



HAL
open science

Contributions à la caractérisation et la reconnaissance de l'écriture arabe dans ses différentes formes

Afef Echi Kacem

► **To cite this version:**

Afef Echi Kacem. Contributions à la caractérisation et la reconnaissance de l'écriture arabe dans ses différentes formes. Informatique [cs]. UNIVERSITÉ DE MANOUBA ÉCOLE NATIONALE DES SCIENCES DE L'INFORMATIQUE, 2017. tel-02103860

HAL Id: tel-02103860

<https://hal.science/tel-02103860>

Submitted on 18 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

RÉPUBLIQUE TUNISIENNE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
UNIVERSITÉ DE MANOUBA
ÉCOLE NATIONALE DES SCIENCES DE L'INFORMATIQUE



RAPPORT DE SYNTHÈSE

Présenté pour l'obtention du diplôme

HABILITATION UNIVERSITAIRE

Spécialité

INFORMATIQUE

Contributions à la caractérisation et la reconnaissance de l'écriture arabe dans ses différentes formes

Par

AFFEF KACEM ECHI

(Docteur en Informatique)

(Maître Assistante à l'École Nationale Supérieure des Ingénieurs de Tunis)

Réalisé au sein de

LATICE : LABORATOIRE DE RECHERCHE EN TECHNOLOGIES DE L'INFORMATION
ET DE LA COMMUNICATION & GÉNIE ÉLECTRIQUE

Soutenu publiquement le 20 Avril 2017 devant le jury composé de :

Président :	Professeur Henda Hajjami Benghezala	Université de Manouba, Tunisie
Rapporteur :	Professeur Jean-Paul Haton	Université de Lorraine, France
Rapporteur :	Professeur Abdelmajid Benhamadou	Université de Sfax, Tunisie
Examineur :	Professeur Abdelwaheb Belaïd	Université de Lorraine, France
Examineur :	Professeur Najoua Essoukri Benamara	Université de Sousse, Tunisie

Dédicace

Je dédie ce modeste travail à :

- À mon très cher père Hamed, à qui je dois tout, qui n'a jamais cessé de m'encourager et de me soutenir moralement et financièrement depuis le début de mes études. Aucun mot ne saurait exprimer mon profond attachement et ma reconnaissance pour l'amour et la tendresse dont tu m'as toujours entourée.
- À ma très chère mère Nélia, qu'elle soit toujours fière de moi. Elle est pour moi la lumière qui me guide et qui m'emmène aux chemins de la réussite. Tu es toujours mon secours dans les moments les plus difficiles. Que dieu me la garde.
- À mon très cher mari Issam. J'aimerais bien qu'il trouve dans ce travail l'expression de mes sentiments de reconnaissance les plus sincères pour sa patience souvent sollicitée, son soutien et son constant encouragement. Que dieu le tout puissant nous accorde un avenir meilleur.
- À mon adorable ange et trésor Yassine. Tu resteras pour toujours le rayon du soleil qui égaye ma vie.
- À mes frères et soeurs et toute ma belle famille pour leur soutien constant et gentillette.
- À toutes les personnes qui ont servi pour ma formation, mon éducation et mon enseignement.

Afef

Remerciements

Ce rapport d'habilitation est le résultat d'un cheminement émotif et académique, dans lequel plusieurs personnes ont joué un rôle crucial. Je m'acquitte donc ici d'un devoir agréable, celui de souligner ma lourde créance à leur égard.

Je remercie très chaleureusement Madame Henda Hajjami Benghezala, Professeur à l'université de Manouba, qui a accepté de présider ce jury. C'est un grand honneur qu'elle me fait.

Un grand merci à Monsieur Jean-Paul Haton, Professeur à l'université de Lorraine, Monsieur Abdelmajid Benhamadou, Professeur à l'université de Sfax et Madame Najoua Essoukri Benamara, Professeur à l'université de Sousse, qui ont accepté d'être respectivement les deux rapporteurs et l'examineur de mon habilitation, connaissant la qualité scientifique de leurs travaux et les éclairages multiples qu'ils sont susceptibles de donner sur mes propres recherches.

Je tiens à présenter mes sincères remerciements et ma profonde gratitude à Monsieur Abdel Belaïd, Professeur à l'université de Lorraine, d'abord pour m'avoir fait découvrir un domaine passionnant et plein de perspectives depuis qu'il était superviseur de ma thèse, ensuite pour la confiance qu'il a placée en ma personne, pour ses conseils et orientations, enfin pour nos discussions diverses et variées et nos collaborations passées et futures.

Mes plus vifs remerciements s'adressent, également à Madame Souad Chebbi et Monsieur Mohammed Jemni, Professeurs à l'université de Tunis, pour m'avoir bien accueillie au sein du laboratoire LaTICE et Monsieur Mourad Elloumi, Professeur à l'université de Tunis-El Manar pour la confiance qu'ils m'ont accordée en acceptant d'être les directeurs de thèses des travaux que j'ai encadrés.

Des remerciements particuliers vont à mes amis et collègues avec qui je partage les problèmes et les plaisirs quotidiens, pour leur soutien moral afin d'aboutir à mon habilitation universitaire.

Je tiens à remercier chaleureusement les étudiants que j'ai encadrés en mastère et ceux que j'ai co-encadrés en thèse et qui ont contribué aux travaux de recherche présentés dans ce rapport.

Finalement, mes remerciements vont à toute personne ayant contribué de près ou de loin à l'aboutissement de ce modeste travail.

Afef

Résumé

Les travaux présentés dans ce rapport portent sur le choix de descripteurs et classifieurs spécifiques pour caractériser et reconnaître l'écriture arabe en mode hors ligne. Différents types d'écritures ont été considérées allant du linéaire au delà du bidimensionnel. Nous nous sommes fixés des défis aussi bien au niveau de l'écriture : manuscrite, non contrainte, multi-scripteurs, multi-fontes, bidimensionnelle, etc. qu'au niveau du vocabulaire : spécialisé, large et celui du document : manuscrits non contraints.

Pour chaque type d'écriture, nous avons à la fois considéré les descripteurs les plus adéquats : des primitives contextuelles pour pallier la variation de la morphologie du mot, sinon des descripteurs à base de forme comme ceux issus de l'histogramme des orientations de gradients pour identifier le script et sa nature. Un système complet de discrimination entre mots arabes et latins, imprimés et manuscrits, a été proposé, testé et comparé sur une base commune de 24000 mots. Un taux d'identification du script de 99.85% est atteint.

Nous avons également utilisé des descripteurs issus de transformations globales comme la transformée généralisée de Hough ainsi que des descripteurs de contexte de forme comme les histogrammes log-polaires déduits du contour pour repérer les mots de liaison et segmenter les zones de connections dans des anciens manuscrits arabes. Nous avons proposé un système de segmentation capable de séparer entre les lignes ou mots inter-connectés à 94% des cas.

Plusieurs types de classifieurs ont été testés et utilisés dans des contextes différents. Pour traiter un vocabulaire large de mots arabes décomposables, issus de racines triconsonantiques saines, nous avons proposé une combinaison de trois réseaux de neurones transparents, chacun spécialisé dans une décomposition morphologique particulière du mot (racine, schème, éléments de conjugaison) en tenant compte des aspects linguistiques de la langue arabe. L'intégration de connaissances linguistiques dans le processus de la reconnaissance ainsi que notre vision du mot ont permis d'accroître le taux de reconnaissance à 95.33% pour un vocabulaire de taille 5757 mots arabes décomposables (soit un corpus de 17271 échantillons).

Pour reconnaître les mots arabes manuscrits, nous avons utilisé des modèles graphiques probabilistes dont les modèles de Markov cachés linéaires et planaires et les réseaux bayésiens réputés par leur puissance dans plusieurs problèmes de reconnaissance qui nécessitent un raisonnement dans l'incertitude. Nous avons proposé de construire des modèles 2D en couplant dans un réseau bayésien dynamique deux réseaux de type modèle de Markov cachés, obtenus respectivement à partir de descripteurs extraits des lignes et des colonnes des mots. Il en résulte une représentation plus fine d'image du mot, due aux regroupements des informations concernant les lignes et les colonnes. Les résultats ont montré qu'un taux de 90.42% pour la reconnaissance de mots arabes manuscrits, issus

de la base IFN-ENIT, est atteignable.

Enfin, pour les aspects bidimensionnels, nous avons proposé un analyseur syntaxique ascendant descendant pour les formules mathématiques arabes. Il s'appuie sur les opérateurs dominants avec décomposition en sous-expressions suivant une grammaire de coordonnées qui tient compte du contexte pour résoudre les cas de confusions. Cela nous a permis d'afficher un taux de reconnaissance de l'ordre de 91% pour les formules imprimés.

Mots clés :

Segmentation, Extraction et sélection de descripteurs, Classification, Reconnaissance, Vocabulaire large, Identification du script, formules mathématiques.

Abstract

The work, presented in this report, concerns the choice of specific descriptors and classifiers to characterize and recognize the Arabic script in off-line mode. Different types of writings were considered from the linear beyond dimensional. We have set challenges in writing (handwritten, unconstrained, multi-writers, multi-fonts, bi-dimensional, etc.), in vocabulary (specialized, large) and in document (unconstrained handwritten).

For each type of writing, we have considered both the most appropriate descriptors : contextual primitives to compensate the word morphology variation, fuzzy logic to apprehend the writing uncertainty, otherwise shape descriptors as those derived from the histogram of oriented gradients to identify the script and its nature. A fully system which discriminates between machine-printed and handwritten words, written in Arabic and Latin scripts has been proposed, tested and compared using a common base (24000 words). An identification rate of 99.85% is reached.

We also used descriptors from global transformations such as Hough generalized transform and shape context descriptors (notably log-polar histogram deducted from contour image) to respectively spot separator words and segment connection zones in ancient Arabic manuscripts. We have proposed a segmentation system able to separate between interconnected text-lines and words in 94% of cases.

Several types of classifiers have been also tested and used in different contexts. To recognize a large vocabulary of Arabic decomposable words, derived from healthy tri-consonantal roots, we proposed a combination of three neural transparent networks, each specialized in a particular morphological decomposition of the word (root, scheme and conjugation elements), taking into account the linguistic aspects of the Arabic language. The integration of language knowledge in the recognition process and our “human” vision of the word allowed us to increase the recognition rate to 95.33% for a vocabulary of size 5757 (a corpus of 17271 word samples).

To recognize handwritten Arabic words, we used probabilistic graphical models : linear and planar hidden Markov models and Bayesian networks, known by their power in several recognition problems that require reasoning under uncertainty. We have proposed to build independent and coupled models. The independent models are hidden Markov model-based architectures whose observable outputs are respectively features extracted from word image columns and rows. The coupled models combine these hidden Markov models architectures into single dynamic Bayesian network. The objective is to consider all possible correlations between parts of word. The results showed that a recognition rate of 90.42%, using IFN-ENIT database, is achievable.

Finally, for two-dimensional aspects, we proposed a symbol recognizer, an ascending and descending parser for Arabic mathematical formula recognition. It is based on the dominant operators with decomposition into sub-expressions according to a coordinate

grammar that takes into account the context to resolve confusion cases. This allowed us to post a recognition rate of 91% for machine-printed formulas.

Keywords :

Segmentation, Descriptor extraction and selection, Classification, Recognition, Large vocabulary, Script identification, Mathematical formulas.

