélation 2D calcule le coefficient de corrélation entre A et B, où A et B sont les matrices ou vecteurs de même taille.

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right)\left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}} \quad (2.4)$$

Où :

$\bar{A}$: La moyenne des valeurs de la matrice A.

$\bar{B}$: La moyenne des valeurs de la matrice B.

2.3.7 *Résultats expérimentaux*

Dans cette section, nous présentons les résultats expérimentaux obtenus en utilisant deux méthodes d'extraction; la méthode proposée « Cadre de Niveau » et les sept premiers Moments de Hu [81] » appliqués sur les caractères arabes imprimés pour évaluer le système proposé.

2.3.7.1 Résultats expérimentaux concernant la reconnaissance de caractères et arabes imprimés

Afin d'évaluer notre approche, nous avons adopté la famille de fonte « Simplifié Arabe Fixed » pour les caractères arabes imprimés (figure 2-25). L'approche proposée est nommée « Cadre de Niveau » a été appliquée sur les images binaires du squelette pour extraire 100 primitives en premier test et 64 caractéristiques pour un deuxième test. La méthode de classification de K-Plus-Proche -voisin a été appliquée pour identifier la classe d'appartenance des caractères. Nous avons analysé 17, 850 images de caractères en utilisant 16,590 images pour l'apprentissage et 1.260 pour le test.

Pour comparer les résultats obtenus, nous avons utilisé trois distances distinctes, Citybloc, Spearman et la distance de corrélation.

❖ **Analyse des résultats de reconnaissance de caractères arabes imprimés**

Le tableau 2-4 présente les résultats expérimentaux obtenus de la reconnaissance des caractères arabes imprimés, en utilisant trois distances : la distance de Spearman, la distance de Corrélation et la distance de Citybloc. Avec la discrimination des caractères arabes imprimés avec 64 et 100 paramètres.

Méthode d'extraction	Distance	64 primitives	100 primitives
« Cadre de Niveau » appliqué sur le squelette	Spearman	96.82%	98.65%
	Correlation	87.69%	96.03%
	Citybloc	91.58%	96.98 %
« Sept Moments de Hu » appliqué sans squelette	Spearman	50.68%	53.12%
	Correlation	50.52%	50.40%
	Citybloc	50.76%	54.64%
« Sept Moments de Hu » appliqué sur le squelette	Spearman	51.40%	54.48%
	Correlation	50.12%	50.80%
	Citybloc	50.84%	54.72%

Tableau 2-4 Résultats de reconnaissance des caractères arabes imprimés avec 1-PPV

- ✓ **Cas d'utilisation de 100 primitives :**

Le taux de reconnaissance varie d'une distance à une autre, plus de 2,62 % de différence pour la méthode « Cadre de Niveau », ce qui explique qu'on peut exploiter différentes distances pour l'évaluation de l'approche d'extraction, et aussi la comparaison des résultats obtenus pour opter pour le meilleur choix ou le plus optimum au sens des distances.

Le taux maximal obtenu est 98.65 % en utilisant la distance de Spearman par contre le minimum taux de reconnaissance est obtenu avec la distance de Correlation 96.03%.

- ✓ **Cas d'utilisation de 64 primitives :**

Le taux de reconnaissance varie de 96.82% à 87.69% avec une différence de 9,13% ; le taux maximal est obtenu avec la distance de Spearman 96.82%, par contre avec la distance de Correlation, nous avons obtenu un taux de reconnaissance minimal de 87.69%.

- ✓ **Choix de l'approche.**

Finalement, on peut conclure que la caractérisation des caractères arabes imprimés en utilisant 100 primitives est meilleure que d'utiliser 64 paramètres caractéristiques.

Dans ce travail, nous avons considéré une famille de fonte sans faire l'identification de la taille et le style des caractères, pour cette raison on prévoit que la reconnaissance préalable de la famille de fonte, la taille et le style peut améliorer significativement la performance du système proposé.

2.4 Conclusion

Dans cette étude, nous avons proposé un nouvel algorithme d'extraction des primitives, nommé « Cadre de Niveau » pour élaborer un nouveau système de reconnaissance des caractères arabes imprimés hors-ligne. Deux approches sont adoptées en utilisant 64 primitives dans une première expérience et 100 paramètres caractéristiques pour une deuxième expérience pour créer une base de références et de test.

L'approche proposée d'extraction a été comparée avec la méthode de Hu, en calculant les sept premiers Moments de Hu, la méthode de classification de K-plus-proche-voisin (KPPV) est utilisée avec trois distances différentes, CityBloc, Spearman, et Corrélation. Les résultats expérimentaux montrent que les 100 caractéristiques donnent un meilleur taux de reconnaissance que les 64 caractéristiques pour les trois distances utilisées, pour les caractères arabes imprimés.

Les résultats expérimentaux montrent que la distance de Spearman donne le meilleur taux de reconnaissance des caractères de 98,65% pour les 100 caractéristiques et 96,82% pour les 64 caractéristiques.

Cette étude a prouvé que le domaine de la reconnaissance des caractères imprimés nécessite plus de précisions dans toutes les étapes pour perfectionner et généraliser la performance des systèmes de reconnaissance. Parmi les améliorations pouvant être apportées à ce travail est l'utilisation d'un sous-système de reconnaissance de fonte.

Chapitre 3: Reconnaissance des noms imprimés des villes et villages du Maroc

3.1 Introduction

La reconnaissance des mots arabe est une approche de reconnaissance de l'écriture adoptant l'approche globale qui considère le mot complet ou une partie du mot (pseudo-mot) comme unité à reconnaître, après la segmentation du texte en mots ou pseudo-mot [33,126]. Plusieurs contributions et approches d'extractions des primitives ont été proposées dans [127, 128, 129, 130, 131, 132], pour la reconnaissance des mots arabe, ces contributions donnent des résultats encourageants vis-à-vis d'autre approche analytiques, du fait que la reconnaissance des mots ne souffre pas de plusieurs problèmes qui connaissent d'autres approches, par exemple la forte similarité entre les formes à reconnaître.

Dans ce chapitre, nous allons présenter un système de reconnaissance des mots arabes, que nous avons appliqués sur les noms imprimés des villes et villages du Maroc composés de dix fontes distinctes et onze tailles différentes. Dans ce système, nous avons adopté une nouvelle approche de discrimination nommée "Zigzag de Poids de Densité (ZPD)". Le système proposé se compose de quatre étapes : acquisition, prétraitement, extraction, et classification. Dans un premier temps, l'image représentant le mot est enregistrée sous forme d'une image RGB dans l'étape suivante, l'image est transformée en image binaire en utilisant la technique de seuillage globale, puis sa taille est normalisée en 96×96 en utilisant l'approche plus proches voisins de normalisation de taille, les pixels qui ne font pas partie ou existant loin de la zone du nom sont supprimés, après l'image résultante est normalisée une autrefois en 96 lignes et 96 colonnes de pixels. La technique de Zigzag de poids de densité est utilisée pour extraire 144 paramètres caractérisant chaque nom dans la base de test ou d'entraînement. Dans l'étape de classification, nous avons créé 16000 exemples pour l'apprentissage et 6000 images des noms pour le test. Pour tester la performance de notre approche, nous avons utilisé les deux classificateurs K plus proches voisins avec la règle consensus et SVM (Support Vecteur Machine).

Ce chapitre est présenté comme suite : dans la section (3-2), nous allons présenter l'architecture du système de reconnaissance des mots proposé. Dans la section (3-3) nous allons détailler la stratégie proposée pour décrire les paramètres caractéristiques des noms utilisés pour le test et l'apprentissage. Dans les sections (3-4 et 3-5), nous présentons la base de données et les techniques de classifications utilisées. Dans la section (3-6) nous présenterons les configurations des classificateurs des mots arabes imprimés à l'aide de Séparateur à Vaste Marge et les k-plus-proches-voisins. Nous verrons aussi sous quelles conditions on peut avoir des meilleurs résultats en conclusion.

3.2 Architecture du Système proposé

L'architecture du système proposé, correspond à l'approche globale, qui considère le mot comme unité à reconnaître. Pratiquement le système ne traite pas la phase de segmentation, en revanche, il considère que le texte est déjà segmenté en mots.

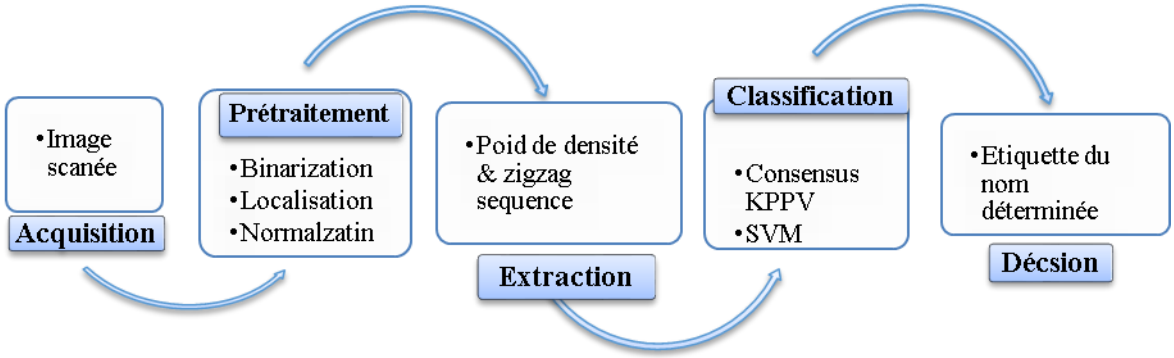


Figure 3-1 Architecture du système de reconnaissance

La (figure 3-1) illustre les différentes phases du système, dans la phase d'acquisition, l'image du mot est scannée dans un premier temps et enregistrée sous forme d'une image RGB, dans la phase du prétraitement l'image est transformée en image binaire puis normalisée à une taille prédéfinie, la position du mot dans l'image est localisée et les parties inutiles entourant le mot sont supprimées. Les paramètres caractérisant l'image résultant sont calculés avec l'algorithme proposé dans la phase d'extraction. Les deux classificateurs KPPV et SVM sont utilisés pour classifier les mots et prédire les classes d'appartenances en sortie.

3.3 Présentation des noms des villes marocaines utilisées

Le présent travail est élaboré en se basant sur 200 noms des villes et villages du Maroc (tableau 3-1), les mots sont composés de différents nombres de pseudo-mots : un, deux, trois, quatre, cinq, six, sept, et neuf pseudo-mots comme montre le tableau (3-2).

أرفود	تافيلالت	أسفي	أسول	أفورار	أكلموس	أمزميز	إفران	اسا الزاك	أغمات
الحاجب	الجزر الجعفرية	الجديدة	امديز	فاس	مكناس	الرباط	أكادير	إيموزار كندر	الدار البيضاء
العيون	الصخيرات	السمارة	العرائش	الصويرة	الريصاني	الراشيدية	الداخلة	ايت باعمران	الحسيمة
القنيطرة	القصر الصغير	الكباب	المنزل	الناظور	امزيز	بولمان	تادلا	عين اللوح	تارودانت
تازة	المحمدية	أزرو	طاطا	وجدة	سلا	تزنيت	تمارة	ترودانت	تطوان
تينمل	بني ملال	خريبكة	صفرو	ميدلت	بركان	تيفلت	وليلي	دوار بكارد	ولماس
ورزازات	تدارالزيان	خنيفرة	فكيك	كلميم	مليلية	مهدية	شفشاون	بلاد الدندون	زاكورة
سبتة	بير كاندوس	تفساي	سطات	كلميمة	كيصر	ليكسوس	مراكش	وادي زو	وادي الذهب
زايو	كاستييخو	تادلة	توريرت	طنجة	طرفاية	طانطان	سكورة	خميس الزمامرة	أبو الجعد
جرسيف	سيدي سليمان	جرادة	دادس	تالسينت	تاوريرت	بوعرفة	تنغير	سوس الأقصى	وزان
أكدز	سيدي سليمان	تزنيت	بوجدور	ايفني	ايت بلال	بوزنيقة	بوعية	فم الحصن	بنجرير
الكويرة	سيدي بوزيد	إملشيل	تمدضية	مريرت	أزيلال	فزنة	اولاد عياد	أزمور	الزحيليكة
طرفاية	اولاد تيما	تيططليل	البراشوة	الزاوية	الجرف	الريش	الخميسات	شتوكاأيت باها	الرماني
الكفاف	تمدضيت	شفشاون	الكارة	الوالدية	تداس	ولماس	العرجات	سيدي علال البحراوي	المعازيز
المعاضيد	سيدي عبد الرزاق	القصابي	بلقصيري	أصيلة	الشماعية	شيشاوي	البير	سيدي بنور	واد اهليل

							الجديد		
ايت مغان	عين الجوهرة	القصيبة	بوفكران	العطاوية	بن خيران	أولاد يعيش	ايت قريبع	سيدي قاسم	تينجداد
المحاميد	الخميس الزعامرة	تارودانت	انيفه	تروك	أم اللعب	الرحامنة	زايد	الفقيه بن صالح	السعيدية
الصخور	بن سليمان	الفنيدق	عين اعتيق	عين عودة	بني درار	المقام	أيت واحي	الكنانط	دمنات
أيت ملول	سوق الأربعاء	بالقصيري	قلعة مكونة	بومالحادس	باب برد	الطاوس	تزنيت	قلعة السراغنة	امنتانوت
تنغير	أيت يكو	حدران	برخم	جرادة	عين الشعير	بوعنان	بودنيب	سيدي إفني	انزكان

Tableau 3-1 Noms des villes et des villages marocains utilisés

سلا	فاس	زايو	جرادة	بوفكران	تارودانت	ورزازات	سيدي علال البحراوي
Un seul pseudo-mot	Deux pseudo-mots	Trois pseudo-mots	Quatre pseudo-mots	Cinq pseudo-mots	Six pseudo-mots	Sept pseudo -mots	Neuf pseudo-mots

Tableau 3-2 Exemples de noms des villes marocaines de différents nombres de pseudo-mots.

Le tableau 3-2 illustre le nombre des pseudo-mots composant les noms des villes et villages du Maroc utilisés, les noms peuvent se composer d'un, deux, trois , quatre, cinq, six, sept ou neuf sans compter les points diacritiques existant au-dessus ou en dessous et Hamza « أ- لأ- ».

Arial	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
Courier New	بـيـر كـانـدوس	غفـسـاي	سطـات	أزيـلال	فـزنـة	بـوفـكـران	أزمـور
SimplefiedArabic	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
Arial Unicode	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
Tahoma	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
Andalus	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
Tradition Arabic	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
SimplefiedArabicFixed	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
Times New Roman	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور
Microsoft Sans Serif	بير كاندوس	غفساي	سطات	أزيلال	فزنة	بوفكران	أزمور

Tableau 3-3 Exemples des noms avec 10 fontes de taille 11 et style simple.

Le tableau ci-dessus illustre dix fontes distinctes de même taille appliquées sur sept noms des villes et villages marocains. En analysant le tableau, on constate que chaque fonte présente

des élongations distinctes, la longueur et la hauteur du même nom change selon la fonte (par exemple, le cas du nom ' بيركاندوس ' avec les deux fontes Simplefied-Arabic-Fixed et la fonte TraditionArabic), suivant l'aspect calligraphie, certains caractères pouvant chevauchés horizontalement ou ligaturés ou collés avec le caractère suivant par exemple le nom 'أزيلال' avec la fonte SimplefiedArabic.

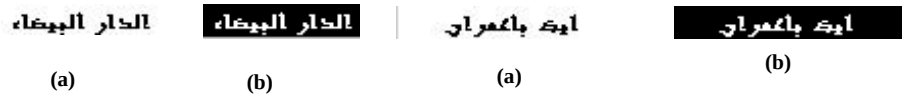
3.4 Prétraitements

Le prétraitement est une étape qui consiste à nettoyer l'image scannée pour produire une version améliorée adoptable aux étapes suivantes. Après l'acquisition d'image du nom d'une ville ou village marocain avec une résolution adéquate, l'image est transformée en format binaire suivant le processus de seuillage global, puis la région d'intérêt du nom est localisée et les parties des extrémités qui ne font pas partie de cette région sont supprimées et la qualité des pixels constituant le mot sont améliorés.