Table des matières

1	Introduction générale	1
2	Contributions en reconnaissance de l'écriture arabe	7
2.1	Introduction	7
2.2	Contribution 1 : Reconnaissance d'un vocabulaire de mots arabes de grande taille	8
2.2.1	Analyse linguistique de l'Arabe	9
2.2.2	Approche proposée, basée sur l'intégration de connaissances linguistiques	10
2.2.3	Système proposé, basé sur la collaboration de classifieurs neuro-linguistiques	11
2.2.4	Expérimentations	16
2.3	Contribution 2 : Reconnaissance de mots arabes manuscrits	19
2.3.1	Etat de l'art	20
2.3.2	Approche proposée, basée sur les modèles graphiques probabilistes	20
2.3.3	Expérimentations	25
2.4	Contribution 3 : Reconnaissance de formules mathématiques arabes	26
2.4.1	Formule mathématique arabe : spécificités et difficultés	27
2.4.2	Etat de l'art	29
2.4.3	Système proposé, dirigé par la syntaxe	30
2.4.4	Expérimentations	36
2.5	Conclusion	38
3	Contribution en identification du script et de sa nature	39
3.1	Introduction	39
3.2	Similarités et différences entre l'Arabe et le Latin	40
3.3	Etat de l'art	42

TABLE DES MATIÈRES

3.4	Approche proposée	43
3.4.1	Etude de descripteurs	43
3.4.2	Méthode des longueurs de plages de niveaux de gris	49
3.4.3	Descripteurs à base d’histogramme des orientations de gradients	51
3.4.4	Expérimentations	57
3.5	Conclusion	59
4	Contribution en segmentation d’anciens manuscrits arabes	61
4.1	Introduction	61
4.2	Etat de l’art	62
4.3	Système proposé	63
4.3.1	Extraction des zones de connections	64
4.3.2	Segmentation des zones de connection	66
4.3.3	Segmentation des lignes en mots	73
4.4	Conclusion	76
5	Conclusion générale et perspectives	79

Table des figures

2.1	Modèle perceptif, inspiré du système de vision humaine [12].	9
2.2	Notre vision du mot en Arabe.	11
2.3	Architecture du système proposé.	12
2.4	Description en primitives structurales du mot متاجرون.	13
2.5	Caractéristiques globales d'une image de mot	13
2.6	Apprentissage par RNT_R d'une même racine à partir de mots différents.	14
2.7	Apprentissage, par RNT_S, d'un même schème à partir de mots différents.	14
2.8	Apprentissage, par RNT_C, des mêmes éléments de conjugaison à partir de mots différents.	15
2.9	Exemple de combinaison pour la reconnaissance du mot بعادين.	17
2.10	Ligne de base du mot entier.	21
2.11	(a) Nombre de transitions, (b) Structure du MMC-H (Etape d'initialisation).	22
2.12	(a) Hampes, jambages, boucles et diacritiques, (b) PAWs, (c) Structure du MMC-V (Etape d'initialisation).	22
2.13	Structure proposée pour Bayes naïf.	23
2.14	Structure proposée pour Bayes naïf augmenté par un arbre.	23
2.15	De gauche à droite : RBD0, RBD1 et RBD2.	24
2.16	Symboles littéraires et en miroir.	28
2.17	Larges opérateurs étirés	29
2.18	Formule manuscrite avec de larges opérateurs étirés.	29
2.19	Une même formule écrite par différentes personnes.	29
2.20	Formules avec (a) symboles qui se touchent, (b) symboles brisés.	29
2.21	Style oriental où on utilise des alphanumériques locaux non seulement pour les nombre et les variables mais aussi pour les noms de fonction et les opérateurs mathématiques.	31
2.22	Architecture du système proposé.	31
2.23	Vecteur des caractéristiques du nom de la fonction limite.	32

TABLE DES FIGURES

2.24	Exemple d'analyse syntaxique d'une formule d'intégrale.	37
2.25	Codage MathML retourné par notre système vs. vérité de terrain.	37
3.1	Types d'écritures arabe et latine.	40
3.2	Lignes d'écriture.	40
3.3	Bande centrale.	40
3.4	Régularités et singularités par rapport à la ligne de base.	41
3.5	Le même mot écrit par différents scripteurs	41
3.6	PAWs en écriture arabe.	42
3.7	Ligatures horizontales et verticales des lettres en mots arabes.	42
3.8	Elongations de l'écriture.	42
3.9	Descripteurs issus du profil de la ligne de base.	44
3.10	Profil haut/bas du mot arabe imprimé : ميزانية.	46
3.11	Positions des points diacritiques.	47
3.12	Positions des boucles.	47
3.13	Formes des jambages.	47
3.14	Différences d'orientation dans les écritures arabe et latine de natures imprimée et manuscrite.	49
3.15	Représentation d'une image par les longueurs de plages de pixels noirs.	50
3.16	Calcul du descripteur HOG.	51
3.17	Calcul de l'image gradient.	52
3.18	HOGs, selon trois bandes horizontales égales d'un (a) Mot arabe imprimé, (b) Mot latin imprimé, (c) Mot arabe manuscrit et (d) Mot latin manuscrit.	53
3.19	Division de la bande centrale en 5 cellules.	53
3.20	(a) 6 colonnes \times 1 ligne, (b) 3 colonnes \times 3 lignes, (c) 2 colonnes \times 3 lignes et (d) 1 colonne \times 6 lignes.	53
3.21	Disposition de 6 cellules en chevauchement.	54
3.22	Illustration de PHOG à chaque niveau de résolution.	54
3.23	Descripteurs HOG à 8 barres d'images de mots manuscrits.	55
3.24	(a) la distance qui peut être utilisée dans CO-MOG, (b) co-occurrence d'images de mots à une distance donnée, (c) vectorisation et concaténation des matrices de co-occurrences pour former le descripteur CO-MOG [53].	56
3.25	(a) Du pixel central ; pixel 1 représente 0° à $d = 1$; pixel 2 représente 45° ; pixel 3 représente 90° et pixel 4 représente 135° à $d = 1$; (b) Matrice CO-MOG calculée à $d = 2, \theta = 0^\circ$ pour un voisinage 8 à 8 autour du pixel en question.	57
3.26	Quelques cas de confusion observés.	58
4.1	Extrait d'une page de registre de dénombrement du 19ème siècle.	62

TABLE DES FIGURES

4.2	(a) Simple touche, (b) Multi-touche, (c) Chevauchement, (d) Mots horizontalement interconnectés.	63
4.3	Extraction des lignes de base.	64
4.4	Extraction des connexions verticales.	65
4.5	Extraction des connexions horizontales.	65
4.6	(a) Contexte de forme d'un point d'une première zone de connection, (b) l'histogramme log-polaire correspondant, (c) Contexte de forme d'un point d'une seconde zone de connection, (d) l'histogramme log-polaire similaire à (b), (e) Contexte de forme d'un autre point de la seconde zone de connection, (f) l'histogramme polaire assez différent de celui de (b) et (d).	67
4.7	(a) Zone de connection (b), Echantillonnage de ses points de contour. . .	68
4.8	Interpolation de la zone de connection en entrée et le meilleur modèle après déformation par la transformation TPS pour calculer D_{be}	69
4.9	Comparaison des résultats lors de la sélection du meilleur modèle.	70
4.10	(a) Contour de la partie <i>A</i> après transformation <i>TPS</i> , (b) Contour de la partie <i>B</i> après transformation <i>TPS</i>	71
4.11	Résultats de la segmentation en utilisant deux métriques différentes : (a) sans ajustement de points intermédiaires, (b) avec ajustement des points intermédiaires ajustés.	71
4.12	Exemples de segmentation pour un Matchscore supérieur ou égal à 0.9. . .	73
4.13	Paramètres de la TGH.	74
4.14	Représentation du modèle.	74
4.15	Même portion appartenant à différents noms propres.	75
4.16	Repérage du nom propre : <i>عمر</i> par la TGH.	76
4.17	(a) Repérage du mot de liaison <i>بن</i> , (b) Segmentation de la ligne en mots. . .	76

TABLE DES FIGURES

Liste des tableaux

2.1	Caractéristiques considérées.	13
2.2	Contribution de la collaboration entre les réseaux et les cycles perceptifs pour le vocabulaire de taille 5757.	16
2.3	Notre système versus quelques systèmes existants.	18
2.4	Reconnaissance hors ligne par les MGPs.	20
2.5	Résultats obtenus.	26
2.6	Système proposé versus quelques travaux similaires.	26
2.7	Evaluation de performance par classifieur.	33
2.8	Etiquettes attribuées.	34
2.9	Exemple de règle de la grammaire proposée et son codage MathML.	36
2.10	Taux de reconnaissance par type de formule.	38
3.1	Synthèse des systèmes d'identification	43
3.2	Test avec différentes méthodes de sélection de descripteurs.	48
3.3	Taux moyen d'identification par classe.	48
3.4	Matrice de confusion.	49
3.5	Evaluation de performance par classifieur.	49
3.6	Evaluation par classifieur.	50
3.7	Résultats par descripteur et classifieur.	58
3.8	Nombre de bins d'orientation vs. Taux d'indentification.	58
3.9	Matrice de confusion.	58
3.10	Notre système versus quelques travaux similaires	59
4.1	Synthèse des travaux sur la segmentation.	64
4.2	Résultats des tests.	72

LISTE DES TABLEAUX

CHAPITRE 1

Introduction générale

Mon sujet d'habilitation concerne mes contributions à la caractérisation et la reconnaissance de l'écriture arabe dans ses différentes formes. Ce sujet soulève des défis importants selon la nature : imprimée ou manuscrite, la dimension : linéaire ou bidimensionnelle, de l'écriture et la taille : réduite ou large, du vocabulaire à reconnaître.

Pour l'imprimé, il existe plusieurs polices de caractères avec de nombreux styles et des mises en page différentes et complexes. Bien que l'écriture imprimée présente des caractères bien alignés horizontalement et souvent bien séparés verticalement, de taille connue et fixe avec un graphisme conforme à une fonte ce qui simplifie la tâche de la reconnaissance, il n'existe pas encore de système capable de lire l'Arabe imprimé dans un environnement non contraint : vocabulaire large, multi-fontes, documents dégradés et/ou voyellés, etc. Comme signalé par Ben Amor et Ben Amara [1], la reconnaissance de l'imprimé arabe n'est pas triviale particulièrement à cause de la multitude de fontes et l'absence de standardisation des calligraphies d'un point de vue typographique.

Dans le manuscrit, l'écriture est souvent sans contrainte. Des lignes successives peuvent se chevaucher. Certains mots d'une même ligne présentent parfois des accollements qu'il faut défaire. Les caractères sont souvent ligaturés et leur graphisme est inégalement proportionné provenant de la variabilité intra et inter-scripteurs. En effet, les personnes n'écrivent pas tous de la même façon, les lettres sont les mêmes mais chacun a sa propre façon de les écrire. De plus, une même personne peut écrire de manières différentes dans des états émotionnels différents. Plus il y a de scripteurs (droitier/gaucher, enfant/adulte, homme/femme, différentes professions), plus il y a de styles d'écritures différents, plus la reconnaissance devient difficile. Ainsi, le système de reconnaissance doit faire face aux

changements des formes des caractères dans un contexte multi-scripteurs où le style du scripteur, le support et le stylo produisent des variations que le système doit contenir. Cela nécessite donc l'emploi de techniques de délimitation très spécifiques et souvent des connaissances contextuelles pour guider la reconnaissance.

Ces défis s'accroissent pour l'Arabe à cause de son aspect cursif, le changement morphologique de ses lettres en fonction de leurs positions dans le mot, la possibilité de ligatures verticales de deux ou trois lettres, etc. Tous ces problèmes et bien d'autres, se trouvent accentués dans le cas du manuscrit arabe où d'autres facteurs interviennent : espaces inter et intra-mots de longueurs différentes, chevauchements possibles entre les parties d'un même mot, accollement et positions erronées des points et des marques diacritiques, etc.

La taille du vocabulaire est également un facteur influant sur les performances d'un système de reconnaissance ainsi que sur la méthode adoptée. Pour des tâches nécessitant un petit vocabulaire (une trentaine de mots dans le cas de la reconnaissance de montants de chèques, par exemple) on cherche à reconnaître le mot comme une entité à part entière. Pour des tâches de très grand vocabulaire on cherche plutôt à reconnaître les lettres au sein du mot. Cependant, les performances obtenues ne sont pas vraiment acceptables, excepté dans le cas de corpus spécialisés. Il est habituel de considérer qu'un petit vocabulaire est constitué d'une dizaine de mots. Un vocabulaire de taille moyenne comporte une centaine de mots. Un grand vocabulaire rassemble des milliers de mots. Enfin, un très grand vocabulaire (ou très étendu) renvoie à plus d'une dizaine de milliers de mots. Il est évident aussi que dans le cas de petits ou moyens vocabulaires, la complexité est moindre, car la réduction du nombre limite l'encombrement mémoire et favorise l'utilisation de méthodes de reconnaissance directes et donc rapides, par balayage systématique de l'ensemble des mots du lexique.

Ainsi en fonction de certains critères choisis (mono-fonte, multi-fontes, omni-fontes, mono-scripteur, multi-scripteurs, omni-scripteurs, vocabulaire large ou réduit, etc.), il peut y avoir des approches de reconnaissance plus dédiées, donc plus efficaces que d'autres. La reconnaissance du manuscrit cursif à vocabulaire de grande taille présente sans doute le cas de difficulté le plus extrême. A l'opposé, l'écriture isolée imprimée est la plus facile, car l'effort de reconnaissance se réduit à une simple reconnaissance de lettres individuelles.

L'écriture bidimensionnelle, comme celle de formules mathématiques, pose aussi des défis cruciaux tant au niveau de la segmentation des éléments de la formule qu'à celui de la reconnaissance de ses symboles et l'analyse de sa structure. Il est difficile de retrouver les opérateurs quand ils ressemblent aux caractères, distinguer les indices et les exposants, dégager les opérateurs implicites, etc. Notons que contrairement aux mots, les formules mathématiques utilisent un arrangement bidimensionnel de symboles pour encoder les informations. Une erreur d'orthographe dans un mot n'empêchera pas la compréhension

de ce mot. En revanche, une erreur d'un caractère dans une formule mathématique peut modifier complètement son sens. L'écriture en manuscrit de la formule soulève des problèmes additionnels, comme la déformation et le défaut de positionnement des symboles, etc. Être capable de comprendre et d'interpréter les formules mathématiques implique résoudre plusieurs problèmes.

Suivant le type d'écriture (imprimée ou manuscrite, linéaire ou bidimensionnelle) et la taille du vocabulaire (réduite ou grande) qu'un système doit reconnaître, les opérations à effectuer et les résultats peuvent varier notablement. Ainsi, la compréhension de l'écriture par l'ordinateur est encore loin d'être pleinement satisfaisante. La raison est liée au fait que l'analyse et la reconnaissance de l'écriture est un domaine très vaste tant par ses applications que par ses techniques.

Ce sujet est motivé par le constat que les principaux travaux envisagés dans la littérature se sont jusqu'à présent principalement focalisés sur des cas spécifiques : la reconnaissance de mots isolés, de quelques fontes, de vocabulaires spécialisés ou pas assez larges, de formules mathématiques simples ou de documents contraints dans leur format et leur disposition spatiale afin d'obtenir des performances acceptables. L'extension de ces approches à des documents au format et au contenu moins contraints pose des difficultés spécifiques comme la reconnaissance multi-fontes ou multi-scripteurs, le traitement d'un vocabulaire de grande taille, la segmentation de zones de connections entre lignes et/ou mots interconnectés, etc. que les approches utilisées pour le Latin ou pour l'analyse de documents imprimés ne permettent pas de résoudre.

Ces problèmes, comme notifié par Ben Amara [2], engendrent une forte inertie à différents niveaux notamment dans 1) le choix de descripteurs pertinents pour absorber la variabilité de la morphologie des lettres et 2) la méthode de segmentation en lettres ou en pseudo-mots (parties de mots qui peuvent se chevaucher surtout dans le cas du manuscrit). Bien qu'il existe déjà différentes approches pour caractériser et reconnaître l'écriture arabe, les résultats enregistrés n'ont pas pu aboutir à des performances qui peuvent égaler celles atteintes dans le cas d'autres scripts tel que le Latin. Face à tous ces problèmes, la nécessité d'une méthodologie robuste s'impose.

Dans ce travail, je fais une synthèse non exhaustive des approches qui pourraient être envisagées pour dépasser les limitations actuelles. J'ai essayé de mener les techniques aux frontières en exploitant les aspects linguistiques, notamment la richesse et la stabilité morphologique de la langue arabe, d'appréhender l'incertitude de l'écriture humaine, de gérer la complexité et la variabilité des notations mathématiques, etc. Je me suis attachée à traiter ces aspects en proposant des systèmes plus génériques, plus flexibles et plus robustes.

Notre contribution a été faite à quatre niveaux relatifs aux différents aspects de l'écriture : 1) linguistique, 2) bidimensionnel, 3) nature du script et 4) manuscrit :

- Au niveau linguistique, nous avons travaillé sur la reconnaissance d'un grand voca-

bulaire de mots arabes décomposables issus de racines triconsonantiques saines par collaboration de réseaux neuro-linguistiques (thèse d’Imen Becheikh). Nous avons essayé d’accompagner une recherche qui était initiée par Cheriet [3] et par Kanoun [4]. Cette recherche préconise l’intégration de connaissances linguistiques dans le processus de reconnaissance. Nous avons tenté ainsi d’exploiter la richesse et la stabilité morphologique de la langue arabe en factorisant le mot autour de la racine. Cela a permis de mettre en facteur plusieurs composants communs des mots, ce qui a eu pour conséquence une économie dans la représentation du vocabulaire. Nous avons combiné trois classifieurs spécifiques dans la reconnaissance d’entités bien déterminées (racine, schème et éléments de conjugaison). Nous avons utilisé des réseaux de neurones transparents comme classifieurs de mots à partir de leurs descriptions en primitives structurelles globales en couplant des cycles perceptifs aux réseaux. Une thèse soutenue, cinq articles de conférence internationales (ICM-WI’10, DRR’10, ACIT’09, ICPR’08, ICDAR’07) et un article en journal (IJPRAI) sont issus de ce travail.

- Concernant l’aspect bidimensionnel, nous nous sommes inspirés d’un ancien travail sur les formules mathématiques. Partant de la thèse de Belaïd [5] sur la reconnaissance en ligne de formules mathématiques latines, j’ai poursuivi d’abord ce travail sur des documents mathématiques hors ligne et ce dans le cadre de ma thèse de doctorat en proposant un système de segmentation par extraction automatique de formules mathématiques à l’intérieur et à l’extérieur du texte. Ensuite, nous avons proposé, dans le cadre de la thèse de Kaouther Khazri, un système dirigé par la syntaxe pour reconnaître des formules mathématiques arabes en mode hors ligne. En s’appuyant sur des descripteurs statistiques et un classifieur K^* , pour représenter et reconnaître les symboles mathématiques, un analyseur syntaxique ascendant-descendant a été proposé. Il repose sur la dominance d’opérateurs pour diviser récursivement la formule en sous expressions plus simples. Dans le système proposé, les modules de reconnaissance des symboles et d’analyse structurelle s’interagissent d’une manière étroite. Il est ainsi possible d’utiliser des informations structurelles pour aider à prédire les symboles en confusion. La représentation de structure de formule est codée en MathML, ce qui facilite le traitement automatique, la recherche et la réutilisation des formules mathématiques. Dans ce travail, j’ai encadré deux mastères de recherche et une thèse. Cinq articles publiés dans des conférences internationales (ASAR’17, ICDAR’15, ICTIA’14, CIFED’10, GREC’09) sont issus de ce travail.
- Au niveau du script et sa nature, nous avons proposé de séparer l’Arabe du Latin de natures imprimée et manuscrite, au niveau du mot (thèse d’Asma Saïdani). Nous avons cherché une représentation du mot qui soit à la fois suffisamment générique pour englober tous les types d’écritures, et suffisamment discriminante

pour ne représenter qu'un des scripts. C'est une représentation intermédiaire, basée sur le calcul de plusieurs descripteurs, permettant de mieux résumer l'information que les seules valeurs des pixels. Au départ, nous avons extrait plusieurs types de descripteurs (statistiques, structurels, etc.) d'images de mots. Plusieurs types de classifieurs (Bayes, k -plus proches voisins, les séparateurs à vaste marge, etc.) sont aussi testés. Ensuite, nous avons proposé une méthode alternative de discrimination entre mots arabes et latins, en se basant sur l'inclinaison et l'orientation de l'écriture. Cette méthode s'appuie sur le concept d'histogramme des orientations de gradients. Dans ce travail, j'ai encadré un mastère et une thèse. Six articles publiés dans des conférences internationales (ATSIP'16, ICDAR'15, SoCPaR'14, ICTIA'14, ICDAR'13, ICDAR'11) et quatre papiers journaux publiés (PAA, ELCVIA, ELCVIA, IJCIS).

- Au niveau du manuscrit, nous avons abordé la problématique de la segmentation en lignes et en mots que nous avons appliquée sur des anciens manuscrits arabes (thèse de Nabil Aouadi). L'objectif est de repérer et segmenter les zones de connections entre lignes successives ou entre mots de la même ligne. Les manuscrits utilisés sont des registres de dénombrement issus du 19ème siècle. Ils présentent beaucoup de spécificités tant au niveau de l'écriture elle-même où diverses distorsions sont présentes, qu'à celui de la syntaxe utilisée pour l'écriture des noms et des prénoms avec beaucoup de répétitions, de renvois aux noms de parents, etc. Le repérage des zones de connections entre lignes s'est fait par analyse du profil de la projection verticale et entre les mots par une squelettisation suivie d'une recherche des points d'intersection au dessous de la ligne de base. Une fois les zones de connections repérées, elles sont segmentées en se basant sur le descripteur de contexte de forme et une fonction de déformation. Nous nous sommes inspirés des travaux de Belongi et al. [6] et celui de Kang et al. [7] et proposé une nouvelle forme d'appariement entre la zone de connection à segmenter et les modèles stockés dans un dictionnaire ce qui a permis une nette amélioration dans les résultats. La méthode de segmentation, mise en place, a permis aussi de segmenter les lignes en entités informatives : nom et prénoms en dégageant les mots de liaison. Nous nous sommes servis de la syntaxe des noms de personnes et nous avons proposé une méthode de repérage des mots de liaison en nous basant sur la transformée généralisée de Hough. Dans ce travail, j'ai encadré deux mastères de recherche et une thèse. Dix papiers de conférences internationales publiés (ICFHR'16, ICDAR'15, ICFHR'14, SoCPaR'14, PSIVT'13, DRR'13, ICEESA'13, CIMTA'13, GREC'11, PATTERNS'11) ainsi que trois papiers journaux (PAA, IJCIS, IAJeT) sont issus de ce travail.