3.4.1 *Seuillage*

L'image après l'acquisition est enregistrée sous forme d'une matrice de valeurs variant entre 0 et 255. Le processus de seuillage vise à remplacer les pixels composant l'image en pixels noirs ou blancs, [133,134].

Dans notre cas (figure 3-2), Expérimentalement, nous avons adopté le seuillage global afin d'obtenir une image binaire suivant un seuil égale à 0.3 (cette valeur est déterminée expérimentalement) : le choix de ce seuil est pour garder le maximum des informations, puisqu'on considère des images de bonne qualité. Les pixels qui sont inférieurs à cette valeur sont remplacés par des pixels noirs et les pixels restants sont remplacés par des pixels blancs. Le mot est représenté avec les pixels blancs. Dans le reste de traitement, les pixels blancs représentent l'information utile.

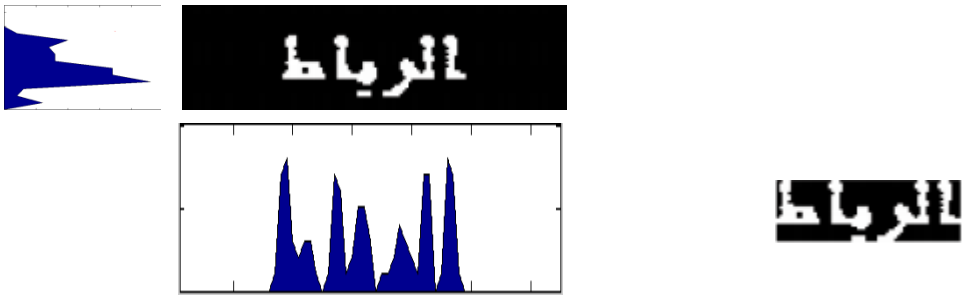


(a) (b) (a) (b)

Figure 3-2 (a) Images avant binarisation (b) Images binarisées avec un seuil=0.3.

3.4.2 *Suppression des parties inutiles.*

L'image binaire du nom est entourée par un ensemble des pixels noir qui ne font pas partie du nom de la ville marocaine. Afin de traiter les informations qui ne concernent pas le nom nous avons procédé à la suppression des parties inutiles, en se basant sur la projection horizontale (pour supprimer les deux parties supérieure et inférieure) et la projection verticale (pour supprimer les deux parties gauche et droite) (figure 3-3). L'utilité de ce prétraitement, c'est de réduire le temps d'exécution.



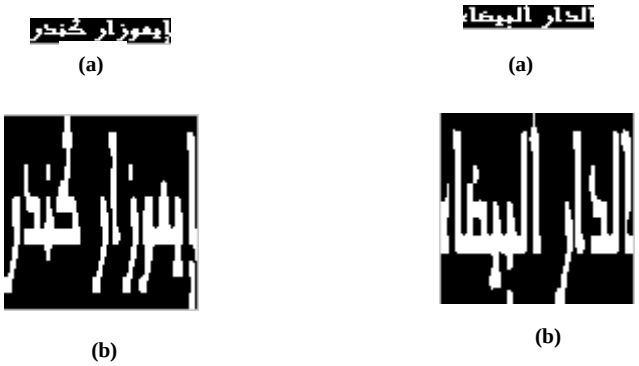
(a) (b)

Figure 3-3 Image avant la suppression (a), Image après la suppression (b).

3.4.3 *Normalisation de la taille des noms*

L'opération de la normalisation de la taille des mots est considérée le plus important prétraitement dans le système de reconnaissance proposé. Normalement, l'image représentant le mot avec des tailles différentes présentées au système est redimensionnée dans un espace standard avec une taille prédéfinie. L'objectif final pour la normalisation de la taille est de réduire la variation entre les mêmes classes (les mêmes mots).

Dans notre système, nous avons utilisé la méthode de normalisation de la taille de type plus proche voisin (détaillé dans le chapitre 1), qui est très utile pour les images binaires. La taille est fixée à 96×96 (figure 3-4).



(a) (a)

(b) (b)

Figure 3-4 (a) Image avant la normalisation, (b) Image après la normalisation de la taille.

La figure (3-5) illustre le résultat de la normalisation de la taille en une image de 96×96, des images du nom de la ville « الرباط» avec différentes tailles.

Figure 3-5 Normalisation de taille en 96×96 d'un nom avec différentes tailles.

La figure ci-dessus illustre que la vraisemblance entre les images normalisées en taille 96×96 est importante ; par exemple, le taux de corrélation entre la première et les deux dernières images normalisées est de 62.37% ce qui influence positivement sur le taux de reconnaissance. Les images résultantes de cette étape seront exploitées dans la phase suivante pour extraire un vecteur caractérisant chaque nom.

3.5 Extraction des primitives

Durant la phase de prétraitement, les matrices originales sont de taille différente, mais après la normalisation la taille est fixée en 96 lignes et 96 colonnes. Ensuite, l'image est transformée en un vecteur de caractéristiques à l'aide d'une nouvelle approche appelée « Zigzag de poids de densité ». Cette transformation a comme objectif l'amélioration de la performance du système en matière de temps d'exécution et le taux de reconnaissance.

3.5.1 *Zigzag de poids de densité*

L'extraction des caractéristiques est faite en deux étapes : la première est appelée "Poids de Densité", comme son nom indique, elle se base sur le calcul de la densité appliquée sur une matrice carrée d'ordre huit. La deuxième est nommée "Séquence zigzag", elle s'agit d'une technique simple qui vise à représenter une matrice sous la forme d'une séquence obtenue suivant un parcours particulier appliqué sur une matrice carrée.

3.5.1.1 Poids de densité

Dans la méthode de Poids de Densité, la matrice binaire de 96 lignes et 96 colonnes résultant de la phase de prétraitements est divisée en 144 zones, chacune représente une sous-matrice carrée d'ordre huit. Pour calculer le poids de chaque matrice carrée d'ordre 8, nous avons proposé une fonction de calcul de poids de densité, qui renvoi 1 si la matrice contient au moins un élément en 1 sinon le résultat sera nulle, par la suite tous les poids des sous-matrices seront représentés par 1 ou 0, afin d'obtenir une matrice carrée d'ordre 12, qui est utilisée pour extraire les séquences suivant le chemin zigzag.

3.5.1.2 Fonction de densité

La fonction de densité calcule le poids de chaque matrice carrée d'ordre 8, le poids est égal à 1 si la matrice contient au moins un pixel à 1 sinon le poids sera nul.

$$f\left(\begin{pmatrix} a_{11} & \cdots & a_{18} \\ \cdots & \cdots & \cdots \\ a_{81} & \cdots & a_{88} \end{pmatrix}\right) = \begin{cases} 1 \ si \ \exists \ a_{ij=1 \ avec \ i,j \in \{1,2,\ldots,8\}} \\ 0 \qquad\qquad\qquad\qquad sinon \end{cases} \tag{3-1}$$

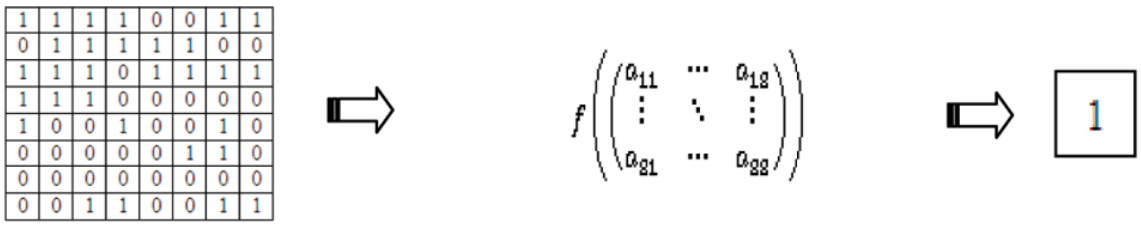


Figure 3-6 Processus de zigzag de poids de densité pour une zone.

3.5.1.3 Zigzag séquences

La matrice d'origine d'ordre 96 est divisée en 144 sous matrice d'ordre 8. En utilisant la fonction de densité nous calculons les images des sous-matrices d'ordre 8, pour obtenir une matrice carrée d'ordre 12 en sortie avec laquelle on peut extraire 144 séquences suivant le parcours zigzag (figure 3-7).

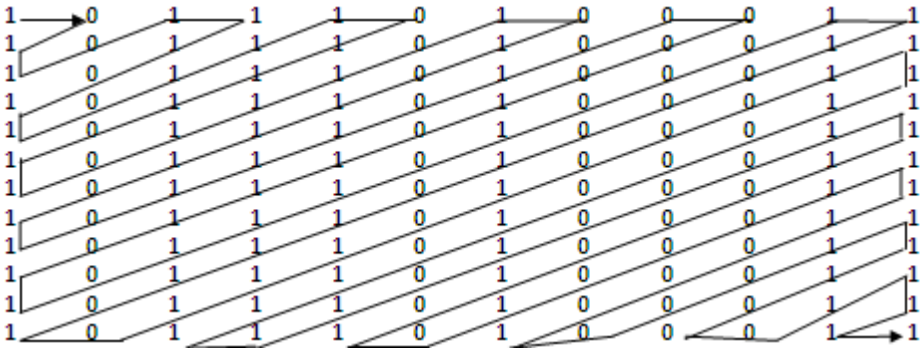


Figure 3-3-7 Parcours zigzag.

L'algorithme simulant le processus d'extraction des séquences zigzag est comme suit :

M=DW (1:12,1:12) % La matrice carrée d'entrée d'ordre 12 des poids de densité(DW).
k=1; % initialiser le nombre des éléments du vecteur Z
Pour i allant de 1 jusqu'au 23% Une boucle pour parcourir les diagonaux.
Début Si(le numéro de diagonal est paire)%si le nombre de diagonal est paire
Début SI Z(k:k+ taille (diagonal (i,M)),1) = diagonal(i,M) Z(k)
% Enregistrer les éléments de diagonal numéro i dans le vecteur Z.
K=k+ taille (diagonal (i, M))} % Incrémenter le nombre des éléments du vecteur Z
Sinon % sinon Si le numéro de diagonal est impair
Z(k:k+size(invers_brows(diagonal(i,M))),1)=invers_brows(diagonal(i,M)) %Enregistrer le diagonal dans la direction inverse dans le vecteur
K=k+ taille (invers_brows (diagonal (i, M)))} % Incrémenter le numéro du diagonal
Fin si
Fin
Z % Vecteur de sortie de 144 éléments.

3.6 Création de la base de données

Afin de tester notre approche sur une base de données des images de bonne qualité, nous avons créé une base de données en utilisant le logiciel Microsoft office 2007. En effet, nous avons créé un tableau de 200 cases, dont chacune contient un nom d'une ville ou village marocain, puis nous avons augmenté l'épaisseur des bordures en 2 points, le tableau résultant est capturé sous forme d'une image, puis nous avons utilisé un algorithme basé successivement sur la projection d'histogramme horizontal et vertical, pour extraire les images des noms contenus dans l'image du tableau, le traitement est appliqué sur 10 familles de fontes distinctes, pour chaque nom avec 11 tailles distinctes {8, 9, 10, 11, 12, 14, 16, 18, 20, 22, 24}. Les polices {11, 16, 22} sont utilisées pour le test et le reste des polices sont utilisées pour l'apprentissage. Le tableau suivant illustre plus de détails pour les familles de fontes, polices, exemples de test et d'entraînement (tableau 3.4).

Famille de Fonte	Nombre de taille	Nombre d'exemples de test	Nombre d'exemples d'entrainement
Arial	11	600	1600
Courier New Arabic	11	600	1600
SimplefiedArabic	11	600	1600
Arial Unicode	11	600	1600
Tahoma	11	600	1600
Andalus	11	600	1600
Tradition_Arabic	11	600	1600
SimplefiedArabicFixed	11	600	1600
Times New Roman	11	600	1600
Microsoft Sans Serif	11	600	1600
Total exemples par :		6000	16000
Total d'exemples utilisés:		22000	

Tableau 3-4 Informations sur la base de données adoptée.

Figure 3-8 Échantillons des noms de la base de données utilisée pour le test.

Figure 3-9 Exemples des noms de la base de données utilisée pour l'entraînement.

Les traitements décrits ci-dessus sont appliqués sur les images créées pour le test (Figure 3-8) et l'apprentissage (Figure 3-9), passant de la phase d'acquisition, la binarisation, la localisation du nom et la normalisation de la taille, puis en utilisant l'algorithme proposé appelé «Zigzag de Poids de Densité (ZPD)» pour la discrimination des images des noms avec un

vecteur de 144 paramètres. Les vecteurs d'extraction obtenus sont regroupés sous forme d'une base de données de test et d'apprentissage.

3.7 Expérimentations et résultats

Dans la phase de la classification des mots arabes imprimés, nous avons adopté deux méthodes de classification supervisées : Support Vecteur Machine (SVM) et la méthode de K-Plus-Proches-Voisin (KPPV).

3.7.1 *Utilisation de SVM pour classifier les mots arabes*

Les configurations choisis pour paramétrer le classificateur support vecteur machine sont les suivantes :

- Les vecteurs à classifier sont composés de 144 paramètres.
- La stratégie de classification adoptée est un contre tous.
- La fonction noyau de type « RBF » avec l'écart type égal à 1 est utilisée pour déterminer l'hyperplan optimal pour classifier les vecteurs de 144 primitives en deux classes.

  Les valeurs issues des taux de reconnaissance de chaque caractère ainsi que celui global sont illustrés dans le tableau suivant:

Nombre d'exemples d'apprentissage	Nombre d'exemples de test	Taux de reconnaissance (%)	Temps écoulé par seconde
1000	6000	99,50	462.413
2000	6000	99,50	2806.54
3000	6000	99,50	1729.67
4000	6000	99,50	2959.27
5000	6000	99,50	3644.64

Tableau 3-5 Taux de reconnaissance des noms des villes marocaines par SVM.

D'après les résultats affichés dans le tableau ci-dessus, on constate que le taux de reconnaissance de 6000 exemples reste stable avec l'augmentation du nombre des exemples d'apprentissage. Le temps nécessaire pour l'apprentissage et la classification augmente par rapport au nombre des exemples d'apprentissage.

3.7.2 *Utilisation de KPPV pour classifier les mots arabes imprimés*

Les paramètres adoptés dans le cas d'utilisation de KPPV sont :

- L'extraction des 144 primitives est effectuée par notre approche « Zigzag de poids de densité Zigzag Séquence ».

- La règle de classification choisie pour KPPV est Consensus, i.e. la classe le plus fréquemment trouvée des k plus proche voisins est élue comme la classe de sortie, dans le cas d'égalité le plus proche est considéré.

- La distance euclidienne est utilisée pour calculer la distance entre les classes de test et d'entrainement en considérant la distance minimal pour déterminer les classes les plus proches.

- Le nombre des voisins K est varié de 1 à 11 pour analyser et comparer les résultats de la classification et déterminer l'optimal nombre des voisins.

Échantillon d'apprentissage	16000										
Échantillon de test	6000										
Nombre de voisins (K)	1	2	3	4	5	6	7	8	9	10	11
Taux (%) Notre approche	95,26	97.91	99.04	99,57	99,77	99,90	99,94	99,98	100	100	100
Temps écoulé par seconde	349,86	375,58	471.1	471,84	496,34	468,51	465,74	466,42	466 ,56	467,97	468,63
Taux (%) Moment de Hu[]	27.75	56.79	82.0459	90.11	96.06	97.58	98.97	100	100	100	100
Temps écoulé par seconde	16.95	25.54	24.67	26.08	26.62	24.90	24.33	26.30	30.40	29.96	30.35

Tableau 3-6 Taux de reconnaissance des noms des villes du Maroc en utilisant KPPV.

Le tableau 3-6 illustre les taux de la reconnaissance des noms des villes et villages du Maroc en utilisant le classificateur K plus proche voisins, en fonction du nombre des voisins K, commençant d'un seul voisin jusqu'au onzième, aussi nous avons calculé le temps d'exécution écoulé par seconde pour la reconnaissance de 6000 exemples en se basant sur 16000 exemples d'apprentissage.

En analysant les résultats obtenus, on constate que le taux de reconnaissance et le temps d'exécution s'augmentent avec l'évolution de K d'une façon remarquable, surtout à partir de k=1 jusqu'au k=5, le taux de reconnaissance passe de 95,26% au 99,77% et le temps d'exécution passe de 349,86 seconde au 496,34 seconde, pour k>5 le temps d'exécution ne dépasse pas 469 seconde ce qui donne 0,07 seconde par mot et le taux de reconnaissance atteint la valeur maximal avec k=9,10 et 11.

3.7.3 *Comparaison des résultats et discussions*

- ✓ **Limite technique**

Les configurations matérielles et le logiciel utilisé pour mettre en place le système développé et tester la performance des techniques proposées sont les suivants: un PC Dual-Core de vitesse 2.00 Ghz, l'espace mémoire (RAM) de 2,00 Go, et le langage de programmation Matlab 2010.

L'utilisation des SVMs multiclasses requière un espace mémoire de taille importante plus que KPPV déterminée en fonction de la taille des échantillons d'apprentissage pour créer la structure avec laquelle on peut classifier les échantillons de test. Cependant, la classification des noms en utilisant la méthode KPPV, nécessite chaque fois le calcul de la distance entre les

échantillons à classifier et les échantillons d'apprentissage, ce qui explique que le temps nécessaire d'apprentissage et classification est toujours en fonction de la taille des populations d'entraînement.

- ✓ **Temps d'exécution**

En matière de temps d'exécution SVM est meilleur que KPPV, si on ne compte pas le temps nécessaire pour créer la structure d'apprentissage, par contre, KPPV est meilleure au terme de temps global d'exécution qui est le temps d'apprentissage plus le temps de classification.

La méthode de classification SVM est limitée suivant la taille mémoire disponible, elle demande un espace mémoire assez important en fonction de la taille de population vis-à-vis la méthode de KPPV.

La convergence de l'algorithme d'apprentissage de SVM n'est pas toujours assurée pour toutes les fonctions noyau, en revanche, elle dépend de la nature et la taille des échantillons d'apprentissage et l'espace mémoire.

Notre approche demande un temps d'exécution plus que la méthode des Moments de Hu, vu le nombre important des paramètres utilisés (144) vis-à-vis 7. Le taux peut être amélioré en optimisant le nombre des primitives calculées.

- ✓ **Taux de reconnaissance**

La méthode de classification KPPV, atteint le taux de reconnaissance maximale pour les neuf plus proches voisins (k=9), cependant, avec la méthode de classification SVM, le taux de reconnaissance reste stable en 99,50 % même avec l'augmentation de la taille de la population d'apprentissage.

Notre approche donne un taux de reconnaissance très encourageant vis-à-vis la méthode des Moment de Hu surtout pour k=1 jusqu'à k=7. Par contre les résultats sont généralement similaires pour $7<k\leq11$.

Dans ce travail, nous avons utilisé des images des mots relativement de bonne qualité, ce qui explique les meilleurs résultats obtenus, cette expérience nous a permet de conclure qu'il est possible de réaliser un système de reconnaissance des mots arabe imprimés très performent en se basant sur un module de prétraitement performant, capable de nous rendre les images non bruitées, bien nettoyées et de bonne qualité.

3.8 Conclusion

Le présent travail s'inscrit dans le domaine de la reconnaissance des mots en respectant l'approche globale.

Dans ce travail, nous avons proposé une nouvelle technique appelée "Zigzag de Poids de Densité (ZPD)" pour caractériser les mots arabes imprimés, notre approche vise à représenter les images des mots à reconnaitre dans un espace de dimension fixe, en se basant sur le calcul de zigzag de poids de densité et l'extraction de 144 séquences suivant le chemin zigzag. Cette technique est appliquée sur les noms arabes imprimés des villes et villages du Maroc.

Cette technique est testée sur une base de données de 22000 images dont 6000 images représentent la base de test et 16000 images représentent la base d'apprentissage, le taux de reconnaissance obtenu atteint le taux maximal dans le cas où le nombre de voisins égal à 9 (KPPV), en revanche le taux de reconnaissance reste stable en 99,5 % pour la méthode de classification SVM.

Avec ce travail, la question que nous avons posé au début, concernant la possibilité de réaliser un système de reconnaissance des mots arabe imprimés performant, reste réalisable en développant un module puissant de prétraitement permettant d'éliminer les bruits et en adoptant le processus décri dans notre système.

Pour perfectionner le travail, il nous faut un sous-module d'identification des familles des fontes, tailles et styles des mots, afin d'améliorer le temps d'exécution et le taux de reconnaissance, ce qu'est l'objectif du chapitre suivant, qui s'intéresse à l'identification des familles de fontes, tailles et styles.

Chapitre 4: Reconnaissance de la famille de fonte arabe, tailles et styles

4.1 Introduction

Les systèmes de segmentation et reconnaissance de l'écriture arabe, souffrent dans l'étape de segmentation [124,125, 135, 136], du fait qu'ils adoptent un seul algorithme pour segmenter en caractères les différents mots composés de différentes famille de fontes, styles et tailles. Cette hypothèse considère que les différentes fontes ont la même structure calligraphique.

Dans les systèmes du traitement du document, les deux phases (segmentation et reconnaissance) sont indispensables pour convertir les images du document en formats (.txt, .doc, .docx); la reconnaissance de la fonte préalable du texte permet d'améliorer l'efficacité du système de segmentation et de reconnaissance de l'écriture arabe. Récemment, autres systèmes intermédiaires de reconnaissance de fonte sans traiter les styles ont été proposés [137, 138, 139], qui souffrent à cause de la forte similarité entre certaines fontes.

Dans la littérature, il existe une approche basée sur la fenêtre de langueur fixe de droite vers la gauche d'image du mot pour estimer la vraisemblance des catégories des fontes en utilisant GMMs [140], aussi, autre approche ont été basé sur l'approche stochastique pour identifier la fonte et la taille, évalué sur les images des mots arabe de basse résolution qui a montrée clairement l'importance et le potentiel de reconnaissance de fonte suivi par la reconnaissance des mots mono-fonte, l'erreur de reconnaissance de caractères et mots peut être réduit de plus de 70% lorsque on utilise un système de reconnaissance de fonte en premier temps suivi d'un système mono fonte OCR[10]. La reconnaissance de fonte arabe basée sur l'approche apriori en utilisant l'algorithme de Steerable Pyramide (SP) avec 6 orientations qui donne un taux de reconnaissance très élevé vis-à-vis l'utilisation de l'algorithme de Niveau de Gris Cooccurrence matrice. L'utilisation de l'algorithme Gabor Filtre et Wavelets évaluent en utilisant la base de données des images du texte arabe imprimé multi-fonte APTID/MF et rétro propagation Réseau de Neurone (BPRN) pour la classification.

Dans ce travail, on propose une nouvelle approche de reconnaissance optique de fonte arabe basée sur l'extraction des dix derniers pixels pour chaque mot ou pseudo-mot de la base des images du texte arabe imprimé APTI de basse résolution de multi-fonts, tailles et styles avec différentes conditions de dégradation [1].

Le système de reconnaissance optique de fonte arabe doit reconnaitre la famille de fonte la taille et aussi le style, pour garder les styles du document source. Le système proposé est basé sur trois étapes : prétraitement, extraction de caractéristiques et classification. Dans l'étape de prétraitement, on utilise la technique de seuillage pour convertir l'image des dix derniers pixels acquise en image binaire. Certaines mesures statistiques de l'histogramme et une nouvelle approche appelée Pixel Continuité de différentes directions de la matrice horizontal, vertical, diagonal et anti-diagonal sont calculés pour extraire 20 primitives pour identifier la

fonte, la taille et le style pour chaque mot ou pseudo-mot. La performance du système est évaluée en utilisant la base de données des images du texte arabe imprimé APTI et le classificateur de K plus proche voisin en utilisant trois distances : la distance Citybloc, la distance Euclidien et la distance Corrélation, les résultats des sorties sont déterminés suivant la majorité de vote, et finalement comparer avec d'autre expérimentation.

4.2 Identification de fonte

Le texte contenu dans les documents arabes imprimés est composé d'une ou plusieurs familles de fonte, style et taille, les auteures cités dans [10, 141, 142] considèrent que l'identification de fonte de certains caractères composant le même paragraphe est suffisante pour déterminer la fonte dominante dans ce paragraphe, d'une autre, il n'est pas nécessaire de reconnaitre la fonte pour tous les caractères composant le même paragraphe.

Dans notre approche, on considère que le document ou particulièrement le paragraphe peut être édité avec plus d'une seule fonte, taille et style, pour cette raison et pour garder la fonte, taille et style du document source, le processus de reconnaissance est fait mot par mot, cette hypothèse est testé sur la base d'image du texte arabe imprimé APTI [1].

4.3 Fonte arabe

La fonte arabe est l'une des fontes complexes de l'écriture imprimée de droite à gauche constituée de 28 caractères contenant un ensemble de points diacritiques en haut et en bas et jamais en haut et en bas au même temps, chaque caractère peut prendre quatre structures différentes (figure 4.3 et 4.4) suivant la position et le style. La richesse des styles calligraphiques des fontes arabes donnent des ligatures verticales et horizontales comme indiquées l'exemple des figures 4.1, 4.2 et 4.5.

Figure 4-1 Ligature horizontale dans les mots arabes de la base APTI [1].

Figure 4-2 Le caractère "ain" dans différentes positions

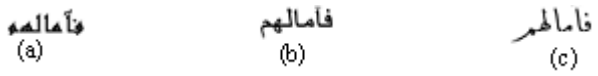


Figure 4-3 Mot en trois familles de fonte de la base APTI [1]
(a) famille de fonte Et Andalus, Gras et taille=10.
(b) famille de fonte Arabic Transparent, Gras et taille=10.
(c) famille de fonte DecoTypeThuluth, Gras et taille=10.

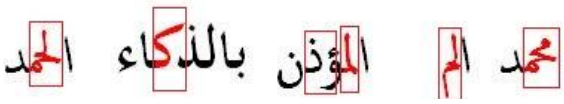


Figure 4-4 Mots présentent des ligatures verticales et horizontales.

بالدكاء. بالدكاء. **بالدكاء**. بالدكاء. بـا لـد كـا ء . بالدكاء

Figure 4-5 Un mot Arabe avec deux styles simple et Gras, taille 14 et six différentes fontes.

La figure 4-6 ci-dessous présente le même mot "لآرائهم" avec trois familles de fontes: Arabic-Transparent, Decotype-Thuluth et Andalus avec diffèrent styles Gras, Gras-Italique, Italique et Simple avec la même taille 14. Elle illustre aussi la similarité entre les styles de la même fonte qui est significative, cet effet est l'un des problèmes major de processus de reconnaissance de fonte et qui influence négativement sur la performance du système (OFAR).

Figure 4-6 Un mot arabe avec trois fontes et quatre styles et la taille 14.

4.4 Architecture du système proposé.

L'architecture du système de reconnaissance de fonte arabe proposé est composée de plusieurs étapes : le prétraitement, l'extraction de primitives des dix derniers pixels du mot et la prise de décision par le classifieur de K plus proche voisin (figure 4-7).

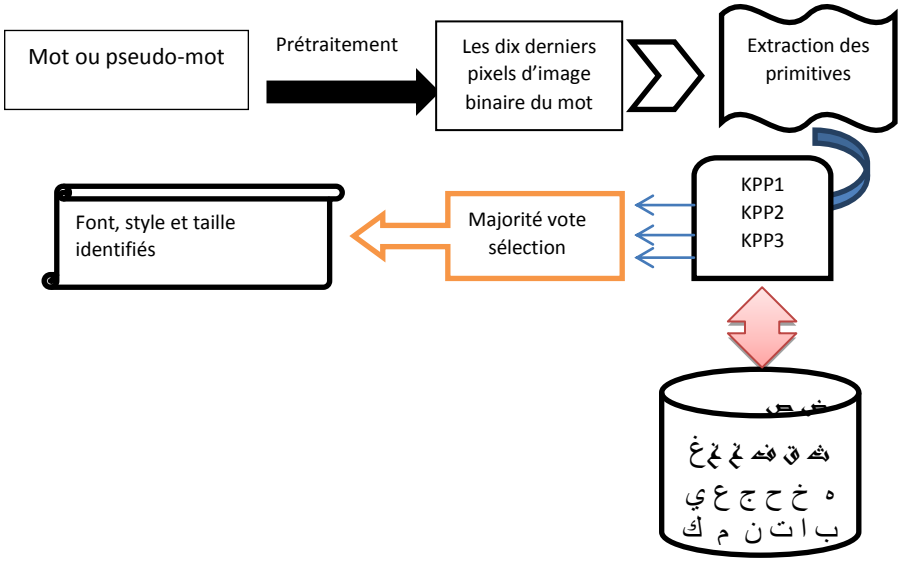


Figure 4-7 System de reconnaissance de font arabe proposé

Après l'acquissions de l'image dans la première étape qui est le prétraitement, l'objectif général est de nettoyer et préparer l'image en vue de faciliter les traitements ultérieurs, l'image du mot arabe imprimé est transformée en image binaire puis les parties inutiles du mot sont supprimées pour localiser le mot en se basant sur l'histogramme horizontal et vertical, finalement, les dix derniers pixels de l'image normalisée sont extraits pour les utiliser dans l'étape suivante.

Dans l'étape d'extraction, en vue de représenter les mots sous forme des séquences d'observation, nous avons proposé un nouvel algorithme appelé "Continuité des Pixels" et autres primitives statistiques, 20 paramètres caractérisant le mot sont extraits de dix derniers pixels d'image binaire du mot ou pseudo-mot. Les données collectées de l'étape d'extraction sont utilisées pour élaborer la base de données d'entraînement et de test.

La classification des fontes des mots arabe imprimés suivant la famille de fonte, la taille et le style est réalisée à l'aide des classifieurs K-plus-proche-voisin en utilisant trois distances, la prise de décision est faite suivant le vote majoritaire des trois résultats issu, des trois KPPV pour chaque distance. Dans la suite de ce chapitre, nous présentons les étapes constituant le synopsis de notre système de reconnaissance de fonte arabe en donnant simultanément les exemples illustrant chaque niveau de processus de traitement.

4.5 Prétraitements

Dans cette section, on présente les méthodes des prétraitements que nous avons appliqué afin de nettoyer et améliorer l'image d'origine. Les images utilisées de la base des mots arabes imprimés, sont souvent entachées de différents types de bruit de différentes sources (Poussière, qualité de scanne, résolution d'image, transformation…). L'objectif des processus qui sont présentés dans ce qui suit, à savoir : la binarisation [143,144], la localisation du mot par la suppression des parties indésirables, se résume essentiellement en deux points : le premier point concerne la réduction de variabilité d'écriture imprimée et surtout de la même fonte, en deuxième point la suppression ou l'atténuation des informations inutiles. L'opération de segmentation n'est pas considérée puisqu'on exploite une base de données de formes isolées.

4.5.1 *Binarisation et changement de bits*

Les images introduites au système ne sont pas toujours des images binaires, mais ils peuvent être en différents modes (Niveau de gris, couleurs…), le processus de binarisation est un cas particulier de seuillage, qui vise en général à représenter l'image en deux classes de pixels noir et blanc afin de distinguer entre les pixels de premier plan associé à l'écriture et les pixels de l'arrière-plan représentant le fond. Nous avons opté pour la méthode d'Otsu [121], qui consiste à calculer un seuil unique global pour toute l'image qui minimise la variance intra-classe entre les deux classes noir et blanc, la figure ci-dessous présente l'image du mot arabe (بالدكاء) avant et après la binarisation et changement de bits.

(a) (b)

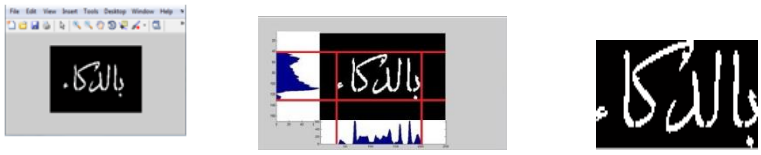
Figure 4-8 Image avant (a) et après (b) la binarisation et changement de bits.