Un autre travail, concernant l'aspect manuscrit de l'écriture, s'attache à la reconnaissance de mots arabes isolés (thèse d'Akram Khémiri). Nous avons expérimenté différents modèles graphiques probabilistes en vue de prouver leur bonne adapta-

tion à modéliser les variations morphologiques des lettres et leur capacité à résoudre les problèmes épineux de la segmentation. Nous avons implémenté des modèles markovien linéaires horizontaux (MMC-H) et verticaux (MMC-V). Nous avons découvert que le MMC-V est capable de mieux reconnaître les mots que MMC-H car l'information est étagée en trois bandes et qu'en exploitant les corrélations entre variables, une modélisation plus fine peut être obtenue. Dans ce sens, nous nous sommes inspirés du travail d'Alkhateeb [8] et nous nous sommes orientés vers les réseaux bayésiens dynamiques par couplage des flux de données en un seul modèle. Ce couplage est effectué grâce à une architecture qui combine MMC-H et MMC-V dont les entrées sont des descripteurs structurels extraits respectivement des bandes horizontales et verticales du mot. Il en résulte une représentation plus fine d'image du mot et une amélioration du cadre basique du MMC. Dans ce travail, j'ai encadré un mastère et une thèse. Six articles dans des conférences nationales et internationales (CUP'16, ICFHR'16, ICDAR'15, ICFHR'14, ICFHR'12, CIDE'12) et un papier journal publié dans ELCVIA sont issus de ce travail.

A travers ces travaux, nous avons essayé de montrer comment la reconnaissance de l'écriture se ramène à des choix précautionneux de descripteurs et de classifieurs aptes à les intégrer avantageusement. Nous avons également étudié l'impact du choix de ces descripteurs et classifieurs sur les performances des systèmes de reconnaissance de l'écriture, d'identification du script ou de segmentation de documents. Des bases de données spécifiques et des documents de vérité ont été construits à cet effet.

Le présent document est structuré comme suit :

- Après ce chapitre introductif, le chapitre 2 est consacré à la présentation de nos contributions en reconnaissance de l'écriture. Il s'intéresse particulièrement au choix de descripteurs et classifieurs spécifiques dans le cadre de : 1) la reconnaissance d'un vocabulaire large de mots arabes, 2) la reconnaissance de mots arabes manuscrits et 3) la reconnaissance formules mathématiques arabes.
- Le chapitre 3 est consacré à l'étude de choix de descripteurs et classifieurs dans le cadre de l'identification du script et de sa nature où l'objectif est de séparer l'Arabe du Latin.
- Le chapitre 4 est destiné à la segmentation en lignes et en mots de manuscrits arabes.
- A ces chapitres sont jointes une conclusion générale en présentant quelques perspectives pour la poursuite de ces travaux et une liste de références bibliographiques.

CHAPITRE 2

Contributions en reconnaissance de l'écriture arabe

2.1 Introduction

La reconnaissance de l'écriture vise la conversion d'un texte écrit en une représentation compréhensible par une machine et facilement reproductible par un logiciel de traitement de texte. Parce que chaque personne produit une écriture qui lui est propre, et parce qu'il existe de nombreuses polices de caractères pour l'imprimé avec de nombreux styles et des mises en page différentes et complexes, l'écriture possède une infinité de représentations ce qui rend difficile sa reconnaissance automatique.

Dans le processus complet d'un système de reconnaissance de l'écriture, la classification joue un rôle important en se prononçant sur l'appartenance d'une forme telle qu'un graphème, un caractère ou un mot à une classe. L'idée principale de la classification est d'attribuer une forme non connue à une classe prédéfinie à partir de la description de la forme. De ce fait, l'extraction de descripteurs (encore appelés caractéristiques, indices visuels, primitives ou même paramètres), c'est-à-dire la représentation des données d'entrée, est une étape critique et cruciale dans les systèmes de reconnaissance car même si on utilise un classifieur très performant celui-ci ne peut compenser un mauvais choix de descripteurs. La sélection des descripteurs pertinents, discriminants et de dimension limitée, tout en évitant le risque de perte des informations importantes et significatives, est alors d'une grande importance pour les raisons suivantes : 1) la mauvaise représentation des données par des descripteurs conduit à des résultats insatisfaisants dans la reconnais-

2.2 CONTRIBUTION 1 : RECONNAISSANCE D'UN VOCABULAIRE DE MOTS ARABES DE GRANDE TAILLE

sance et 2) beaucoup de redondance de la représentation conduit à des confusions. Par conséquence, de très nombreux travaux sont consacrés à la recherche sur des méthodes d'extraction de descripteurs pertinents, vue l'importance de sélectionner un ensemble de descripteurs discriminants pour la conception des systèmes de reconnaissance puissants et performants. Ainsi, les descripteurs et les classifieurs sont des éléments fondateurs en reconnaissance de l'écriture. Leur choix et associations sont capitales en fonction des caractéristiques de la langue et la qualité du document.

La synthèse des travaux considérés, par Ben Amor et Ben Amara dans [9], montre que les différents types de descripteurs (structurels, géométriques, statistiques, issus de transformations globales, corrélations, etc.) ont été pratiquement toutes utilisées pour la description de l'Arabe, cependant le problème de la caractérisation reste encore d'actualité. Ce chapitre présente l'étude de nos choix de descripteurs et classifieurs dans le cadre de nos trois premières contributions notamment 1) la reconnaissance d'un grand vocabulaire de mots arabes, 2) la reconnaissance de mots arabes manuscrits et 3) la reconnaissance de formules mathématiques arabes.

2.2 Contribution 1 : Reconnaissance d'un vocabulaire de mots arabes de grande taille

Ma première contribution, faisant l'objet d'une thèse que j'ai encadré celle d'Imen Bencheikh, porte sur la reconnaissance d'un vocabulaire large de mots arabes décomposables, issus de racines saines tri-consonantiques. Ce problème pose des défis à cause de la complexité morphologique de la langue arabe, la rareté des données d'apprentissage disponibles, le nombre gigantesque de mots arabes (les mots effectifs arabes se comptent en millions [10]) ce qui rend leur traitement automatique irréaliste. Plusieurs travaux de recherche ont ciblé la reconnaissance d'un vocabulaire large de mots arabes. Certains chercheurs ont reconsidéré des approches proposées pour le Latin en les adaptant aux particularités de l'écriture arabe, sans exploiter l'aspect linguistique. D'autres chercheurs ont opté pour l'intégration de connaissances linguistiques pour guider le processus de la reconnaissance. Mais aucune approche n'a vraiment exploité la richesse et la stabilité de la langue arabe et leur impact sur le processus de la reconnaissance de l'écriture. Nous persistons à croire que les approches prometteuses sont celles intégrant dans le processus de reconnaissance des informations linguistiques permettant de guider le processus de reconnaissance. Pour le prouver, nous allons montrer comment l'incorporation de connaissances linguistiques dans un modèle perceptif permet la reconnaissance d'un vocabulaire large de mots arabes où l'approche proposée allie la morphologie de la langue arabe au processus de lecture humaine. Cette approche est inspirée du système neuronal cognitif de Mc Clelland et Rumelhart (voir la Figure 2.1) permettant une représentation

du mot par couches, allant du local au global et inversement [11].

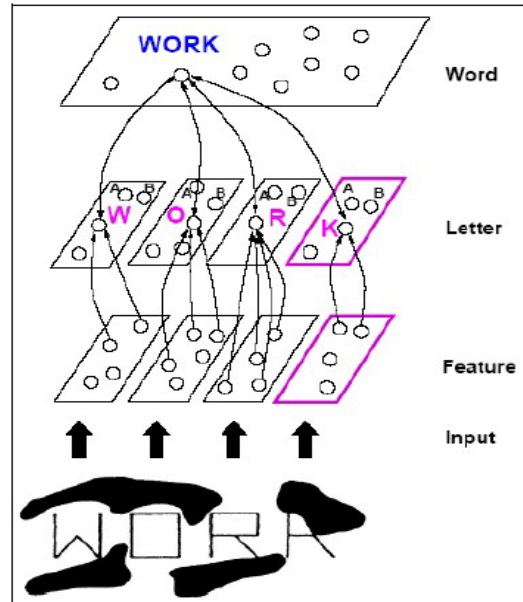


FIGURE 2.1 – Modèle perceptif, inspiré du système de vision humaine [12].

Dans ce qui suit, nous présentons une brève analyse linguistique de l'Arabe. Ensuite, nous décrivons les manières avec lesquelles les connaissances linguistiques arabes ont été considérées pour la reconnaissance de l'écriture arabe dans des travaux de recherche connexes antérieurs. Après, nous discutons certains problèmes qui doivent être surmontés par le biais de l'approche neuro-linguistique proposée. Enfin, nous détaillons l'approche proposée, basée sur une collaboration de classifieurs neuro-linguistiques en détaillant les étapes d'extraction de descripteurs et classification. Des séries d'expérimentations ont été effectuées pour démontrer la robustesse et la stabilité des classifieurs, face à un vocabulaire de plus en plus large.

2.2.1 Analyse linguistique de l'Arabe

Un mot arabe est décomposable ou non. فرنسا (France), دكتور (Docteur) et ثمانية (Huit) sont des exemples de mots non décomposables. Un mot décomposable dérive d'une racine selon un schème donné. La racine est la forme du verbe nue. Elle est soit tri-consonantique (trois lettres), soit quadri-consonantique (quatre lettres). Elle est aussi soit saine, soit mal saine (contient au moins une lettre voyelle : ا, و, ي). La racine est interférée avec des schèmes pour former des mots. Le schème peut être considéré comme étant un patron qui est régi par des règles bien déterminées. Ces schèmes produisent des noms et des verbes [13]. Ainsi, le mot décomposable est composé de morphèmes :

Notre vision du mot est différente : un mot décomposable est le résultat de la dérivation de sa racine selon un schème conjugué. Ce dernier est l'association du préfixe et du suffixe (lettres de la conjugaison issues du temps, du genre, du nombre, etc.) à un schème bref. La dérivation de la racine suivant le schème bref donne le radical comme expliqué dans la Figure 2.2. Cette dérivation donne naissance à un mot représentant éventuellement : un préfixe, un infixe et un suffixe. Les schèmes brefs ne dépassent pas les 75. Les schèmes conjugués sont de l'ordre de 1400. Ainsi, pour reconnaître un mot, nous avons juste besoin d'identifier sa racine et son schème conjugué, sans avoir besoin de le segmenter en lettres.

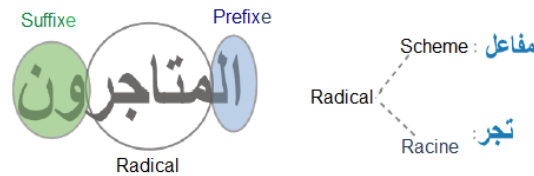


FIGURE 2.2 – Notre vision du mot en Arabe.

Le problème de la reconnaissance d'un vocabulaire large de mots arabes se réduit ainsi à un problème de factorisation autour du mot en associant les aspects linguistiques de manière à élargir le spectre de la reconnaissance à d'autres mots présentant les mêmes caractéristiques linguistiques. Dans ce sens, nous avons proposé une approche linguistique générique fondée sur la collaboration de trois classifieurs linguistiques pour reconnaître respectivement la racine de laquelle le mot se dérive, le schème que le mot suit et ses éléments de la conjugaison. Le mot est enfin reconnu par sa reconstitution à partir de sa racine, son schème et ses éléments de conjugaison.