L'image binaire résultante du processus de binarisation représente le mot avec la classe noire, au niveau matriciel les informations utiles représentant le mot sont les zéros. Le processus de changement de bits vise à remplacer la classe blanche par la classe noire. Au niveau des calculs matriciels, il est préférable de traiter les uns que de traiter les zéros pour avoir minimisé le temps de calcul.

4.5.2 *Localisation du mot*

L'étape précédente produit une image binaire avec le changement de bits, à ce niveau, le mot se trouve au milieu d'image, pour minimiser le temps d'exécution des calculs dans le reste du traitement, nous devons supprimer les parties d'image entourant le mot et qui ne donnent aucune information utile pour le système.

Le processus de localisation du mot dans l'image se base sur l'utilisation de l'histogramme. La Figure 4-9 ci-dessous présente une image d'un mot arabe imprimé, son histogramme vertical et horizontal, et le résultat obtenu du découpage des parties inutiles.



(a) (b) (c)

Figure 4-9 Processus de localisation du mot;
(a) Image binarisée et bits changés. (b) Détermination de la zone du mot. (c) Mot localisé.

La position du mot est localisée en exploitant les histogrammes horizontal et vertical de l'image binaire du mot. En effet, pour déterminer la position de découpage des deux parties gauche et droite, on utilise l'histogramme vertical de l'image du mot, pour déterminer le point de découpage, on parcourt l'histogramme vertical du côté gauche jusqu'à ce que l'avant position où l'histogramme devenir non nulle, le même processus s'applique pour déterminer les points de découpage à droite, mais cette fois, on parcourt l'histogramme vertical d'histogramme du côté droit. Les points de découpage en haut et en bas sont déterminés en exploitant l'histogramme horizontal et avec le même processus adopté pour le cas de découpage des

parties gauche et droite, on parcourt l'image de haut vers le bas pour le découpage de la partie en haut et du bas vers le haut pour le découpage de la partie en bas.

4.5.3 *Extraction des dix derniers pixels*

L'utilisation de la totalité du mot pour identifier la fonte augmente le temps de traitement, dans notre approche, on considère qu'une petite partie du mot suffit pour extraire certaines caractéristiques pour identifier la fonte. Dans le même aspect, on considère que les dix derniers pixels d'image binaire du mot sont suffisants pour extraire les primitives de chaque mot.

(a) (b)

Figure 4-10 Extraction des dix derniers pixels.
(a) Mot localisé.
(b) Dix derniers pixels du mot.

La figure ci-dessus illustre un mot et ces dix derniers pixels, le processus est appliqué sur l'image du mot après la suppression des parties inutiles entourant le mot, on commence du côté gauche d'image du mot et on extrait dix pixels qui sont utilisés dans le reste de traitement. Quand l'image du mot ou pseudo-mot de langueur moins de dix pixels l'image entière est retenue.

4.6 Extraction des primitives

L'intérêt de cette phase est de transformer l'image du mot arabe imprimé en un vecteur de caractéristiques de dimension fixe, Cependant l'utilisation d'image du mot pour identifier directement le mot semble très difficile à cause de la morphologie des mots et de la grande variabilité liée au style et taille d'écriture utilisée et dans certains cas du bruit entachant l'image, pour ces raisons, il est nécessaire de sélectionner ou de calculer à partir de la matrice du mot, un ensemble de primitives pertinentes permettant l'identification de façon efficace et facile ces caractéristiques qui peuvent être de divers types, mais ils doivent être discriminantes [146, 147, 148].

Dans cette étude, le mot est représenté avec les pixels de valeur égale à 1 dans la matrice d'image binaire des dix derniers pixels extraits du mot. L'approche proposée de caractérisation de fonte appelée, Continuité de Pixels (CP), permet de fournir des informations sur la distribution des chaînes des pixels de quatre directions : horizontale, verticale, diagonale, et anti-diagonal.

La technique d'extraction proposée calcule le nombre des pixels successifs en 1, le maximum, et la moyenne de longueur de chaîne des pixels de quatre directions sont calculés afin d'obtenir 12 paramètres caractérisant la continuité de pixels pour chaque fonte.

Les longueurs de la chaîne de pixels horizontale et verticale sont extraites pour identifier les styles Simple et Gras, tandis que les longueurs de la chaîne de pixels diagonale, et anti-diagonale pour identifier et fournir plus d'information sur les styles Italique et Italique-Gras. Les dix derniers pixels (figure 5-11) du mot ou pseudo-mot sont analysés pour extraire certaines primitives : minimum, maximum, et la moyenne de Continuité de Pixels (CP) horizontale, verticale, diagonale, et anti-diagonale. D'autres primitives sont calculées en se basant sur les statistiques extraites de l'histogramme vertical et horizontal.

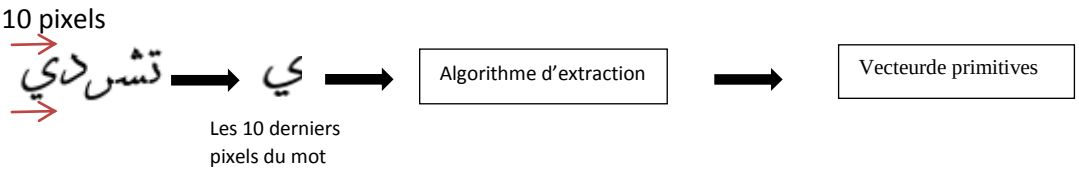


Figure 4-11 Processus d'extraction de primitives.

Durant le processus d'extraction de primitives, un cas particulier est survenu, lorsque l'image du mot ou pseudo-mot après le processus de localisation est de langueur moins de dix pixels, cette fois-ci l'image entièrement est utilisée.

Dans cette étude, on considère que la fonte est déterminée suivant la famille de fonte, la taille et le style.

Figure 4-12 Les dix derniers pixels du mot (لآرائهم) de fonte Arabe Transparent de taille 14.

La figure 5-12 illustre la similarité importante entre les styles de la même fonte et de même taille, la fonte Arabic-Transparent de taille 14 présente plus de 73% de similarité entre les deux styles Gras-Italique et Italique; et plus de 77% de similarité entre le style Gras et Simple. Le problème de forte similarité entre certain fonte et intra fonte est l'un des problèmes major empêchant la réalisation d'un système fiable de reconnaissance de fonte multi-taille et multi-styles.

4.6.1 *Continuité des pixels horizontaux*

Chaque fonte présente une épaisseur de trait d'écriture qui reste généralement inchangeable au milieu des graphèmes du mot ou caractère, certains changements peuvent apparaitre au niveau des extrémités et les fins du graphème, aussi au point d'accolement de deux caractères, cet aspect est traduit au niveau de la distribution horizontale des pixels horizontal, les ascendants verticaux des caractères présentent clairement le trait d'écriture et la

distribution horizontale des pixels (figure 5-13), l'aspect esthétique, calligraphique, et structurelle de l'écriture varie en relation avec la famille de fonte, la taille et le style choisi

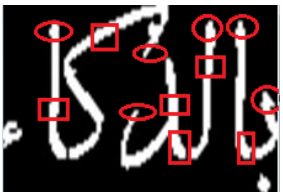


Figure 4-13 Variation de trais d'écriture du mot.

Chaque fonte est définie suivant la famille de fonte, la taille et le style, ces derniers déterminent l'aspect visuel, structurel et calligraphique des caractères et mots.

Pour calculer le nombre des pixels continus au niveau horizontal composant une seule chaine, il faut parcourir chaque ligne dans la matrice binaire des dix derniers pixels du mot afin de déterminer la langueur de pixels continus. Ce processus est illustré dans la figure (5 -14).

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

(a) (b)

Figure 4-14 (a) Avant la sélection horizontale (b) après la sélection horizontale

La figure 5-14explique le processus de sélection qui s'opère sur les lignes de la matrice binaire, en Effet, le processus analyse chaque ligne en comptant le nombre de pixels connexes dont la valeur est 1.

Dans la première ligne (figure 5-14), nous avons deux chaines de pixels blancs, la longueur de la première chaine est un pixel, cependant la deuxième chaines constitue de trois pixels blancs. Dans la deuxième ligne, nous avons une seule chaine de longueur sept pixels blancs et la dernière ligne contient deux chaines, la longueur de chacune vaut trois. Le minimum, le maximum et la moyenne de la longueur des chaines de pixels connexes sont utilisés comme des primitives.

L'application de ce processus a donné les résultats suivants:

- Continuité des pixels horizontaux(M) = [1 3 7 3 3]
- Max(Continuité des pixels horizontaux) = 7
- Min(Continuité des pixels horizontaux > 1) = 3
- Moyenne(Continuité des pixels horizontaux) = 3.4

Certaines chaines ont une longueur de pixels égale à 1, pour éviter le cas des pixels isolés, le minimum est calculé pour la longueur de chaines supérieures à 1.

4.6.2 *Continuité des pixels verticaux*

La continuité des pixels au niveau vertical, varie dans le même mot (Figure 5-15), en effet, le même mot arabe imprimé se compose de plusieurs caractères de différentes structures synthétiques et calligraphiques donnant une variation de distribution de pixels verticalement, certains de ces caractères présentent une langueur verticale maximale de la chaine, comme le cas de "م ,ا", et minimal par exemple"ت ,ب". Généralement la variation de distribution de pixels est toujours encadrée entre une langueur maximale et autre minimale déterminée suivant la propriété de la fonte,

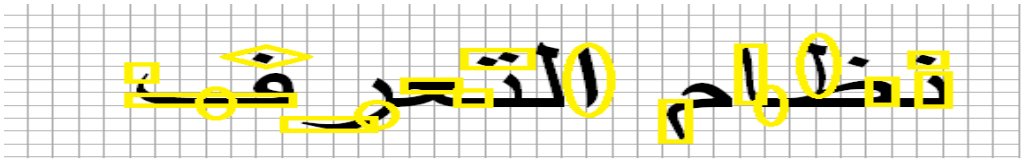


Figure 4-15 Variation de distribution des pixels dans des directions différentes d'un mot arabe imprimé.

Pour déterminer l'aspect général de distribution verticale de pixels dans un mot arabe, nous allons calculer la longueur de continuité de pixels maximale, minimale et moyenne dons le cas vertical.

Le processus de calcul de la continuité de pixels, verticalement, est appliqué sur la matrice binaire représentant le mot. La continuité de pixels est calculée en analysant les colonnes, pour trouver les chaînes de pixels continus verticalement (figure 5-16). La longueur des chaînes est déterminée en comptant le nombre des pixels blancs connexes.

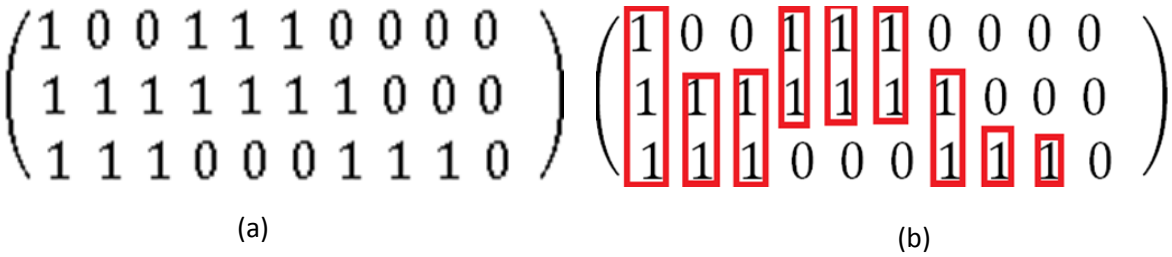


Figure 4-16 (a) avant la sélection verticale (b) après la sélection verticale.

Après le calcul, les résultats obtenus sont :

- Continuité des pixels verticaux(M) = [3 2 2 2 2 2 1 1]
- Max(Continuité des pixels verticaux) = 3
- Min(Continuité des pixels verticaux > 1) = 2
- Moyenne(Continuité des pixels verticaux) = 1.88

Pour éviter le cas des pixels isolés, les chaînes des pixels blancs connexes de langueur moins de deux pixels ne sont pas considérés.

4.6.3 *Continuité des pixels diagonaux*

La distribution des pixels des fontes de style Italique et Italique-Gras présente certain différence pour les deux autres styles Gras et Simple. Le style Italique étale le caractère arabe imprimé vers la droite tandis que le style Gras augmente l'épaisseur de trait d'écriture

Pour cerner l'aspect de distribution de pixels dans toutes les directions matricielles et fournir plus d'informations pour identifier la fonte, on procède au calcul de la longueur des chaînes diagonales de pixels, reflétant certains aspects calligraphiques et structurels surtout pour les extrémités du mot.

Dans ce processus, les lignes diagonales sont parcourues pour déterminer la continuité de pixels (figure 5-17).

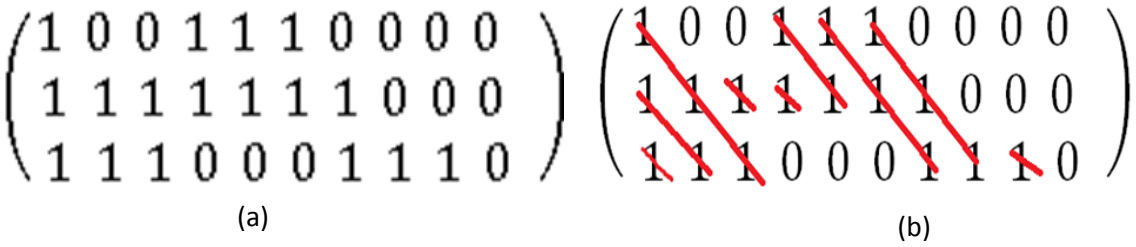


Figure 4-17 (a) avant la sélection diagonale (b) après la sélection diagonale.

Les résultats obtenus, après l'adoption des étapes de ce processus, sont:

- Continuité de Pixels Diagonaux(M) = [1 2 3 1 1 2 3 3 1]
- Max(Continuité de Pixels Diagonaux) = 3
- Min(Continuité de pixels Diagonaux > 1) = 2
- Moyenne(Continuité de Pixels Diagonaux) = 1.87

La longueur de la chaine des pixels diagonaux devenir minimale dans les extrémités, pour les longueurs de chaine de pixel égaux à 1 ne sont pas retenus comme des longueurs minimales, puisque ils ne caractérisant pas la fonte.

4.6.4 *Continuité des pixels antidiagonaux*

Les pixels formant les lignes antidiagonaux de la matrice binaire sont adoptés pour calculer le minimum, le maximum, et la moyenne de la chaine de pixels antidiagonaux continus.

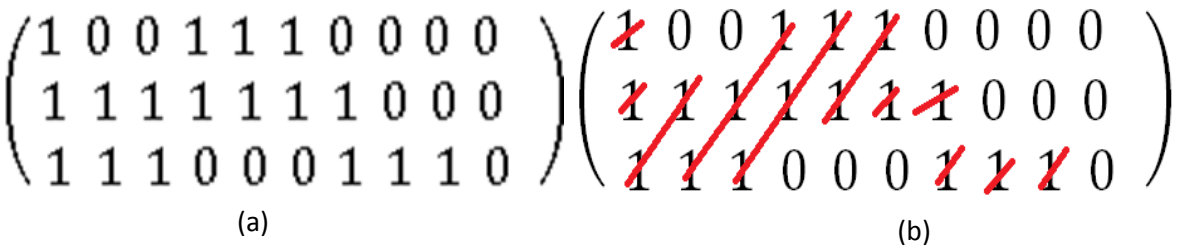


Figure 4-18 (a) avant la sélection antidiagonale (b) après la sélection antidiagonale.

L'exécution de la dite méthode a donné les résultats suivants:

- Continuité des pixels antidiagonaux(M) = [1 1 2 3 3 2 1 1 1 1 1]
- Max(Continuité des pixels antidiagonaux) = 3
- Min(Continuité des pixels antidiagonaux > 1) = 2
- Moyenne(Continuité des pixels antidiagonaux) = 1.6

Finalement, on obtient un vecteur de 12 primitives pour identifier la fonte du mot ou pseudo mot.

V=[7 3 3,4 3 2 1,88 3 2 1,87 3 2 1,6].

Huit primitives additionnelle ont été considérées pour fournir des statistiques sur les histogrammes horizontal et vertical en calculant ces valeurs maximum, minimum et moyenne. La hauteur et la densité des pixels blancs des dix derniers pixels sont aussi calculés. Dans la section suivante on présentera en détaille ces caractéristiques.

4.6.5 *Autre primitives statistiques utilisées*

La variation de la taille de fonte, implique le changement d'échelle de représentation des caractères ou des mots dans toutes les directions, tandis que le changement de style donne un nouvel aspect calligraphique et structurel.

En plus de primitives indiquées ci-dessus, les paramètres statistiques suivants sont utilisés:

- ✓ Minimum d'histogramme horizontal supérieur à 1.
- ✓ Maximum d'histogramme horizontal.
- ✓ Moyenne d'histogramme horizontal.
- ✓ Minimum d'histogramme vertical supérieur à 1.
- ✓ Minimum d'histogramme vertical.
- ✓ La moyenne d'histogramme vertical.
- ✓ Le taux de densité de pixels blancs dans les 10 derniers pixels d'image du mot.
- ✓ La hauteur des 10 derniers pixels d'image du mot.

L'analyse de projection d'histogramme vertical, fourni des informations en relation avec la taille et le style, en effet, le pic ou la valeur maximale représentant le caractère le plus long verticalement, la valeur minimale représente soit les extrémités ou l'épaisseur de la ligne de base, la valeur moyenne d'histogramme présente la valeur approximative de distribution des pixels pour une fonte.

4.7 Technique de classification

Les vecteurs de primitives de chaque mot ou pseudo-mot extraits de la phase d'extraction, sont collectés et regroupés en deux base de données, une pour l'apprentissage et l'autre pour le teste, chacune représente une famille de fonte de taille et style unique, par exemple la fonte nommée « DecoType_Thuluth_12_Gras-Italique » représente la famille de fonte « DecoType_Thuluth » de taille 12 et de style Gras-Italique. Les classes représentant chacune des fontes sont étiquetées en utilisant des nombres de 1 à 36.

La technique de classification supervisée est adoptée pour déterminer la classe d'appartenance de chaque famille de fonte suivant la taille et le style. En utilisant trois classificateurs de type K Plus Proche Voisin, chacune avec une distance différente ; la prise de décision est déterminée suivant le vote majoritaire sur les résultats obtenus des trois classificateurs. Figure 5-19.

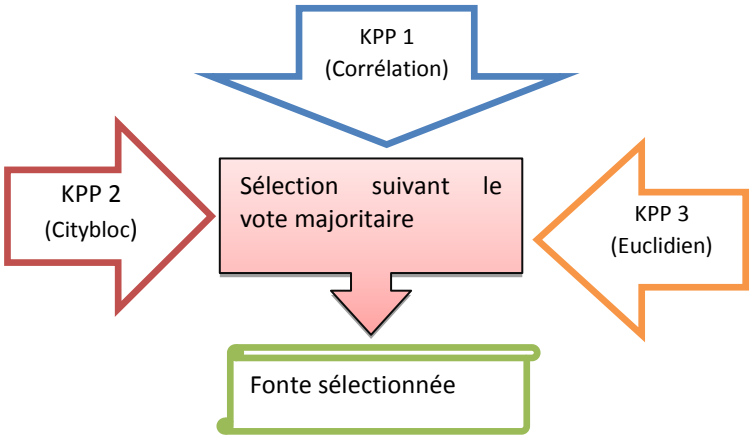


Figure 4-19 Processus de décision par le vote Majoritaire.

Le processus de décision illustré dans la (figure 5-19), s'opère en utilisant les résultats obtenu pour chacun des trois classifieurs, les résultats de classification sont comparés deux à deux, si deux résultats sont les même, l'un des résultats est retenu comme une décision final, tandis que si les trois résultats sont défirent le vote est nul et aucune décision n'est prise.

L'algorithme de sélection par vote majoritaire se déroule comme suit:

```
Résultat 1← Résultat de KPP1
Résultat 2← Résultat de KPP2
Résultat 3← Résultat de KPP3
Résultat-sélectionné  ← 0
Début
SI (Résultat 3 égale à Résultat 2)
Résultat-sélectionné ←Résultat 3
Fin SI
SI (Résultat 2 égale à Résultat 1)
Résultat-sélectionné ← Résultat 2
Fin SI
SI (Résultat 1  égale à Résultat 3)
Résultat-sélectionné ←Résultat 1
Fin SI
Fin
Résultat-sélectionné  % résultat final de la sélection
```

Dans le cas d'égalité des résultats des classifieurs, l'un des résultats est considéré, par contre, le cas de rejet correspond au non égalité des résultats des classifieurs.

4.8 Expérimentations et résultats

Les différentes phases de notre système de reconnaissance de fonte des mots arabes imprimés sont décrites ci-dessus. L'approche proposée est indépendante de la nature de script. Elle traite généralement la forme du mot et du caractère sans segmentation préalable. Nous avons adopté la modélisation de type K plus proche voisin, en se basant sur les caractéristiques de continuité des pixels dans différentes directions, qui sont extraites des images binaires des dix derniers pixels du mot, en plus de huit paramètres statistiques de l'histogramme.

Pour mettre l'éprouve de notre système de reconnaissance de fonte arabe, nous avons réalisé des expériences sur la base APTI (Arabic Printed Text Image); qui est composée d'un nombre important des images des mots arabes imprimés de différentes familles de fontes, tailles et styles. Dans la suite, nous présentons la base de données APTI utilisée et les expérimentations effectuées.

4.8.1 *Base APTI (Arabic Printed Text Image)*

Pour tester la performance de la méthode proposée, des sous-ensembles de la base APTI ont été utilisés pour réaliser des expériences.

APTI est considérée la première base de données «disponible publiquement » de vocabulaire large et à très basse résolution (72 dpi), pour la reconnaissance du texte arabe multi-fontes, multi-taille et multi-styles. Cette base a été développée en 2009 dans le cadre d'une collaboration entre le groupe DIVA (Document, Image and Voice Analysis) de

l’université de fribourg-Suisse et le groupe REGIM (Research Group in Intelligent Machines) de l’université de Sfax-Tunisie [1]. La base de données est composée de plusieurs groupements: chaque famille de fonte est regroupée sous un répertoire comportant les différentes sous regroupement suivant les styles et les tailles, cette base des mots est générée à partir d'un lexique de 113284 mots arabes de 10 fontes et 10 tailles. Les figures (5-20, 5-21, 5-22) suivantes illustrent des échantillons utilisés de la base des mots arabes imprimés APTI.

Figure 4-20 Echantillon de la fonte DecoTypeThuluth 12 Gras [1].

Figure 4-21 Echantillon de la fonte ArabicTransparent 12 Italique [1].

Figure 4-22 Echantillon de la fonte Andalus 16 Gras Italique [1].

4.8.2 *Tests et résultats expérimentaux*

Pour évaluer notre approche, nous avons utilisé trois familles de fonte avec trois tailles (10, 12 et 14) et quatre styles: Gras, Italique, Gras-Italique et Simple style.

Le tableau 5-1 présente la taille d'échantillons d'entrainement et de test de la base des images du texte arabe imprimé APTI,

Famille de fonte	Echantillon de test	Echantillon d'apprentissage
ArabicTransparent	36000	107999
Andalus	36000	108000
DecoTypeThuluth	36000	108000
Total	108000	323999

Tableau 4-1 Nombre d'échantillonsde fontes de teste et d'apprentissage utilisé.

Dans ce travail on considère que la fonte est déterminée suivant 3-uplet (famille de Fonte, Taille et Style)

Pour évaluer la performance du système proposé, nous avons utilisé 3000 exemple pour le teste et 9000 d'apprentissage pour chacune de 36 classe de fonte.

Les résultats de classification sont obtenus en utilisant l'algorithme proposé qui permet de calculer la continuité de pixels dans toutes les directions matricielles; en plus de huit paramètres statistiques qui sont détaillés ci-dessous pour caractériser chaque mot de 20 primitives et classifier les données de test.

Trois classificateurs de type KPP avec trois distances différentes:

- KPP1 avec la distance Corrélation.
- KPP2 avec la distance Citybloc
- KPP3 avec la distance Euclidien

Le nombre de voisin K est fixé à 5 suivant des expériences effectuées. La prise de décision est basée sur le vote majoritaire illustré ci-dessus pour les résultats obtenus.

Les configurations matérielles et logiciel utilisés pour mettre en place le système développé et tester la performance des techniques proposées sont les suivant: Dual-Core PC de 2.00 Ghz, RAM de 2,00 Go, les programmes composant le système pour les différentes étapes de notre système sont développés en utilisant le logiciel et langage de programmation Matlab 2010.

Dans notre approche, le processus de reconnaissance de fonte a été divisé en trois étapes: Dans la première étape nous avons fait la reconnaissance par famille de fonte, l'étape suivante est consacrée à la reconnaissance de taille tandis que dans la troisième étape consiste à identifier le style de la fonte.

Dans la suite, on présente les résultats expérimentaux obtenus pour la reconnaissance de la famille de fonte, la reconnaissance de la fonte et la taille, terminons par l'illustration des résultats de la reconnaissance de la famille de fonte, de taille et de style.

Les résultats de la reconnaissance de la famille de fonte et la reconnaissance de la famille de fonte et taille sont comparés avec d'autre expérimentation.

4.8.2.1 Reconnaissance de la famille de fonte

Le tableau 5-2, présente les résultats obtenus pour chacune des trois famille de fonte des mots arabes imprimés de la base APTI, la moyenne de taux de reconnaissance est calculée en utilisant l'algorithme proposé (Continuité de Pixels (CP) plus de huit primitives statistiques (SP)) dans l'étape d'extraction ainsi que le principe de vote majoritaire de trois classificateur de KPPV dans la phase de classification (avec K=5 et trois distances différentes :Citybloc, Euclidien et Corrélation distance).

Famille de Fonte / Taille	ArabicTransparent	Andalus	DecoTypeThuluth
10	99.50	99.29	99.56
12	99.60	99.70	99.72
14	99.55	99.70	99.35
Taux de reconnaissance	99.55	99.57	99.54
Taux moyen de reconnaissance	**99.55**		

Tableau 4-2 Taux de reconnaissance par famille de fonte.

Nous constatons que le meilleur score obtenu est 99,57% obtenu avec la famille de fonte Andalus, les deux autres familles de fonte Arabic Transparent et DecotypeThuluth aussi présente des résultats très encourageant plus proche ou égale à la moyenne générale de taux de reconnaissance qui est 99,55%. Les taux de reconnaissance obtenus expérimentalement, montrent clairement l'efficacité de notre approche basée sur le calcul de la continuité des pixels dans les quatre directions matricielle et les statistiques des histogrammes vertical et horizontal, aussi le calcul de densité des pixels blancs par rapport aux pixels noirs qui sont appliqués sur les dix derniers pixels du mot. la variation de la taille d'écriture de la famille de fonte influence sur le taux de reconnaissance pour le cas de la famille de fonte Andalus avec un taux de 0,41% passant de la taille 10 à la taille 14, pour les deux autres familles de fonte ArabicTransparent et DecotypeThuluth, la variation de taux de reconnaissance est due au changement dans l'aspect calligraphique et structurelle qui ne fournit qu'une quantité minimale dans les dix derniers pixels. Le taux d'erreur de 0,45% est en générale expliqué par la forte similarité entre les familles de fonte, en effet la variation de la taille et/ou style d'une famille de fonte donne une autre fonte avec des caractéristiques structurelles proche d'une autre fonte de famille différente.

Le système de reconnaissance de fonte proposé donne un taux de reconnaissance très encourageant (99,55%), aussi on constat que la performance de la technique d'extraction proposée donne les meilleurs résultats pour la taille 12 de trois familles de fonte étudiées.

Pour comparer les résultats obtenus par notre approche (CP & PS en utilisant multi KPP) pour la reconnaissance de la famille de fonte, on considère la méthode d'extraction Steerable Pyramid (SP) avec 6 directions testée sur les familles de fonte indiquées dans le tableau 4-1 de la base APTI en utilisant le classifieur KPP en premier temps et Le classifieur de Réseau de Neurone avec l'algorithme d'apprentissage Rétro-propagation de gradient [145]. Dans la suite

(Figure 5-23) on présente une comparaison entre les résultats obtenus avec les techniques d'extractions appliquées sur trois familles de fonte.

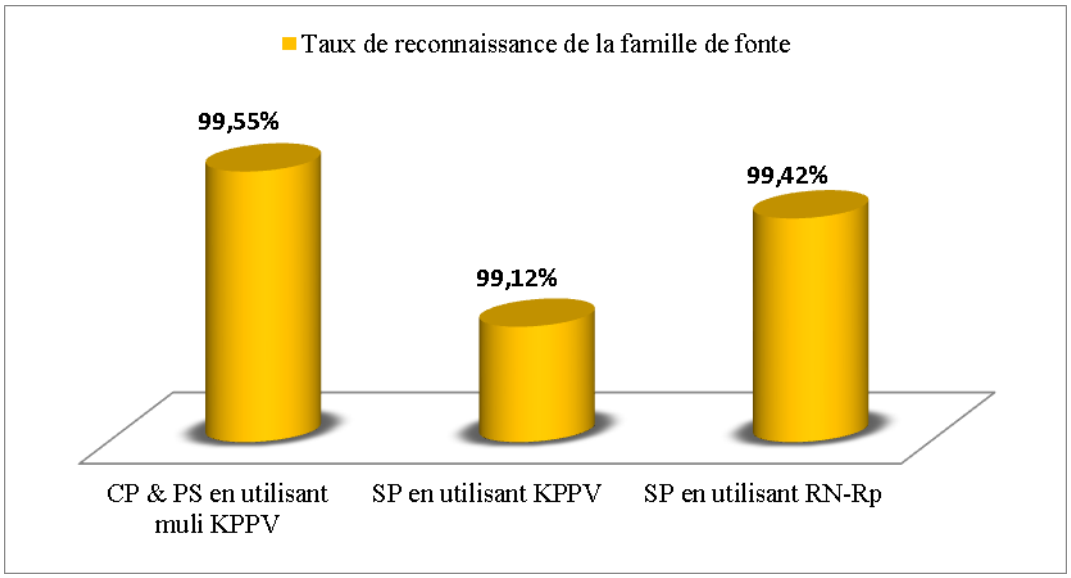


Figure 4-23 Taux de reconnaissance de la famille de fonte avec trois méthodes défirentes.

Les résultats illustrés dans la Figure 5-23 prouvent l'amélioration des résultats de taux de reconnaissance apportée par notre approche vis-à-vis la méthode de Steerable Pyramide de reconnaissance de la famille de fontes, cela est due à la capacité d'identification de la famille de fontes à travers le calcul de la continuité des pixels dans les quatre directions, les statistiques extraits de l'histogramme et le calcul de la densité de pixels dans l'image du mot.

Dans la section qui suit, nous présentons les résultats expérimentaux de la reconnaissance de fonte avec la prise en compte de la taille.

4.8.2.2 Reconnaissance de la famille de fonte et de la taille

Dans cette section, nous rapportons les résultats expérimentaux de la reconnaissance de la famille de fontes avec la prise en considération de la taille. Le tableau 5-3 illustre les résultats obtenus de la reconnaissance pour chacune des trois familles de fontes suivant les tailles 10,12 et 14, en utilisant l'algorithme proposé Continuité de Pixel (CP) en plus de huit primitives statistiques (SP),pour la classification, nous avons adopté trois classificateurs de type K Plus Proche Voisin (KPPV) avec le nombre de voisin K = 5, la prise de décision pour obtenir les résultats finaux est basée sur le processus du vote majoritaire. Les taux de reconnaissance moyen et total sont calculés pour évaluer la performance générale de notre approche.