2.2.3 Système proposé, basé sur la collaboration de classifieurs neuro-linguistiques

Le système proposé consiste en la collaboration de trois classifieurs bien déterminés, spécialisé chacun en l'apprentissage et la reconnaissance de l'un des trois concepts : racines, schèmes et éléments de conjugaison. Ces classifieurs peuvent être des RNAs, tels qu'ici proposés, des MMCs ou autre type de classifieurs. Selon l'approche de reconnaissance adoptée (globale, locale ou hybride) avec ou sans retours (pour une vérification locale), et le type de descripteurs extraits, différents systèmes peuvent être conçus. Dans ce travail, nous avons adopté l'approche hybride (qui opère globalement puis localement). Nous avons utilisé des réseaux de neurones transparents (RNT_R, RNT_S et RNT_C). Nous avons extrait des primitives structurelles globales et opté pour un raffinement local à des cycles perceptifs (voir la Figure 2.3).

2.2 CONTRIBUTION 1 : RECONNAISSANCE D'UN VOCABULAIRE DE MOTS ARABES DE GRANDE TAILLE

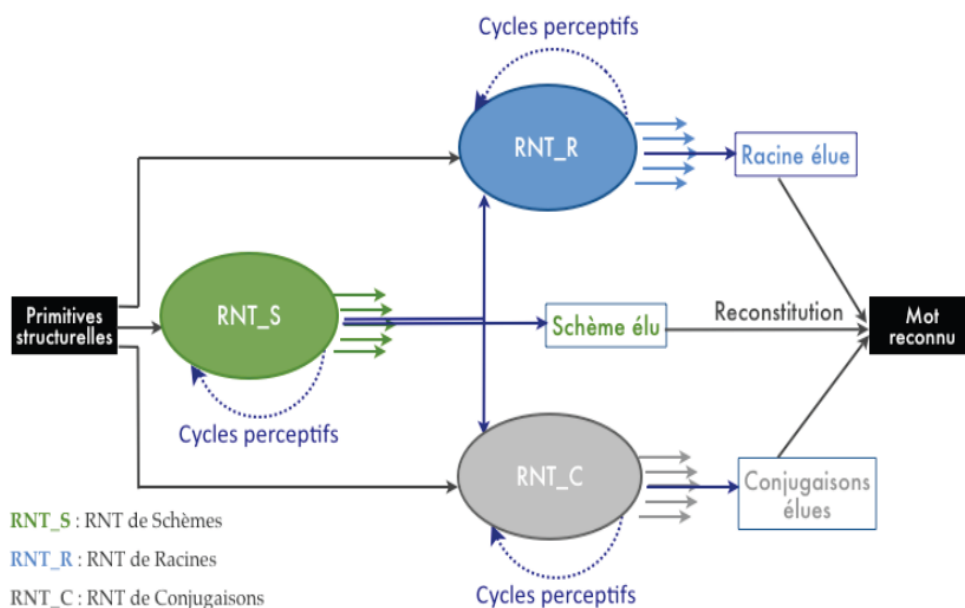


FIGURE 2.3 – Architecture du système proposé.

Notre choix pour les RNAs est justifié par 1) leur aptitude naturelle à fournir des surfaces de décision complexes (par combinaison non linéaire des primitives) et 2) leur grande capacité d'apprentissage. Leur inconvénient étant l'effet de boîtes noires. Pour remédier à cela, nous nous sommes orientés vers les réseaux de neurones transparents (RNTs) où un concept bien déterminé est associé à chaque neurone ce qui favorise l'analyse et la correction rapides des résultats. De plus, ce genre de modèle transparent est inspiré du système à vision humaine (voir la Figure 2.1) développé d'abord par W.S. McCulloch [12], utilisé ensuite pour le Latin par M. Côté [17] et reconsidéré pour l'Arabe par S. Maddouri [18], par l'utilisation d'un RNT ayant autant de couches que de niveaux de décomposition du mot arabe. Rappelons que ce modèle opère d'abord globalement par extraction de primitives structurelles globales, puis en raffinant la vision par un retour local suivant des cycles perceptifs. Notons que ce modèle ne nécessite pas une étape de segmentation qui pose problème en raison de l'aspect cursif de l'écriture arabe. Il suffit d'extraire des primitives structurelles globales pour représenter le mot.

Extraction de primitives structurelles

Comme le montre la Table 2.1, une caractéristique est une hampe, un jambage, une boucle, des diacritiques ou rien de ce qui précède et une position dans le PAW : au début, au milieu, à la fin ou isolée. Nous avons considéré une primitive toute combinaison de caractéristiques. L'association primitive-position est faite parce qu'une même lettre peut présenter plusieurs caractéristiques de forme en même temps (boucle, diacritiques,

TABLE 2.1 – Caractéristiques considérées.

Désignation	Caractéristique	Description
H	Hampe	Caractéristique au dessus de la bande de base
J	Jambage	Caractéristique au dessous de la bande de base
B	Boucle	Intersection de deux contours fermés
P	Points dicritiques hauts	Diacritiques au dessus du corps du mot
Q	Diacritiques bas	Diacritiques au dessous du corps du mot
R	Rien	Aucune des cinq caractéristiques listées ci-haut
D	Début	Position de la caractéristique au début du PAW
M	Milieu	Position de la caractéristique au milieu du PAW
F	Fin	Position de la caractéristique à la fin du PAW
I	Isolé	Caractéristique isolée

etc.) et une seule position dans un PAW. Par exemple, la primitive BD désigne une boucle au début du PAW comme pour les lettres م ou ص et la primitive JPI désigne un jambage isolé avec des diacritique hauts comme pour les lettres ش, ن et ق. L'étape d'extraction de primitives vise ainsi à générer, pour un mot donné, sa description en primitives structurales, de droite à gauche, comme le montre la Figure 2.4.



FIGURE 2.4 – Description en primitives structurales du mot متاجرون.

Pour cela, il faut extraire des caractéristiques globales telles que des hampes, des jambages, des boucles, etc. avec leurs positions respectives, comme le montre la Figure 2.5.

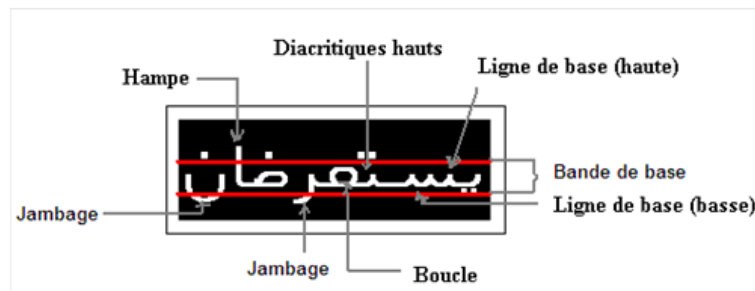


FIGURE 2.5 – Caractéristiques globales d'une image de mot

Apprentissage des classifieurs

Nous avons fixé pour chaque mot (échantillon de la base d'apprentissage), les entrées et la ou les sorties désirées pour un apprentissage supervisé de classifieurs. La fonction d'activation des neurones étant la fonction "Sigmoidé". Les pondérations des connexions

2.2 CONTRIBUTION 1 : RECONNAISSANCE D'UN VOCABULAIRE DE MOTS ARABES DE GRANDE TAILLE

entre les neurones étant initialisées aléatoirement. La règle d'apprentissage utilisée est la règle de rétro-propagation du gradient.

- RNT_R est un réseau à trois couches (primitives, lettres et racines) contenant respectivement 50, 104 et 101 neurones qui apprend comment 1) ignorer les préfixes, infixes et suffixes et 2) ne tenir compte que des lettres de la racine du mot. Cette modélisation reste fidèle à la morphologie de l'Arabe où trois lettres permettent de générer plusieurs racines. Par exemple, بعد, دعب, بدع et عبد sont quatre racines tri-consonantiques formées des mêmes lettres ب, ع et د. Notons que contrairement aux systèmes de reconnaissance classiques où l'apprentissage d'un mot se fait via des échantillons différents de ce même mot, RNT_R permet plutôt l'apprentissage d'une racine via des mots différents (mais issus d'une même racine) comme le montre la Figure 2.6.

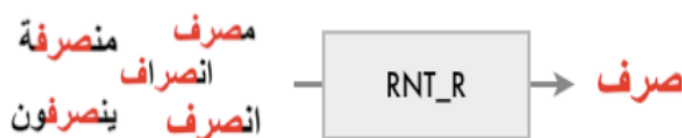


FIGURE 2.6 – Apprentissage par RNT_R d'une même racine à partir de mots différents.

- RNT_S a trois couches (primitives, lettres et schèmes) contenant respectivement 50, 28 et 22 neurones. A l'opposé de RNT_R, RNT_S apprend à ignorer les lettres de la racine et à se concentrer sur celles du schème que le mot suit. Ainsi, face à un échantillon, suivant un schème donné, RNT_S se spécialise dans l'apprentissage des schèmes, en faisant contribuer quelques lettres déterminantes des schèmes parmi les 28 lettres vers le schème en question. RNT_S favorise l'apprentissage d'un schème via des mots différents (mais dérivés de la même schème) et non pas des échantillons différents de ce schème (voir la Figure 2.7).



FIGURE 2.7 – Apprentissage, par RNT_S, d'un même schème à partir de mots différents.

- RNT_C est un réseau à trois couches (primitives, lettres et conjugaisons) contenant respectivement 50, 30 et 12 neurones. Ce réseau apprend comment ignorer les infixes et ne tenir compte que des lettres du mot qui correspondent aux éléments de la conjugaison (flexion). RNT_C favorise l'apprentissage des éléments de

conjugaisons via des mots différents mais ayant subi une même conjugaison (voir la Figure 2.8).

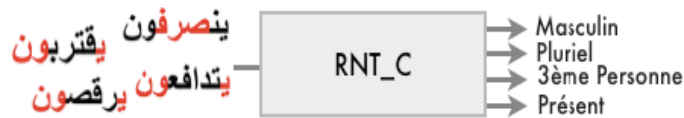


FIGURE 2.8 – Apprentissage, par RNT_C, des mêmes éléments de conjugaison à partir de mots différents.

Collaboration des classifieurs

Nous expliquons ici les phases d'apprentissage et de reconnaissance dans un contexte de collaboration des trois RNTs.

- **Apprentissage dans un contexte de collaboration** : Nous avons observé les comportements des réseaux après avoir effectué leurs apprentissages séparément. Nous avons constaté que les scores de RNT_S sont les meilleurs et qu'il vaudrait mieux compter sur ce réseau dans la supervision des autres réseaux : RNT_R et RNT_C. Pour cela, nous avons suivi les étapes suivantes : 1) alimenter chaque réseau par les primitives structurelles du mot à apprendre, 2) fournir à RNT_R la racine du mot, à RNT_S son schème et à RNT_C ses éléments de conjugaisons, afin que chaque réseau ajuste les pondérations de ses connections selon la sortie désirée correspondante, et 3) compter sur le schème du mot pour superviser RNT_R et RNT_C.
- **Reconnaissance dans un contexte de collaboration** : Les étapes à ce niveau sont : 1) alimenter les trois réseaux par les primitives structurelles du mot à reconnaître, pour que les trois réseaux proposent leurs candidats respectifs pour la racine, le schème et les éléments de conjugaison, 2) autoriser RNT_S à superviser RNT_R et RNT_C, 3) appliquer des cycles perceptifs, séparément à RNT_R, RNT_S et RNT_C pour raffiner la vision et écarter ainsi les mauvais candidats, 4) utiliser des contraintes linguistiques pour rejeter d'autres confusions, en inférant les sorties de RNT_R et celles de RNT_S, puisqu'en Arabe, une racine n'est pas cohérente avec tous les schèmes et vice-versa et 5) reconstituer le mot à partir de la racine, du schème et des éléments de conjugaison retenus.

Notons que les cycles perceptifs servent à éliminer les mauvais candidats de racines, de schèmes et d'éléments de conjugaison via des retours arrières à vision locale qui complète ce qui a échappé à la vision globale. Il s'agit de reprendre le processus de reconnaissance, chaque fois que le réseau rencontre une ambiguïté, et ce en activant certaines entrées, en inhibant d'autres ou bien en agissant directement sur des

2.2 CONTRIBUTION 1 : RECONNAISSANCE D'UN VOCABULAIRE DE MOTS ARABES DE GRANDE TAILLE

neurones des couches internes. Tout ceci est évidemment favorisé par la transparence des réseaux utilisés. L'application de contraintes linguistiques, consiste en la vérification de règles de cohérence entre les racines et les schèmes candidats. Cette phase devrait exclure les fausses propositions.

2.2.4 Expérimentations

Pour générer un vocabulaire, nous avons utilisé entre 51 et 101 racines et nous avons dérivé plusieurs mots selon 22 schèmes en faisant varier les éléments de la conjugaison (masculin, féminin, singulier, duel, pluriel, présent, passé, etc.). Pour construire la base de mots, nous avons utilisé trois fontes arabes : Arial, AGA Abasan Regular et AF_Buryidah dans des polices différentes où il est possible d'observer plusieurs variations dans la hauteur des lettres, de la distance entre les lettres, de la forme et la hauteur des hampes, de la profondeur et la forme des jambages, ligature verticale de quelques lettres, etc.

Pour étudier la stabilité des réseaux face à la taille du vocabulaire, nous avons utilisé 8 vocabulaires de tailles différentes qui varient de 1000 à 1700 mots en injectant 100 nouveaux mots à chaque fois. Nous avons constaté que RNT_R se comporte d'une manière stable avec l'accroissement de la taille du vocabulaire et que plus le vocabulaire est large, plus RNT_S apprend mieux puisque le nombre des schèmes est fixe.

Pour évaluer la performance des réseaux, nous avons utilisé un vocabulaire large de taille 5757, issu de 101 racines (soit 17271 échantillons dont les deux tiers ont servi pour l'apprentissage et le reste pour les tests). L'évaluation est d'abord faite pour chacun des réseaux séparément, ensuite en collaboration et finalement en considérant les cycles perceptifs. Nous avons constaté à chaque fois une nette amélioration des taux de reconnaissance (voir la Table 2.2). Cela prouve la contribution de la collaboration entre les réseaux et l'intérêt des cycles perceptifs quant la reconnaissance d'un vocabulaire large de mots arabes. Notons que les résultats affichés sont fortement liés à la phase d'extraction de primitives (évaluée à 97% sur un vocabulaire de taille 5757 mots).

TABLE 2.2 – Contribution de la collaboration entre les réseaux et les cycles perceptifs pour le vocabulaire de taille 5757.

	RNT_R(%)	RNT_S(%)	RNT_C(%)
Sans collaboration	90.73	95.65	
Avec collaboration/ Sans cycles	94.91	95.65	97
Avec collaboration/ Avec cycles	95.62	98.75	97.95

Pour calculer le taux de reconnaissance des mots ou de leur restitution à partir des racines, schèmes et éléments de conjugaisons, nous avons combiné les trois sorties des réseaux. Pour restituer le radical, à partir des schèmes et racines proposés, nous avons

calculé pour chaque mot, le maximum des produits de la probabilité majoritaire (à priori) du schème avec la probabilité à posteriori de la racine sachant ce schème. Le top_1 , total de restitution suite à cette combinaison est de 95.33% pour un vocabulaire de taille 5757 mots. Bien que, ce taux soit satisfaisant, il importe de noter que la présence de racines morphologiquement semblables (décrites par les mêmes primitives structurales) nuit à l'apprentissage et par conséquent abaisse le taux total de restitution.

La Figure 2.9 illustre un exemple de reconnaissance d'un mot par combinaison des sorties des trois réseaux. Notons que la combinaison a permis de rejeter le schème **فاعل**, le premier candidat de RNT_S (bien que son activation est égale à 0.8), et retenir le schème **فعال**, le deuxième candidat (activé à 0.73). En effet, étant supervisé par le schème **فاعل**, le réseau RNT_R propose la racine **عدل** avec une précision 0.61, alors que supervisé par le schème **فعال**, il retourne la racine **بعد** avec une certitude de 0.92. En multipliant les probabilités respectives et en les comparant : $(0.8 * 0.61 = 0.48 < 0.67 = 0.73 * 0.92)$, le mot **بعادين** (combiné à 0.67) prend le dessus sur **يعادلن** (combiné à 0.48).

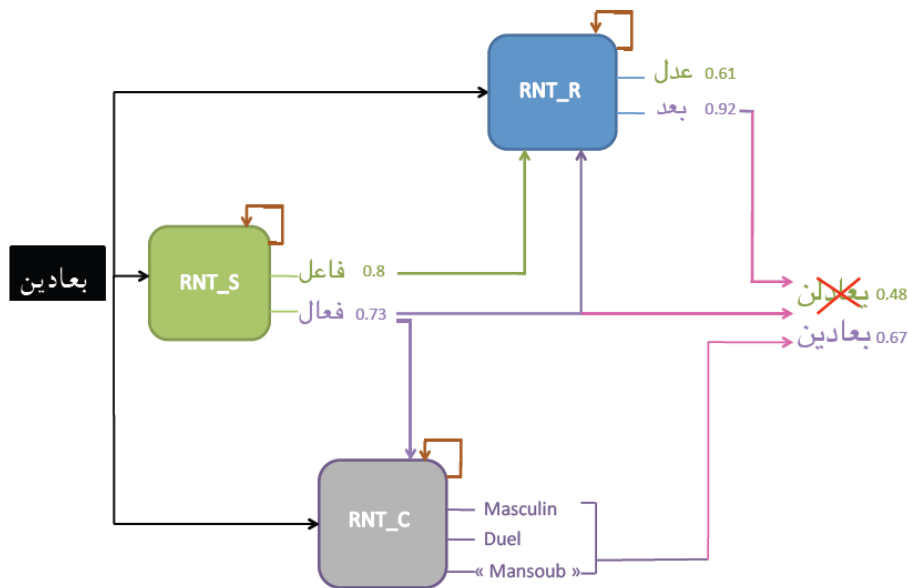


FIGURE 2.9 – Exemple de combinaison pour la reconnaissance du mot **بعادين**.

Il importe de noter que notre système est capable de reconnaître des échantillons de mots qui n'ont pas été appris auparavant : des mots qui n'apparaissent pas dans le vocabulaire considéré. Par exemple, les deux mots **مخالفان** et **أدرکت** ne figurent pas dans le vocabulaire traité mais leurs entités morphologiques ont bien été reconnues puisque d'autres mots ont permis l'apprentissage des racines **خلف** et **درک** et des schèmes **مفاعل** et **أفعل**. Ceci confirme l'aspect générique de l'approche et son aptitude à reconnaître un plus grand nombre de mots sans nouvel apprentissage.

2.2 CONTRIBUTION 1 : RECONNAISSANCE D'UN VOCABULAIRE DE MOTS ARABES DE GRANDE TAILLE

Comme notre système est basé sur une approche neuro-linguistique, à vision globale qui intègre des connaissances linguistiques alors qu'elle traite un vocabulaire large, nous l'avons comparé à d'autres systèmes qui 1) utilisent des propriétés linguistiques, comme ceux proposés dans [14] et [16] et 2) sont conçus pour traiter des vocabulaires larges, comme ceux proposés dans [19] et [20], bien que les expérimentations de certains de ces systèmes sont menées sur des vocabulaires qui ne sont pas suffisamment larges (voir la Table 2.3).

Touj et al. dans [19] ont proposé une approche à base de MMCs planaires pour la reconnaissance de l'écriture arabe manuscrite et ce en décomposant l'écriture en cinq régions horizontales homogènes (région médiane, des hampes, des jambages, des diacritiques hauts, des diacritiques bas). Chaque région est décrite par un 1D-HMM. Kanoun et al. dans [14] ont proposé un système pour la reconnaissance de textes imprimés. Leur approche est dite affixale et consiste en la segmentation des mots en lettres et la reconnaissance des entités morphologiques de ces mots. Les morphèmes sont validés en utilisant des dictionnaires de racines, infixes, préfixes et suffixes. Des restrictions affixales et sémantiques ont été utilisées pour guider la reconnaissance.

TABLE 2.3 – Notre système versus quelques systèmes existants.

Système	Nature de l'écriture	Taille du vocabulaire	Taux de reconnaissance(%)
[14]	Imprimée	1000	74
[16]	Imprimée	1423	81.3
[19]	Manuscrite	25	88.7
[20]	Imprimée	113000	91.7
Notre système	Imprimée	5757	RNT_R :95.62, RNT_S :98.75, RNT_C :97.95

Nous avons également comparé notre système à ceux proposés dans la compétition d'ICDAR 2011, sur une base commune : la base APTI [21]. C'est une base synthétique, constituée de 113284 échantillons de mots arabes imprimés décomposables. Les résultats obtenus ont montré que notre système atteint un taux de reconnaissance satisfaisant, en comparaison aux systèmes concurrents, puisque les mots ont été parfaitement restitués à partir de leurs morphèmes linguistique à 78.2% des cas.

En conclusion, nous avons proposé une approche hybride en utilisant une méthode neuronale globale couplée à une vision locale. Elle reconnaît les racines, les schèmes et les éléments de conjugaisons de mots à partir de leurs descriptions en primitives structurelles. Des expérimentations détaillées ont été mises en oeuvre, des résultats encourageants sur un vocabulaire de 5757 mots ont été obtenus et des comparaisons à des systèmes concurrents ont été faites, démontrant ainsi l'efficacité de l'approche proposée face aux vocabulaires larges et surtout son intérêt pour l'Arabe. Il importe de noter aussi que cette approche se distingue par une contribution très originale quant à la résolution du problème d'explosion du corpus d'apprentissage. Grâce au nouveau type d'apprentissage

proposé et suite à des expérimentations faites sur des échantillons de mots n'appartenant pas au vocabulaire, nous avons démontré la capacité de notre approche à traiter des vocabulaires de plus en plus larges.

2.3 Contribution 2 : Reconnaissance de mots arabes manuscrits

L'objectif de ma deuxième contribution, concernant la thèse d'Akram Khémiri, est de montrer l'intérêt des modèles graphiques probabilistes (MGPs), notamment les modèles de Markov cachés (MMCs) et les réseaux bayésiens (RBs) quant au traitement des distorsions non linéaires de l'écriture arabe manuscrite.

Notons que la reconnaissance de l'écriture arabe manuscrite est une tâche difficile en raison de sa grande variabilité morphologique, son aspect cursif et l'incertitude de l'écriture humaine. Comme les MPGs se sont avérés très puissants dans plusieurs problèmes de reconnaissance qui nécessitent un raisonnement dans l'incertitude, nous nous sommes orientés vers ces modèles pour la reconnaissance de l'écriture arabe manuscrite. Les MPGs sont principalement caractérisés par la combinaison judicieuse des avantages de la théorie des probabilités avec ceux de la théorie des graphes. Alors que le premier fournit une base solide pour gérer l'incertitude, le dernier offre une modélisation structurelle pour asseoir correctement son raisonnement. C'est dans le contexte de la reconnaissance hors ligne de l'écriture arabe manuscrite que nous nous sommes placés pour étudier sa variabilité et son incertitude et la résoudre par des moyens appropriés, les MPGs. Pour cela, nous avons procédé par l'extraction de quelques primitives structurelles globales (Points diacritiques, Hampes, Jambages, Boucles, etc.) compte tenu de leur nombre, types et positions dans le mot. Ensuite, nous avons réalisé un premier modèle stochastique de type MMC à partir des vecteurs de caractéristiques qui décrivent les mots. Nous avons couplé aussi divers MMCs et implémenté des réseaux bayésiens statiques et dynamiques.