Famille de fonte / Taille	ArabicTransparent	Andalus	DecoTypeThuluth	Taux moyen	Taux Total
10	99.50	99.29	99.56	99.45	**99.55**
12	99.60	99.70	99.72	99.67	
14	99.55	99.70	99.35	99.53	

Tableau 4-3 Taux de reconnaissance de la famille de fontes suivant la taille

L'analyse des résultats du tableau 5-3, montrent que: le taux de reconnaissance varie d'une valeur qui ne dépasse pas 0,5%, par exemple le cas des deux familles de fonte ArabicTransparent et Andalus. L'analyse du taux moyen, nous montre que notre approche est plus efficace pour la taille 12 pour les trois familles de fonte et particulièrement pour la famille de fonte DecoTypeThuluth avec laquelle nous avons obtenu un taux maximal de 99,72%. La variation de la taille est engendrée avec une variation dans les différents aspects calligraphique, structurelle et morphologique des mots et par la suite sur les dix derniers pixels qui sont utilisé pour extraire 20 primitives. Le taux d'erreur de 0,45% est expliqué par le taux de vraisemblance importante engendrant le changement de la taille, en effet la variation de la taille d'une fonte augmente le taux de similarité avec d'autre fonte.

Pour comparer notre approche de reconnaissance de la taille d'écriture avec d'autre expérience utilisant les mêmes familles de fonte étudiées de la base APTI, on considère les modèles mixtes gaussien (MGMs) détaillés dans [10], testés sur la base APTI, pour estimer la vraisemblance de la famille de fonte et la taille de fonte en respectant les primitives locaux.

La figure 5-24 illustre une comparaison entre notre approche nommée " Continuité de Pixels (CP) & Statistiques Primitives (SP)" en utilisant trois classificateurs de type KPPV avec trois distances distinctes et la méthode de modèles de mélange gaussien (MGM) utilisé dans [10].

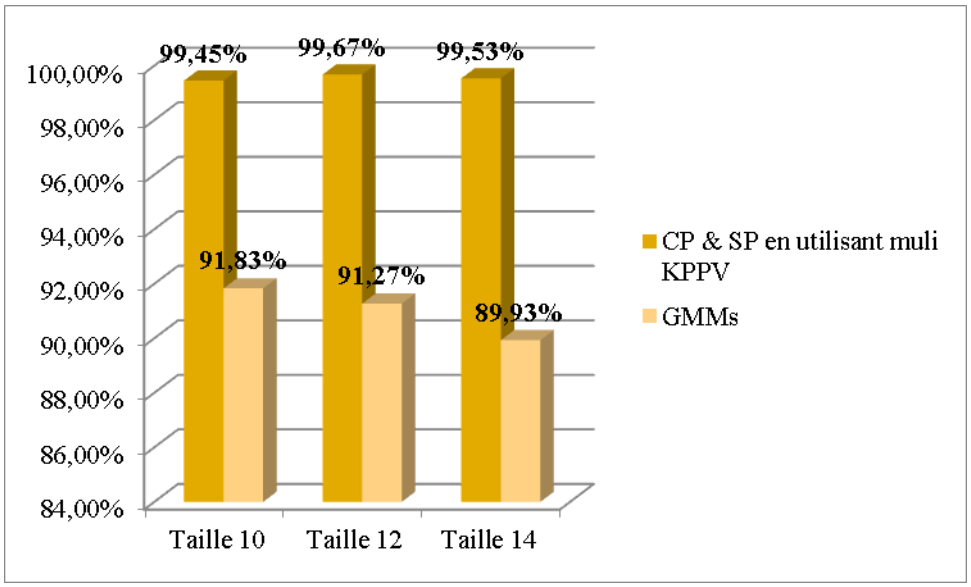


Figure 4-24 Taux de reconnaissance de la fonte & la taille.

Le tableau 5-24 illustre qu'avec la méthode GMMs[10], plus que la taille augmente plus que le taux de reconnaissance diminue, le taux de reconnaissance passe de 91,83%(pour la taille 10) à 89,93% pour la taille 10 avec une baisse de 1,9%, ce qui explique la sensibilité de cette méthode à la variation de la taille d'écriture, par contre pour notre approche le changement de la taille ne change pas le taux de reconnaissance d'une manière significatives, le baisse de taux de reconnaissance au maximum ne dépasse pas 0,08%.

Avec notre approche nous avons réalisé un gain de reconnaissance de 7,62% pour la taille 10, 8,4% de gain pour la taille 12 et 9,6% de gain pour la taille 14,

Le taux de reconnaissance total de la taille des familles de fonte de 99,55% réalisé est très encourageant, le taux d'erreur de 0,45% est améliorable en ajoutant d'autre primitives pertinentes des différentes caractéristiques des fontes étudié. Avec ce travail nous avons contribué à l'amélioration de taux de reconnaissance de la taille d'écriture des mots.

4.8.2.3 Reconnaissance de la famille de fonte, de taille et de style

Dans les deux sections précédentes, la reconnaissance de style de la fonte n'est pas étudiée. L'objectif de cette section est de présenter des expérimentations menées sur trois familles de fonte pour évaluer la performance de notre approche pour la reconnaissance de la famille de fonte, de taille et de style et par la suite identifier exactement les trois propriétés d'une fonte unique.

Le tableau 5-4 illustre le détail des résultats expérimentaux réalisés avec trois familles de fonte (Arabic Transparent, Andalus et DecoTypeThuluth), avec trois tailles de fonte (10,12 et 14), et quatre styles (Simple, Gras, Italique et Italique-Gras). Dans cette expériences 36 classes de fonte (famille de fonte, taille, style) sont considérés,

Font / Taille	Arabic Transparent				Andalus				DecoType Thuluth			
Styles	**Gras**	**Italique**	**Gras-Italique**	**Simple**	**Gras**	**Italique**	**Gras-Italique**	**Simple**	**Gras**	**Italique**	**Gras Italique**	**Simple**
10	99,23	99,36	99,36	99,3	98,83	99,13	97,96	99,16	99,26	99,26	94,46	98,13
12	99,66	99,60	99,50	99,4	99,63	99,50	98,96	99,56	96,20	96,60	91,90	92,46
14	99,80	99,50	97,90	99,33	100	100	99,83	99,76	96,03	92,66	97,46	93,93
Taux par style	99,73	99,55	98,70	99,37	99,42	99,54	98,92	99,49	97,73	96,17	94,61	94,84
Taux moyen	**99,34**				**99,34**				**95,84**			

Tableau 4-4 Taux de Reconnaissance des familles de fonte par styles

Le taux d'erreur moyen de la reconnaissance de style des deux familles de fonte Andalus et Arabic Transparent est de 0,66% et moins que le taux d'erreur de 4,16% pour la famille de fonte DecoType Thuluth, en relation avec la structure et de distribution des pixels dans les dix derniers pixels. La structure des dix derniers pixels du mot ou pseudo mot est sensible au changement de la taille, en effet le changement de la taille implique un changement

global au niveau structurel du mot et par la conséquent sur les dix derniers pixels qui peuvent fournissent des informations insuffisantes pour décrire la nature et les caractéristiques de la fonte.

La figure 5-25, illustre l'influence de changement de la taille sur le taux de reconnaissance des styles des trois familles de fontes étudiés.

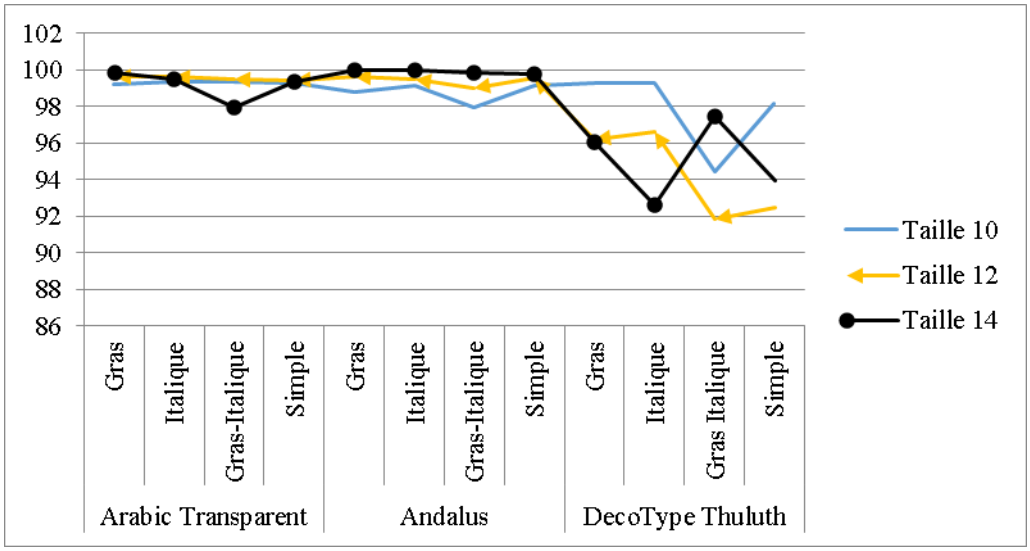


Figure 4-25 Taux de reconnaissance de styles par taille et famille de fonte par multi KPPV.

L'analyse des résultats de reconnaissance de styles illustrés dans la figure 5-23, présente l'efficacité de la méthode développée pour les deux familles de fonte Arabic Transparent et Andalus avec un taux de reconnaissance de 99,34%, le minimum taux de reconnaissance est obtenu avec le style Gras-Italique, ce qui explique la forte vraisemblance morphologique de ce style avec les autres styles surtout pour la même taille.

La complexité morphologique et structurelle de la famille de fonte DecoType Thuluth fournit une quantité d'informations insuffisante dans les dix derniers pixels pour bien identifier les quatre styles.

Le système proposé a réalisé un taux de reconnaissance meilleur pour les deux styles Gras et Italique de la Famille de fonte Andalus avec la taille 14.

L'analyse des résultats de la reconnaissance de style montrent que: le système proposé reconnaît les styles Gras, Italique et Simple plus que le style Italique-Gras, c'est à cause de la fort similitude entre les styles et exactement pour les mêmes tailles, par exemple: La fonte ArabicTransparent de la taille 14 présente plus de 73% de similitude entre les deux styles Italique-Gras et Italique, et plus de 77% de similitude entre les styles Gras et Simple. Le changement de style pour certain fonte n'a pas d'impact important sur l'aspect calligraphie et structurelle de fonte.

4.9 Conclusion

Les expériences ont montrées que les systèmes traitant une seule fonte est plus performant que les systèmes multi-fontes, généralement les documents sont édités en utilisant multi familles de fontes, tailles et de styles, ce qui explique la nécessité d'un système pour identifier la fonte arabe ROFA (Reconnaissance Optique de Fonte Arabe).

Dans ce travail, un système de reconnaissance optique de fonte arabe (ROFA) est proposé et présenté pour décomposer la complexité des systèmes de reconnaissance optique des caractères (OCR).

L'algorithme d'extraction proposé est basé sur la mesure de la distribution continue des pixels dans les quatre directions horizontale, verticale, diagonale et antidiagonale en plus de huit primitives statistiques extraite de l'histogramme vertical et horizontal, la densité des pixels blancs par rapport aux pixels noirs est aussi considérée à fin d'obtenir 20 primitives identifiant la fonte du mot, ces primitives sont extraites des dix derniers pixels du mot. La technique de classification supervisée est adoptée en utilisant trois classificateurs de type K-Plus-Proche-Voisin, la prise de décision est basée sur le processus de vote majoritaire en utilisant les trois résultats issus des trois classificateurs.

Les expérimentations ont été réalisées sur 108000 échantillons des mots en se basant sur 323999 exemples d'apprentissage composant trois tailles 10, 12 et 14 avec quatre styles: Simple, Gras, Italique et Italique-Gras, et trois familles de fonte Arabic Transparent, Andalus et DecoTypeThuluth de la base APTI,

Le taux global de la reconnaissance de la famille de fontes obtenu est 99,55%, il est comparé avec d'autre expérimentations, l'avantage de l'approche proposée est l'adaptation avec les familles de fonte de nature morphologique complexe.

Le taux global de la reconnaissance de la famille de fontes et de la taille est 99,55% ce qu'est très encourageant vis-à-vis d'autres méthodes.

Le taux global de la reconnaissance de la famille de fontes, la taille et le style obtenu est 98,17%.La variation de la taille et/ou style de fonte augmente le taux de similarité avec d'autres fontes et par la suite influence sur la performance générale du système.

Une étude particulière concernant la morphologie de chaque fonte en prenant en considération les styles et les tailles est nécessaire pour concevoir un système performant de reconnaissance multi-fontes en adoptant la reconnaissance de fonte en trois étapes; reconnaissance de la famille de fonte puis la taille et finalement le style.

À la fin de ce travail, on remercie les auteurs de la base de données APTI des mots arabes imprimés d'avoir mettre à notre disposition cette base de données pour tester la performance de l'approche proposée.

Conclusion générale et perspectives

Dons ce travail de thèse, nous nous sommes intéressés à la reconnaissance de l'écriture arabe imprimée. Nous avons présenté les systèmes de reconnaissance en général passant de déférentes phases : prétraitement, extraction classification et la prise de décision. Ensuite, nous avons présenté un état de l'art des méthodes et algorithmes proposés pour résoudre les problèmes des systèmes de reconnaissance telle que les prétraitements, la segmentation l'extraction de caractéristiques et la classification. L'étude de l'état de l'art nous a permis de dégager plusieurs problèmes empêchant la réalisation d'un système de reconnaissance fiable. Pour ce faire, nous avons proposé quatre approches, chacune traite une partie de l'ensemble des problèmes. En effet, nous avons proposé une méthode de segmentation de texte. Une méthode d'extraction des primitives pour la reconnaissance des caractères arabes. Une autre approche a été proposée pour la reconnaissance des mots arabes. La quatrième approche traite le problème de la reconnaissance de fonte.

En effet, nous avons proposé une méthode de segmentation de texte. Dans la première approche, après la phase de prétraitement, le texte est segmenté en paragraphes, chaque paragraphe est segmenté en lignes du texte en utilisant la projection horizontale de l'histogramme, chaque ligne est segmentée en mots en se basant sur la projection verticale de l'histogramme, la segmentation des mots en caractères est réalisée en analysant la projection verticale de l'histogramme et la suppression de la ligne de base, à ce niveau la segmentation exacte de la ligne de base n'est pas toujours efficace surtout pour le cas des fontes dont la ligne de base ne présente pas une valeur maximale au niveau de la projection horizontale de l'histogramme de la ligne de texte. La sur-segmentation et fausse estimation des points de segmentation du mot en caractères sont aussi parmi les problèmes de segmentation rencontrés.

Dans la deuxième approche, nous avons proposé un algorithme discriminant les caractères arabes isolés, basé sur double approche structurelle et statistique. Les résultats obtenus sont encourageants, mais la vraisemblance inter fontes et certains caractères, nous a empêché à avoir un score maximal.

Dans la troisième approche, nous nous somme focalisée sur la réalisation d'un système, de reconnaissance de mots arabes imprimés, pour ce faire, nous avons proposé un algorithme d'extraction des caractéristiques, visant à transformer l'image normalisée à une taille d'une matrice carrée d'ordre 12, pour extraire 144 séquences en se basant sur le parcours zigzag. La modélisation est basée sur la méthode de classification KPPV. Les résultats expérimentaux obtenus sont très encourageante grâce à la normalisation de la taille durant la phase de prétraitement et l'utilisation des images non bruitées.

La quatrième approche s'articule sur la reconnaissance de fonte arabe imprimée, par famille, taille et style. Cette approche a été basée sur le calcul de la longueur des pixels continus dons les quatre directions de la matrice (horizontale, verticale, diagonale

et antidiagonale). Les tests expérimentaux réalisés sur la base des images des mots à basses fréquences APTI, montrent que la forte similarité entre les styles des mêmes fontes est l'un des difficultés rencontrées.

La modalisation des systèmes proposés est basée sur l'utilisation de deux classifier : KPPV et support vecteur machines (SVM) vu ces avantages d'implémentation et performance de prédiction.

Et comme perspectives. La résolution des problèmes suivants serait un apport considérable, tant au niveau simplification de la tache de l'AOCR, qu'aux niveaux validation, portabilité et performance des produits réalisés.

- ✓ Classification des calligraphies des caractères arabes.
- ✓ L'étude approfondie relative à la classification des fontes du point de vue calligraphie et corps.
- ✓ La réalisation des outils tels que les dictionnaires, bases de données et des statistiques se rapportant à l'écriture arabe.
- ✓ L'optimisation des résultats peut être proposée par : le développement des algorithmes performant de prétraitements :
- ✓ Le développement des algorithmes de segmentation spécialisé chacun pour une classe de la famille de la font.