Après un bref rappel de l'application des MGPs, particulièrement les RBs, dans le domaine de la reconnaissance de l'écriture, nous présentons notre approche et expliquons les étapes d'extraction des primitives structurelles et de classification de mots. Nous présentons et comparons entre diverses architectures de MMCs et RBDs proposés pour atteindre une meilleure performance de reconnaissance. Des expériences menées sur la base IFN/ENIT, soutiennent fortement la faisabilité de l'approche proposée.

2.3 CONTRIBUTION 2 : RECONNAISSANCE DE MOTS ARABES MANUSCRITS

2.3.1 Etat de l'art

Ci-dessous un tableau récapitulatif qui résume quelques travaux existants, auxquels nous nous sommes comparés, pour la reconnaissance hors ligne basée sur des MGPs.

TABLE 2.4 – Reconnaissance hors ligne par les MGPs.

Réf.	Forme	Méthode	Base de données	Taux de reconnaissance(%)
[22]	Chiffres manuscrits artificiellement dégradés	Différents couplages de MMCs en RBDs : AR, STCPL, GNLCPL, ARCPL	MNIST	AR-Vertical=93.2, AR-Horizontal=87.7, STCPL=92.4, GNLCPL=93.4, ARCPL=94.9
[22]	Caractères latins réellement dégradés	Idem	Documents anciens	AR-Vertical=94.5, AR-Horizontal=91.2, STCPL=95.5, GNLCPL=94, ARCPL=96
[23]	Chiffres arabes manuscrits	Idem	ADBase	95.26
[8]	Mots arabes manuscrits	Idem	IFN/ENIT	66.56
[24]	Chiffres latins manuscrits	4 modèles de RBD	MNIST	Modèle ₁ =67.6, Modèle ₂ =69.9, Modèle ₃ =74.8, Modèle ₄ =71.2
[25]	Mots arabes manuscrits	RB Naïf, RB augmenté par un arbre, RB augmenté par une forêt, RBD	IFN/ENIT	RB Naïf=73, RB augmenté par un arbre=80, RB augmenté par une forêt=82.56, RBD=83.7

2.3.2 Approche proposée, basée sur les modèles graphiques probabilistes

Nous avons préconisé au début, une approche s'appuyant sur les MMCs, connus pour leur capacité d'intégration du contexte et d'absorption du bruit. Nous avons proposé deux modèles (MMC-V, MMC-H). Nous avons constaté par la suite que même si les MMCs ont eu de bonnes performances, ils sont cependant des modèles essentiellement monodimensionnels. Or l'écriture hors ligne a une nature réellement 2D. Ainsi, nous avons essayé de modéliser l'écriture par une approche entièrement bidimensionnelle. En exploitant les corrélations entre variables, une modélisation plus fine peut être obtenue. Dans cette optique, nous nous sommes orientés vers les RBDs qui ont l'avantage de surmonter les limitations des MMCs. L'idée principale consiste à construire des modèles 2D en couplant dans un RBD deux réseaux de type MMC. Le premier MMC est obtenu à partir des descripteurs extraits des colonnes (MMC-V), le second à partir des descripteurs extraits des lignes (MMC-H). L'un des avantages de ces modèles est d'obtenir une bonne modélisation des images des mots manuscrits, due aux regroupements des informations concernant les lignes et les colonnes. Comme ces modèles nécessitent des séquences d'observations en entrée, nous avons extrait des primitives structurales à partir d'images des mots, comme expliqué ci-après.

2.3 CONTRIBUTION 2 : RECONNAISSANCE DE MOTS ARABES MANUSCRITS

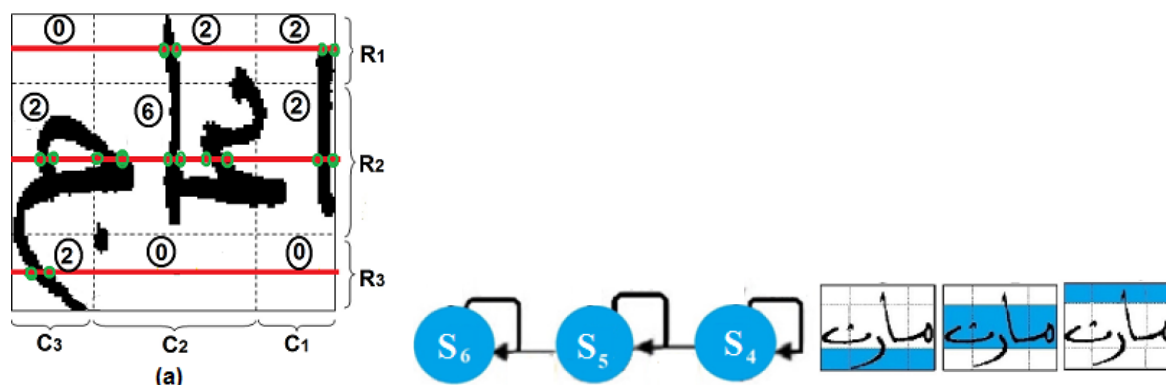


FIGURE 2.11 – (a) Nombre de transitions, (b) Structure du MMC-H (Etape d’initialisation).

- **MMC Vertical (MMC-V)** : Dans MMC-V, la description du mot est effectuée de droite à gauche comme une séquence de primitives structurales : hampes, jambages, boucles, diacritiques (voir Figure 2.12(a)) et nombre de PAWs (Voir Figure 2.12(b)) recueillies à partir de chaque colonne d’image du mot. La Figure 2.12(c) présente la structure du MMC-V.

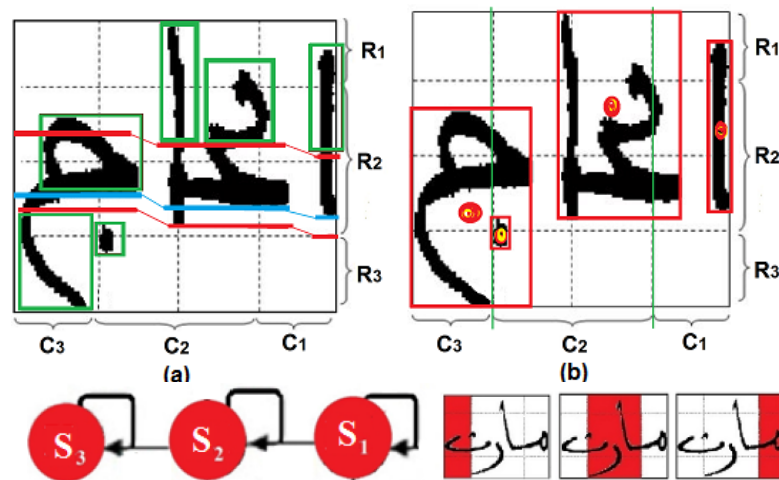


FIGURE 2.12 – (a) Hampes, jambages, boucles et diacritiques, (b) PAWs, (c) Structure du MMC-V (Etape d’initialisation).

MMC-H et MMC-V sont réalisés à partir des vecteurs de caractéristiques qui décrivent les mots. Ces MMCs calculent, à partir du vecteur de caractéristiques du mot à reconnaître, le logarithme de vraisemblance par rapport à tous les modèles de mots de référence et le mot reconnu est celui qui a la valeur maximale. Notons que MMC-H et MMC-V sont des MMCs discrets, à une dimension et de type

droite à gauche sans sauts d'états. Comme MMC-V et MMC-H sont des modèles essentiellement mono-dimensionnels puisqu'ils supposent l'indépendance entre les rangées et les colonnes des mots, nous pensons qu'en proposant une modélisation bidimensionnelle et l'exploitation des corrélations entre variables, une modélisation plus fine peut être obtenue. Dans ce sens, nous nous sommes orientés vers les RBs comme expliqué ci-dessous.

- **RBs statiques** Pour résoudre le problème multidimensionnel, nous nous sommes orientés vers les réseaux bayésiens statiques puis dynamiques. Nous avons essayé d'abord avec le Bayes naïf (BN) qui malgré sa simple structure (voir la Figure 2.13) a donné de bons résultats.

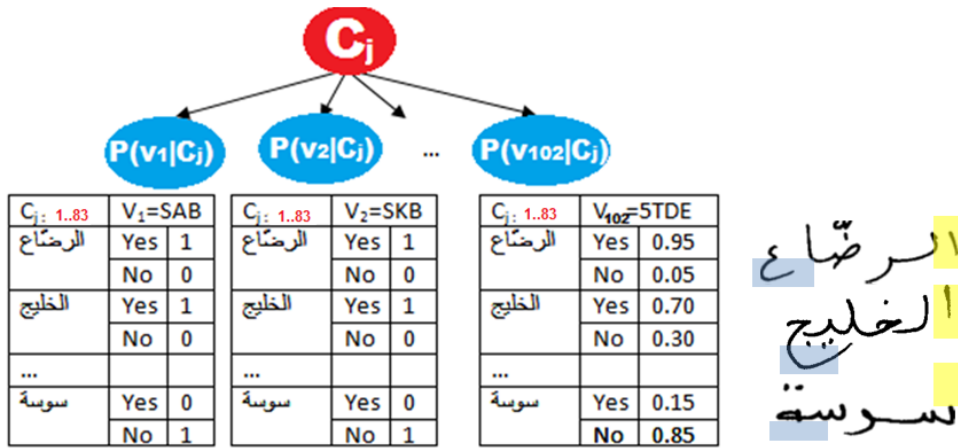


FIGURE 2.13 – Structure proposée pour Bayes naïf.

Or l'hypothèse d'indépendance entre les attributs est généralement fautive. A cet effet, il existe différentes techniques pour assouplir cette hypothèse. Elles consistent à identifier les dépendances conditionnelles entre les attributs. Nous augmentons ainsi le réseau bayésien naïf par un arbre (TAN, voir la Figure 2.14). Toutefois, si les RBs statiques gèrent bien la multi-dimensionnalité, il ne gère pas l'aspect temporel, d'où notre recours aux réseaux bayésiens dynamiques (RBDs).

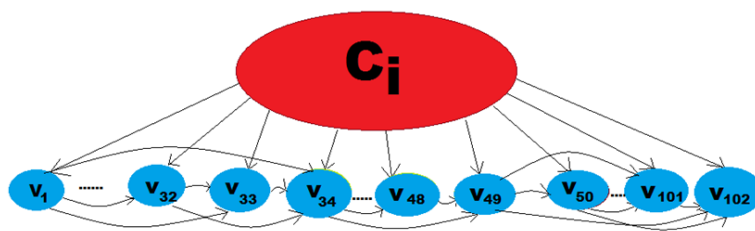


FIGURE 2.14 – Structure proposée pour Bayes naïf augmenté par un arbre.

2.3 CONTRIBUTION 2 : RECONNAISSANCE DE MOTS ARABES MANUSCRITS

- **RBDs, conçus par couplages de MMC-V et MMC-H** : Nous avons conçu trois RBDs par différents couplages de MMC-V et MMC-H en ajoutant des liens orientés entre les variables (noeuds). Nous avons modélisé les dépendances entre les noeuds en créant des interactions entre les deux MMCs. Le MMC-V (resp. MMC-H) prend pour séquence d'observations les primitives structurelles extraites des colonnes du mot (resp. les rangées du mot). La modélisation 2D résulte du couplage entre variables d'états de ces deux MMCs. Il s'agit bien de modélisations bidimensionnelles puisqu'elles regroupent les données selon les lignes et les colonnes simultanément (voir la Figure 2.15).