Le développement des minis systèmes de reconnaissance mono fonte (par Famille, taille et style) spécialisés chacune pour une fonte bien déterminée, dont le choix de la méthode de segmentation sera en fonction de la nature de la fonte détectée.

Par ailleurs, l'utilisation des modèles de la combinaison des classificateurs pour bénéficier de la puissance de classification de chacun, notamment les algorithmes de classification montrant des performances généraux, pouvant diminuer le taux d'erreur de notre système et par conséquent avoir un système très performant.

Références

[1] F. Slimane, R. Ingold, S. Kanoun, A. M. Alimi, and J. Hennebert. “A new arabic printed text image database and evaluation protocols”, ICDAR, pages 946–950, 2009.

[2] Zeki, Ahmed M. "The segmentation problem in arabic character recognition the state of the art." Information and Communication Technologies, 2005. ICICT 2005. First International Conference on. IEEE, 2005.

[3] Lamia Hadrich Belguith, Leila Baccour et Ghassan Mourad, “Segmentation de textes arabes basée sur l’analyse contextuelle des signes de ponctuations et de certaines particules”, TALN 2005, Dourdan, 6-10 juin 2005.

[4] Baccour, Leïla, G. Mourad, and L. Belguith Hadrich. "Segmentation de textes arabes en phrases basée sur les signes de ponctuation et les mots connecteurs." troisième journées scientifiques des jeunes chercheurs en génie électrique et informatique (2003).

[5] Touj, Sodien, Najoua Essoukri Ben Amara, and Hamid Amiri. "Reconnaissance de l'écriture arabe imprimée par transformée de Hough Généralisée." Conférence Internationale Francophone sur l'Ecrit et le Document (CIFED 04). 2004.

[6] Parisa Shirvani and Mehrdad Vatankhah Khouzani, “A new method to separation of Farsi and Arabic sub-words using image processing techniques”, Pattern Recognition and Image Analysis (PRIA) 2013pp. 1-3, 2013.

[7] ALGINAHI, Yasser M. A survey on Arabic character segmentation. International Journal on Document Analysis and Recognition (IJDAR), vol. 16, no 2, p. 105-126, 2013.

[8] Al-Muhtaseb, H. A., Mahmoud, S. A. and Qahwaji, R. S. R. Recognition of off-line printed Arabic text using Hidden Markov Models. Signal Processing, Vol. 88, No. 12, pp. 2902-2912 (2008).

[9] Shaaban, Z. "A new recognition scheme for machine-printed arabic texts based on neural networks." Proceedings of World Academy of Science, Engineering and Technology”. Vol. 31. 2008.

[10] Oussama Zayene, Fouad Slimane, “Reconnaissance de l’écriture arabe multi-fonte à très basse résolution”, JJC 2012, pp. 443–448, Bordeaux, 21-23 mars 2012.

[11] Fouad SLIMANE Slim KANOUN Adel M. ALIMI Rolf INGOLD, Jean HENNEBERT, “Gaussien Mixture Models for Arabic Font Recognition”, International Conference on Pattern Recognition page 2166, 2010.

[12] Fouad Slimane, Slim Kanoun, Jean Hennebert, Adel M. Alimi, Rolf Ingold, “A study on font-family and font-size recognition applied to Arabic word images at ultra-low resolution Pattern”, Recognition Letters 34 209–218, (2013)

[13] Belaid, A., and J. C. Anigbogu. "Use of many classifiers for multifont text recognition." Traitement du signal 11 (1994): 57-57.

[14] Nizar Zaghden, Sami Ben Moussa, Adel M Alimi. “Reconnaissance des fontes arabes par l'utilisation des dimensions fractales et des ondelettes”. Laurence Likforman-Sulem., SDN06, pp.277-282,Sep 2006.

[15] Nicholas Journet, Anne Vialard, Jean-Philippe Domenger. « Analyse de fontes anciennes : de la génération de données synthétiques a la reconnaissance ». Colloque International Francophone Sur l'Ecrit et le Document (CIFED2010), Mar 2010, Tunisie. 2010.

[16] Ibrahim Abuhaiba, “Arabic Font Recognition Based on Templates”. The International Arab Journal of Information Technology, Vol. 1, No. 0, July 2003.

[17] Ibrahim S. I. Abuhaiba, “Arabic Font Recognition using Decision Trees Built from Common Words”, Journal of Computing and Information Technology - CIT 13, 3, 211–223.2005.

[18] Hamzah Luqman, Sabri A. Mahmoud, and SamehAwaida, “Arabic and Farsi Font Recognition: Survey”, International Journal of Pattern Recognition and Artificial Intelligence Vol. 29, No. 1 1553002 (23 pages), (2015).

[19] Zaghden, Nizar, Sami Ben Moussa, and Adel M. Alimi. "Reconnaissance des fontes arabes par l'utilisation des dimensions fractales et des ondelettes." *Actes du 9ème* Colloque International Francophone sur l'Ecrit et le Document.SDN06, 2006.

[20] Khosravi, Hossein, and Ehsanollah Kabir. "Farsi font recognition based on Sobel–Roberts features." Pattern Recognition Letters 31.1 75-82, 2010.

[21] Fouad SLIMANE, Rolf INGOLD, Slim KANOUN, Adel M. ALIMI, Jean HENNEBERT, “A New Arabic Printed Text Image Database and Evaluation Protocols”, 10th International Conference on Document Analysis and Recognition 2009.

[22] G.Lorette, Y.Lecourtier, "Reconnaissance et Interprétation de Texts Manuscrits Hors-line: Un Problème d’Analyse de Scène", Bigre Num 80-CNED Colloque National sur l’Ecrit et le Document, Nancy, CNED, Juillet 1992.

[23] E.Lecolinet, O. Barett: « Cursive word récognition: Methods and stratégies». In NATO/ASI, Fundamentals in handwriting récognition, Bonas, France June 21-July 3, 1993.

[24] B. Al-Badr, R.M. Haralick: « Segmentation-free word récognition with application to Arabie ». IEEE. Proc. 3rd International conférence on document analysis and récognition (ICDAR'95), pp. 355-359, Montréal, Canada, 1995.

[25] P.M. Lallican, C. Viarp-Gaudin, S. Knerr: « From off-line to on-line handwriting récognition ». Proc. Workshop on frontiers in handwriting récognition, pp. 303-312, Amsterdam 2000.

[26] A.Belaid, " Reconnaissance Automatique de l'Ecriture et du Document". Pour la science, 2001.

[27] Adel M. Alimi. An evolutionary neuro-fuzzy approach to recognize on-line Arabic handwriting. In ICDAR '97: Proceedings of the4th International Conference on Document Analysis and Recognition, IEEE Computer Society, pp.382–386, 1997.

[28] Neila Mezghani, Amar Mitiche, and Mohamed Cheriet, On-line recognition of handwritten Arabic characters using a kohonen neural network. In IWFHR '02: Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR'02), IEEE Computer Society, page 490, 2002.

[29] Neila Mezghani, Mohamed Cheriet, and Amar Mitiche, Combination of pruned kohonen maps for on-line Arabic characters recognition, In ICDAR '03: Proceedings of the Seventh International Conference on Document Analysis and Recognition, IEEE Computer Society, page 900, 2003.

[30] A. Mitiche N. Mezghani and M. Cheriet, Reconnaissance en-ligne de caractères arabes manuscrits par un réseau de kohonen, In Vision Interface, IEEE Computer Society, pp.86–191, Calgary, Canada, 2002.

[31] Ben Amara N., Belaïd A., Ellouze N., "Utilisation des modèles Markoviens en reconnaissance de l'écriture arabe état de l'art", Colloque International Francophone sur l'Ecrit et le Document (CIFED'00), Lyon, France, pp.181-191, 2000.

[32] A. Amin, A. Kaced, J.P. Haton, and R. Mohr. Handwritten Arabic character recognition by the irac system. In ICPR, pp.729–731, 1980.

[33] Plamondon R., Srihari S. N. "On-Line and Off-Line Handwriting Recognition: A comprehensive survey", IEEE Trans on Pattern Analysis and Machine Intelligence, Vol. 22, N°.1, pp.63-84, 2000.

[34] C.C. Tappert, C.Y. Suen, and T. Wakara, The state of the art in on-line handwriting recognition, IEEE Trans. on Pattern Analysis and Machine Recognition, Vol.12, N°.8, pp.787–808, 1990.

[35] I.R. Tsang: «Pattern recognition and complexe systems». Thèse de doctorat, université d'Antwerpen, 2000.

[36]B. Al-Badr, R.M. Haralick:"Segmentation-free word recognitionwith application to Arabie". IEEE. Proc. 3rd International conférence on document analyses and récognition (ICDAR'95), pp. 355-359, Montréal, Canada, 1995.

[37] A.Ameur, K. Romeo-Packer, H. Miled, and M. Cheriet: “Coupling observation/letter for a Markovianmodelisationapplied totherecognition of Arabie handwriting”. IEEE. Proc. 4* International conference on document analysis and recognition (ICDAR'97), pp. 580-583, Ulm, Germany, 1997.

[38] A.Benouareth. “Reconnaissance de Mots Arabes Manuscrits par Modèles de Markov Cachés à Durée d’Etat Explicite”, Thèse de doctorat, Université Badji Mokhtar -Annaba V, 2007.

[39] E.Lecolinet, O. Barett: “Cursive word recognition: Methods and strategies”. In NATO/ASI, Fundamentals in handwriting recognition, Bonas, France June 21-July 3, 1993.

[40] B. Al-Badr, R.M. Haralick: “Symbol recognition without prior segmentation”. Conference SPIE-EI,1994.

[41] R.G. Casey, E. Lecolinet: “Strategies in character segmentation: A survey”. IEEE.Proc. 3rd international conference on document Analysis and recognition (ICDAR’95), pp. 1028-1033, Montreal, Canada, 1995.

[42] N. Benamara, A. Belaid, N. Ellouze : “Modélisation pseudo bidimensionnelle pour la reconnaissance des chaînes de caractères arabes imprimées”. Proc. 1er Colloque International francophone sur l'écrit et le document (CIFED'98), pp. 131-140, Québec, Canada ,1998.

[43] El-Hajj R., Mokbel C. and Likforman L.,“HMM-based Arabic Cursive Handwritten Recognition System”, The RTST conference (Int’l. Conference on Research Trends in Science and Technology).LAU UniversityBeirut Lebanon, March 2005.

[44] Sriganesh Madhvanath, Venu Krpasundar, and Venu Govindaraju,“Syntactic methodology of pruning large lexicons in cursive script recognition”.Pattern Recognition, Vol.34, N°.1, pp.37–46, 2001.

[45] D. Bouchaffra, V. Govindaraju, and S. N. Srihari, "Postprocessing of recognized strings using nonstationary Markovian models", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.21, pp.990-999, 1999.

[46] Gilloux(M.). – Real-Time Handwritten WordRecognition Within Large Lexicon. Proc. IWFHR5, p. 301-304, Colchester, sept. 1996.

[47] O’Boyle(P.), Owens(M.) et Smith (F.J.). –A weighted average n-gram model of natural language. Computer Speech and Language, 8, pp. 337-349, 1994.

[48] AL Badr B., Mahmoud S.A., “Survey and bibliography of Arabicopticaltext recognition”, Signalprocessing, Vol 41, pp 49-77, 1995.

[49]N.Otsu. “A threshold selection method from grey-level histograms”, IEEE Trans. Syst. Man. Cybern, vol.SMC-8, 1978.

[50] J. Bernsen. Dynamic thresholding of grey-level images.In Proc. Eighth Int ’l Conf. on Pattern Recognition, pp.1251–1255, 1986.

[51] Madhvanath, S., Kim, G., Govindaraju, V, “Chaincode contour processing for handwritten word recognition”.IEEE Trans. Pattern Anal. Mach. Intell. Vol. 21, N°.9, pp.928–932, 1999.

[52] S.N. Srihari & E.J. Keubert.“Integration of handwritten address interpretation technology into the United States postal service remote computer reader system”.ICDAR, pp.892–896, 1997.

[53] MAKKAR, Naazia et SINGH, Sukhjit. A brief tour to various skew detection and correction techniques. International journal for science and emerging Technologies with Latest Trends, 2012, vol. 4, no 1, p. 54-58.

[54] A.Bagdanov, J.Kanai, “Projection Profile Based Skew Estimation Algorithm for JBIG Compressed Images”, International Conference on Document Analysis and Recognition, pp 401-405, 1997.

[55] A.Amin, S.Fisher, T.Parkinson, R.Shiu, “Fast Algorithm for Skew Detection”, SPIE Proceeding, vol 2661, pp.29-30 Janvier, 1996.

[56] S.Bergler, S.Khoury, B.C.Y.Suen, B.Waked, “Skew Detection, Page Segmentation and Script Classification of Printed Document images”, IEEE International Conference on Systems Man and Cybernetics, pp.4470-4475, October 1998.

[57] A.Sehad, L.Mezai, M.T.Laskr, M.Cheriet, "Méthode Rapide et Fiable pour la Détection de l’Angle d’Inclinaison des Documents imprimés par la Régression et la Transformée en Ondelettes", Text Image and Speech Recognition Workshop, pp 211-217, Annaba, Décembre, 2005.

[58] Khorsheed M. S., “Automatic recognition of words in Arabic manuscripts”, PhD thesis, Churchill college, University of Cambridge, England, Also available as University of Cambridge, Computer Laboratory Technical Report N°. 495, June 2000.

[59] Khorsheed M.S., “Recognising handwritten Arabic manuscripts using a single hidden Markov model”, Pattern Recognition Letters, Vol. 24, N°.14, pp. 2235-2242, October 2003.

[60] Ball G., Srihari S. N., Srinivasan H, “Segmentation-free and segmentation dependent approaches to Arabic word spotting”, IWFHR’06, 10th International Workshop on Frontiers in Handwriting Recognition, pp.53-58, La Baule, France, October 2006.

[61] M. Pechwitz, V. Märgner, ”HMM based approach for handwritten Arabic word recognition using the IFN/ENIT– database”, Proceeding of ICDAR’03, 7th International Conference on Document Analysis and Recognition, Vol. 2, pp. 890-894, Edinburgh, Scotland, 2003.

[62] Pechwitz M., Märgner V., El Abed H, “Comparison of two different feature set for off-line recognition of handwritten Arabic words”, Proceedings of IWFHR’06, 10th International Workshop on Frontiers in Handwriting Recognition, pp.109-114, La Baule, France, October 2006.

[63] Märgner V., El Abed H., Pechwitz M., “Off-line handwritten word recognition using HMM- a character approach without explicit segmentation”, Actes CIFED’06, 9ème ColloqueInternational Francophone sur l’Ecrit et le Document, pp. 259-264, Fribourg, Suisse, September2006.

[64] Al-Ma'adeed S., Higgins C., Elliman D.,”A database for Arabic handwritten text recognition research", Proceedings of IWFHR’02, 8th International Workshop on Frontiers in Handwriting Recognition, pp. 485-489, Ontario, Canada, August 2002.

[65] Al-Ma'adeed S., Higgins C., EllimanD.,"Recognition of off-line handwritten Arabic words using hidden Markov model approach", Proceedings of ICPR’02, 16th International Conference on Pattern Recognition, Vol. 3, pp. 481-484, Quebec City, Canada, August 2002.

[66] Al-Ohali Y., “Handwritten Word Recognition – Application to Arabic Cheque Processing”, PhD Thesis, Concordia University, Montreal, Quebec, Canada, February 2002.

[67] Al-Rashaideh H., “Preprocessing phase for Arabic word handwritten recognition”, Information Transmissions in Computer Networks, Vol. 6, N°.1, pp. 11-19, 2006. (Disponible sur la toile www.jip.ru/2006/11-19-2006.pdf).

[68]T.Y. Zhang et C.Y .Suen, “A fast parallel algorithm for thinning digital patterns”, Communications of the ACM, 27(3): 236–240, mars 1984.

[69] D. Arrivault, “Apport des Graphes dans la Reconnaissance Non- Contrainte de Caractères Manuscrits Anciens”, Rapport de thèse, Université de Poitiers, 2006.

[70] B. Al-Badr, S.A.Mahmoud: “Survey and bibliography of Arabic optical text recognition”. Signal processing, vol. 41, pp. 49-77, 1995.

[71] A. Amin, H.B. Al-Sadoun: “A new segmentation technique of Arabic text”. IEEE.Proc. 11th IAPR, pp. 441-445, The Hague, the Netherlands, 1992.

[72] T. Rath, V. Lavrenko and R. Manmatha, “A statistical approach to retrieving historical manuscript images without recognition”, Tech. rep., Center for Intelligent Information Retrieval technical, 2003.

[73] R. El-Hajj, L. Likforman-Sulem, C. Mokbel, "Arabic handwriting recognition using baseline dependent features and Hidden Markov Modeling", ICDAR 05, Seoul, Corée du Sud, 2005.

[74] Al-Ma'adeed S., Higgins C., Elliman D.,"A database for Arabic handwritten text recognition research", Proceedings of IWFHR'02, 8th International Workshop on Frontiers in Handwriting Recognition, pp. 485-489, Ontario, Canada, August 2002.

[75] Al-Ma'adeed S., Higgins C., Elliman D.,"Recognition of off-line handwritten Arabic words using hidden Markov model approach", Proceedings of ICPR'02, 16th International Conference on Pattern Recognition, Vol. 3, pp. 481-484, Quebec City, Canada, August 2002.

[76] Souici L., "Système connexionniste pour la reconnaissance des caractères arabes manuscrits", Mémoire de Magister, Département d'informatique, Université Badji-Mokhtar, Annaba 1996.

[77] A. Webb.Statistical Pattern Recognition, John Wiley & Sons, England, 2002.

[78]S. Theodoridus and K. Koutroumbas. Pattern Recognition, Elsevier Academic Press, United States of America (USA), 2003.

[79] P.Y. Yin, Pattern Recognition Techniques: Technology and Applications, InTech, Vienna-Austria, 2008.

[80] G.X. Ritter and J.N. Wilson."Handbook of Computer Vision: Algorithms in Image Algebra", CRC Press, United States of America (USA),2001.

[81] H. Singh and R.K. Sharma, "Moment in online handwritten character recognition", Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT), India, pp. 225-229, 2007.

[82] EL-Dabi S. S., Ramsis R., Kamel A., "Arabic character recognition system» a statistical approache for recognizing cursive typewritten text", Pattern Recognition, Vol. 23, N° 5, pp 485-495, 1990.

[83] Al-Yousefi H., Udpa S.S., "Recognition of Arabic characters", IEEE Transactions on pattern Analysis and Machine Intelligence, Vol. 14, pp 853-857, Aug 1992.

[84] N.Ben Amara: "Application des PHMMs pour la reconnaissance de l'écriture arabe imprimée". 1ères Journées Scientifiques et techniques (JST FrANCIL) pp.389-392, Avingnon,

[85] A. Bultheel,"Wavelets with Applications in Signal and Image Processing", 2002.

[86] P. Melin and O. Castillo. "Hybrid Intelligent Systems for Pattern Recognition using Soft Computing"; An Evolutionary Approach for Neural Networks and Fuzzy Systems, Springer, Germany, 2005.

[87] Mahmoudi S., "Arabic Caracter recognition using Fourier descriptors and contour encoding", Pattern Recognition, Vol. 27, N°. 6, pp. 815-824, 1994.

[88] A. Amin, J.F. Mari: “Machine recognition and correction of printed Arabic text”. IEEE Transaction on systems, man, and cybernetics, vol. 19, n°5, pp. 1300-1304, September October 1989.

[89]Pal, U., Belaïd, A., and Choisy, C. Water reservoir based approach for touching numeral segmentation. International Conference on Document Analysis and Recognition, 892, 2001.

[90] Kapoor, R., Bagai, D., and Kamal, T. Representation and extraction of nodal features of DevNagri letters.ICVGIP, 2003.

[91] El-Hajj R., Mokbel C. and Likforman L., "HMM-based Arabic Cursive Handwritten Recognition System", The RTST conference (Int’l. Conference on Research Trends in Science and Technology). LAU University Beirut Lebanon, March 2005.

[92] El-Hajj R., Mokbel C., Likforman L., "Reconnaissance de l’écriture Arabe cursive: Combinaison de classifieurs MMCs à fenêtres orientées", actes de CIFED 2006, pp: 271 – 276, Fribourg Suisse, 2006.

[93] R. Al-Hajj. Reconnaissance hors ligne detextes manuscrits cursifs par l’utilisation de systèmes hybrides et de techniques d’apprentissage automatique. Thèse de Doctorat, Ecole Nationale Supérieure de Télécommunications, Paris, 2007.

[94] H. miled, c. olivier, m. cheriet, k. romeo-pakker, ”une méthode rapide de reconnaissance de l'écriture arabe manuscrite” ,seizième colloque gretsi — 15-19— grenobleseptembre.1997

[95] Fouad Slimane, Slim Kanoun, Jean Hennebert, Adel M. Alimi et Rolf Ingold, “Modèles de Markov Cachés et Modèle de Longueur pour la Reconnaissance de l’Ecriture Arabe à Basse Résolution”, MajecSTIC 2009 Avignon, France, du 16 au 18 novembre 2009.

[96] Blumenstein M., Cheng C. K., Liu X. Y., "New Preprocessing techniques for Handwritten Word Recognition", Proc. of the 2nd IASTED Conf. on Visualization, Imaging and Image Processing , pp. 480-484, 2002.

[97] Amin A., "Recognition of hand-printed characters based on structural description and inductive logic programming", Pattern Recognition Letters, Vol. 24, pp. 3187-3196, 2003.

[98] Azizi N., Sari T., Souici-Meslati L., Sellami. M., "Une architecture de combinaison floue de classifieurs neuronaux pour la reconnaissance de mots arabes manuscrits", CIFED’02, 7ème Colloque International Francophone sur l’Ecrit et le Document, pp. 89-96, Hammamet, Tunisie, Octobre 2002.

[99] Klassen T. J., "Towards the on-line recognition of Arabic characters", Proceedings of IJCNN’02, International Joint Conference on Neural Networks, pp. 1900-1905, Honolulu, Hawaii, USA, May 2002.

[100] Sari T., Sellami M., "Cursive Arabic script segmentation and recognition system", International Journal of Computers and Applications, Vol. 27, N°. 3, 2005.

[101] Farah N., Souici L., Sellami M., "Classifiers combination and syntax analysis for Arabic literal amount recognition", Engineering Applications of Artificial Intelligence, Vol. 19, N°. 1, pp. 29-39, February 2006.

[102] Al-Ma'adeed S., "Recognition of off-linehandwritten Arabic words using neural network", Proceeding of GMAI'06, International Conference on Geometric Modeling and Imaging, pp. 141-114, London, England, July 2006.

[103] Souici-Meslati L, "Reconnaissance des mots arabes manuscrits par integration neuro symbolique", Thèse de Doctorat d'Etat, Labo. LRI, Département d'informatique, Université d'Annaba, Algérie, Février 2006.

[104] Zermi N., Ramdani M., Bedda M., "Arabic handwriting word recognition based on hybride HMM/ANN approach", International Journal of Soft Computing, Vol. 2, N°. 1, pp. 5-10, 2007.

[105] Torres Moreno, Juan Manuel. Apprentissage et généralisation par des réseaux de neurones: étude des nouveaux algorithmes constructifs. Diss. 1997.

[106] Al-Muhtaseb, H. A., Mahmoud, S. A. and Qahwaji, R. S. R. Recognition of off-line printed Arabic text using Hidden Markov Models. Signal Processing, Vol. 88, No. 12, pp. 2902-2912 (2008).

[107] Dreyfus, Gérard, et al. "Réseaux de neurones." Méthodologie et applications. Eyrolles, Paris 1 (2002).

[108] Rivals, Isabelle, et al. "Modélisation, classification et commande par réseaux de neurones: principes fondamentaux, méthodologie de conception et illustrations industrielles." Les réseaux de neurones pour la modélisation et la commande de procédés, JP Corriou, ed.(Lavoisier Tec & Doc, 1995) (1995).

[109] Dreyfus, Gérard, et al. "Réseaux de neurones." Méthodologie et applications. Eyrolles, Paris 1 (2002).

[110] Comon, P. "Classification supervisée par réseaux multicouches." Traitement du signal 8.6 (1991): 387-407.

[111] C. J. C. Burges."A tutorial on support vector machines for pattern recognition".KnowledgeDiscovery and Data Mining.2(2), 1–43, 1998.

[112] N. Cristianini and J. Shawe-Taylor."An Introduction to Support Vector Machines andOther Kernel-Based Learning Methods".Cambridge University Press, 2000.

[113] Vapnik V., "The nature of statistical learning theory", Springer,New Work, 1995.

[114] Joachims, Thorsten. Making large-scale SVM learning practical. No. 1998, 28. Technical Report, SFB 475: Komplexitätsreduktion in Multivariaten Datenstrukturen, Universität Dortmund, 1998.

[115] Platt, John C., Nello Cristianini, and John Shawe-Taylor. "Large margin DAGs for multiclass classification." Proceedings of the 12th International Conference on Neural Information Processing Systems. MIT press, 1999.

[116] Fouad SLIMANE, Rolf INGOLD, Slim KANOUN, Adel M. ALIMI, Jean HENNEBERT, "A New Arabic Printed Text Image Database and Evaluation Protocols", 10th International Conference on Document Analysis and Recognition 2009.

[117] ABDELWAHAB ZRAMDINI, ROLF INGOLD, "Optical font recognition from projection profiles", ELECTRONIC PUBLISHING, VOL. 6(3), 249–260 (SEPTEMBER 1993).

[118] Rocha J, Pavlidis T. "Character Recognition without segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, pp. 903-909, 1995.

[119] Alsallakh B., Safadi H., "AraPen: an Arabic online handwriting recognition system", Proceeding of ICTTA'06, 2nd IEEE International Conference on Information & Communication Technologies: from Theory to Applications, Vol. 1, pp. 1844-1849, Damascus, Syria, April 2006.

[120] Baghshah M. S., Shouraki S. B., Kasaei S., "A novel fuzzy approach to recognition of online Persian handwriting", Proceedings of the ISDA'05, 5th International Conference on Intelligent Systems Design and Applications, pp.268-273, Wroclaw, Poland, September 2005.

[121] N.Otsu. "A threshold selection method from grey-level histograms", IEEE Trans. Syst. Man. Cybern, vol.SMC-8, 1978.

[122] A. Vinciarelli and J. Luettin.A new normalization technique for cursive handwritten words.Pattern Recognition Letters, Vol. 22, N°.9, pp.1043–1050, 2001.

[123] M. Eden and M. Hall, "The characterization ofcursives writing", proc.4thsymp. Informatics Theory, London 1961, pp: 287-299.

[124] M Fakir, C Sodeyama. "Machine recognition of Arabic printed scripts by dynamic programming matching", Transaction on InformaticsSystems, 76 (2): 235-242, 1993.

[125] M.S.xKhorsheed "Offline recognition of Omni-font Arabic text using the HMM Toolkit (HTK)", Pattern Recognition Letters 281563–1571, (2007).

[126] LAWGALI, Ahmed, BOURIDANE, Ahmed, ANGELOVA, Maia, et al. Handwritten Arabic character recognition: Which feature extraction method?. International Journal of Advanced Science and Technology, 2011, vol. 34, p. 1-8.Sept. 2011

[127] MILED, H., OLIVIER, C., CHERIET, M., et al. Une méthode rapide de reconnaissance de l'écriture arabe manuscrite. In : 16° Colloque sur le traitement du signal et des images, FRA, 1997. GRETSI, Groupe d'Etudes du Traitement du Signal et des Images, 1997.

[128] MILED, H., OLIVIER, C., CHERIET, M., et al. Une méthode rapide de reconnaissance de l'écriture arabe manuscrite. In : 16° Colloque sur le traitement du signal et des images, FRA, 1997. GRETSI, Groupe d'Etudes du Traitement du Signal et des Images, 1997.

[129] Eshmawi, Ala Abdulmajid. The roving proxy for SMS spam and phishing detection.Diss. Southern Methodist University, 2015.

[130] MENASRI, Farès, VINCENT, Nicole, AUGUSTIN, Emmanuel, et al. Un système de reconnaissance de mots arabes manuscrits hors-ligne sans signes diacritiques. In : Colloque International Francophone sur l'Ecrit et le Document. Groupe de Recherche en Communication Ecrite, 2008. p. 121-126.

[131] ABANDAH, Gheith et ANSSARI, Nasser. Novel moment features extraction for recognizing handwritten Arabic letters. Journal of Computer Science, 2009, vol. 5, no 3, p. 226.

[132] Ouarda hachour, "Segmentation and Recognition of Arabic Printed Script"; JEP-TALN 2004 Traitement Automatique de l'Arabe, Fès, 20 Avril 2004.

[133] Kong, T. Yung and AzrielRosenfeld,"Topological Algorithms for Digital Image Processing", Elsevier Science, Inc., 1996.

[134] Pratt, William K., Digital Image Processing, John Wiley & Sons, Inc., 1991.

[135] S. Nouri, M. Fakir, "Segmentation and Recognition of Arabic Printed Script", IAES International Journal of Artificial Intelligence (IJ-AI) Vol.2, No.1, pp 20~26 ISSN: 2252-8938. , March 2013.

[136] B.M.F.Bushofa and M.Spann,"Segmentation and Recognition of Printed Arabic Characters ", BMVC, doi:10.5244/C.9.54, 1995.

[137] Najoua Essoukri Ben Amara, Sami Gazza "Une approche d'identification des fontes arabes " .CIFED, (8):21–25, 2004.

[138] AL-MUHTASEB, Husni A., MAHMOUD, Sabri A., et QAHWAJI, Rami S. Recognition of off-line printed Arabic text using Hidden Markov Models. Signal processing, 2008, vol. 88, no 12, p. 2902-2912.

[139] Ibrahim S. I. Abuhaiba, "Arabic Font Recognition using Decision Trees Built from Common Words", Journal of Computing and Information Technology - CIT 13, 3, 211–223, 2005.

[140] Fouad Slimane, Slim Kanoun, Adel M. Alimi, Rolf Ingold, Jean HENNEBERT "Gaussian Mixture Models for Arabic Font Recognition", International Conference on Pattern Recognition DOI 10.1109/ICPR, 532,2010.

[141] I.S.I. Abuhaiba, "Arabic Font Recognition Based on Templates", the International Arab Journal of Information Technology, 1, pp. 33–39, 2003.

[142] A. Belaid et J.C .Anigbougu, "Use of Many Classifiers for Multifont Text Recognition" Traitement du Signal - Volume 11 - n ° 1,1994.

[143] Kong, T. Yung and Azriel Rosenfeld, "Topological Algorithms for Digital Image Processing", Elsevier Science, Inc., 1996

[144] Rosenfeld, Azriel and John Pfaltz, "Sequential operations in digital picture processing", Journal of the Association for Computing Machinery, Vol. 13, No. 4, , pp. 471-494,1966.

[145] Faten Kallel Jaiem,Slim Kanoun and Veronique Eglin Arabic font recognition based on a texture analysis 2014 14th International Conference on Frontiers in Handwriting Recognition.page 673-677 DOI 10.1109/ICFHR..118, 2014.

[146] Grandidier F., Sabourin R., Suen C.Y., and Gilloux M., "Une nouvelle stratégie pour l'amélioration des jeux de primitives d'un système de reconnaissance de l'écriture", Colloque International Francophone sur l'Écrit et le Document, (CIFED'2000), pp. 111-120, Lyon, France, July2000.

[147] Oliveira L.S., Benahmed N., Sabourin R., Bortolozzi F, and Suen C.Y., "Feature Subset Selection Using Genetic Algorithm for Handwritten Digit Recognition", 14th Brazilian Symposium on Computer Graphics and Image Processing, (SIBGRAPI'2001), pp. 362-369, , Brazil,October 2001.

[148] Britto, A.S., Sabourin R., Bortolozzi F. AndSuen C.Y., "Foreground and Background Information in an HMM-Based Method for Recognition of Isolated Characters and Numeral Strings", 9the International Workshop on Frontiers in Handwriting Recognition (IWFHR-9), pp.371-376, Tokyo, Japan, October, 2004.