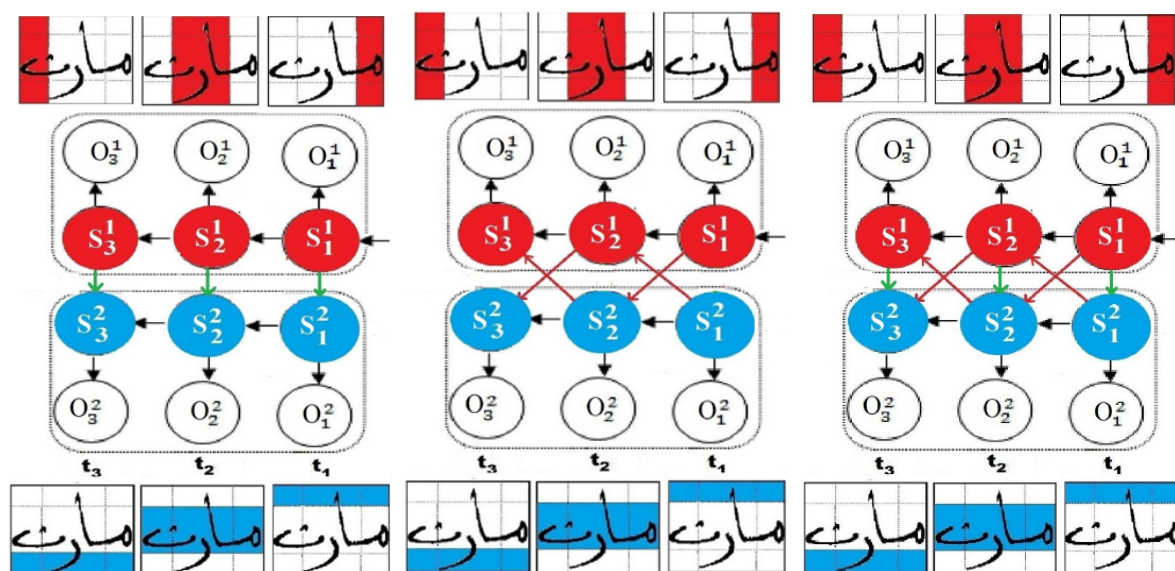


FIGURE 2.15 – De gauche à droite : RBD0, RBD1 et RBD2.

Les MMC-H et MMC-V couplés sont divisés en 3 tranches de temps (3 flux d'observations). Dans chaque tranche de temps, le RBD contient un certain nombre de variables aléatoires représentant les observations et états cachés du processus. Les dépendances entre MMC-V et MMC-H, modélisant à la fois les données verticalement et horizontalement, sont effectuées par les relations entre les états. Notons que le sens des arcs reflète la circulation de l'information et apporte une information qualitative. Nous avons renforcé l'influence du flux vertical en orientant les arcs du flux vertical vers le flux horizontal (écriture de haut en bas et de droite à gauche) car il s'est avéré que MMC-V, appliqué d'une manière indépendante, est capable de mieux reconnaître les mots que MMC-H. Ce qui est traduit par les directions des flèches des variables d'états colonnes vers les variables d'états lignes.

Soient pour $i = 1..2$, $(S_t^i)_{1 \leq t \leq T}$, avec $T = 3$, les variables d'états cachées et $(O_t^i)_{1 \leq t \leq T}$, les séquences d'observations correspondant aux colonnes du mot (resp.

les lignes) à chaque instant t . Pour chaque $i = 1..2$, les S_t^i agissent sur les O_t^i pour chaque mot. Dans RBD0, les variables d'états cachées S_t^1 du MMC-V sont liées aux S_t^2 du MMC-H. Dans RBD1, un état d'un MMC-V est connecté aux états voisins dans la la tranche suivante du MMC-H. Dans RBD2, un état d'un MMC-V est connecté aux états voisins dans la même tranche et la tranche suivante du MMC-H. Le processus, modélisé par ces RBDs, est Markovien de premier ordre et stationnaire puisque les parents de toute variable S_t^i appartiennent à la tranche du temps t ou $t - 1$. Ces modèles de couplage, bien qu'ils sont simples, ils sont plus riches d'informations que les MMCs indépendants.

Pour avoir une spécification complète de notre RBD, nous avons besoin de définir les paramètres suivants : 1) la probabilité de transition entre les états $(S_t|S_{t-1})$, 2) la probabilité conditionnelle des états cachés étant donnée une observation $P(O_t|S_t)$ et 3) la propabilité de l'état initial $P(S_1)$. Pour apprendre les paramètres du RBD, un modèle est élaboré pour chaque classe. Les modèles de toutes les classes partagent une structure unique du RBD, mais leur paramètres changent d'une classe à l'autre. L'apprentissage des paramètres du RBD est effectué indépendamment modèle par modèle, en utilisant l'algorithme de Baum-welch. Pour reconnaître un mot, ses primitives structurelles sont extraites. Ensuite, la probabilité de chaque modèle par rapport à l'échantillon est calculée en utilisant un algorithme d'inférence Forward et le mot est associé à la classe qui donne le maximum de vraisemblance.

2.3.3 Expérimentations

Pour évaluer la performance des modèles proposés, nous avons utilisé un sous ensemble de la base IFN-ENIT (base de données standard de noms de villes tunisiennes, écrits à la main). Nous avons utilisé 7881 échantillons relatifs aux 83 mots différents. Les résultats obtenus montrent que les modèles couplés (RBD0, RBD1 et RBD2) aboutissent aux meilleurs taux de reconnaissance que les modèles indépendants (MMC-H, MMC-V, BN, TAN). Notons aussi que RBD0 a réussi à mieux reconnaître les mots que les RBD1 et RBD2 (voir la Table 2.5). La performance supérieure du RBD0 peut être attribuée au fait que ce modèle représente mieux les aspects perceptuellemt pertinents du mot arabe manuscrit et qu'il considère les différentes variations morphologiques spécifiques à l'écriture arabe. En conséquence, le RBD a le potentiel de bien reconnaître les mots manuscrits en tant qu'images à 2D. Mais il lui faudrait des matrices de covariance considerables pour tenir compte des dépendances entre les flux horizontaux et verticaux. Comment extraire des primitives utiles et faire usage du tout le potentiel des RBDs nécessite plus d'investigation.

La table 2.6 fournit quelques résultats de systèmes similaires. Il est clair que notre système présente des résultats concurrentiels. Mais, comme les systèmes impliqués uti-

2.4 CONTRIBUTION 3 : RECONNAISSANCE DE FORMULES MATHÉMATIQUES ARABES

TABLE 2.5 – Résultats obtenus.

BN(%)	TAN(%)	RBD0(%)	RBD1(%)	RBD2(%)
78.58	81.64	90.02	84.10%	86.07

lisent différents protocoles et sous ensembles de la base IFN/ENIT, il n'est pas possible de faire une comparaison équitable. Notons que nous nous sommes intéressés que par des travaux spécifiques, fondés sur des RBDs ou des MMCs planaires.

TABLE 2.6 – Système proposé versus quelques travaux similaires.

Système	Corpus	Taux de reconnaissance(%)
[8]	IFN/ENIT	MMC=82, RBD=66
[26]	25 mots, 2347 échantillons	MMC Planaire=88.7
[25]	18 mots, 3600 échantillons	FAN=82.56, TAN=80, BN=73, RBD=83.7
Notre système	21 mots, 83 mots, 7881 échantillons	BN=78.58, TAN=81.64, RBD0=90.02, RBD1=84.10, RBD2=86.07

En conclusion, nous avons construit une variété de modèles markovien traditionnels indépendants (MMC-V, MMC-H). Nous nous sommes orientés par la suite au formalisme réseaux bayésiens (BN, TAN, RBD), couramment employé pour le raisonnement sous incertain. Ce formalisme constitue en fait un cadre intéressant pour deux raisons. D'une part, grâce à sa structure graphique, les RBs offre un outil naturel pour représenter les dépendances entre les différentes variables d'un système donné. D'autre part, en exploitant les indépendances conditionnelles entre les variables, il introduit une certaine modularité dans les systèmes complexes. Nous avons proposé de coupler des flux de données en un seul RBD. Ce couplage est effectué grâce à une architecture qui combine les deux MMCs de base : MMC-H et MMC-V dont les sorties sont les primitives structurelles extraites respectivement des colonnes et des lignes. Il en résulte une représentation plus fine d'image du mot et une amélioration du cadre basique du MMC.

2.4 Contribution 3 : Reconnaissance de formules mathématiques arabes

Nous nous intéressons dans cette étude au problème posé par la reconnaissance de formules mathématiques arabes hors lignes et ce dans le cadre de la thèse de Kaouther Khazri. Il s'agit d'un vaste champ de recherche où peu de travaux internationaux existent. En particulier aucune base de formules mathématiques arabes hors ligne n'est aujourd'hui publiquement accessible. Les résultats actuels sont insuffisants dans le cadre d'une utilisation industrielle. Il est en effet nécessaire, d'une part d'obtenir de meilleurs

taux de reconnaissance, et d'autre part d'éviter une relecture fastidieuse et coûteuse de l'ensemble des formules.

L'un des grands problèmes qui peut gêner la reconnaissance de l'écriture mathématique arabe vient de l'incohérence de direction dans les flux d'écriture. Ceci est facile à constater dans le style occidental où le texte est naturellement écrit de droite à gauche, par contre le flux mathématique est dans la direction opposée. Ce phénomène, connu sous le nom de bidirectionnalité, introduit une série de difficultés supplémentaires à l'analyse de formules en particulier dans le cas de la reconnaissance hors ligne. La notation arabe pose d'autres problématiques. Il s'agit notamment de traitement d'une vaste collection de nouveaux glyphes, y compris les alphabets arabes de base ou les extensions de ces alphabets, en tenant compte des formes pointillés ou non pointillés des lettres et d'autres séries de chiffres. En outre, nous devons reconnaître tout un ensemble de symboles supplémentaires et ligatures pour certaines fonctions et opérateurs.

La problématique envisagée peut se définir selon deux niveaux :

- une tâche de segmentation et reconnaissance de symboles élémentaires que contient la formule. Il s'agit de regrouper les composantes connexes appartenant à un même symbole et de lui affecter une étiquette symbolique. A ce stade, il est nécessaire de préciser les hypothèses guidant cette étape de regroupement. Notons que la problématique de la reconnaissance de symboles est plus complexe que celle de caractères dans la mesure où le nombre de classes se trouve être relativement élevé.
- une tâche d'analyse et interprétation permettant de reconstruire la formule correspondante. Cette tâche va chercher à déduire de la disposition spatiale et de la taille relative de ces symboles une description structurée cohérente de la formule.

Ce travail est motivé principalement par l'importance des informations que peut contenir les formules mathématiques et l'utilité de rendre ces informations accessibles à tout le monde. Il porte sur la méthodologie proposée pour atteindre ce but par combinaison des résultats de reconnaissance des symboles mathématiques, contenus dans les formules, et leur analyse structurelle. Une première sous section permettra de prendre connaissance des difficultés liées à la reconnaissance de la notation mathématique arabe. Nous faisons, ensuite un survol des techniques de reconnaissance présentées dans la littérature. L'architecture ainsi que la méthode mise en oeuvre seront décrites en une troisième sous section. Une dernière sous section aura pour rôle de rendre compte, discuter des résultats obtenus et évaluer la qualité de la reconnaissance.

2.4.1 Formule mathématique arabe : spécificités et difficultés

Ci-après quelques spécificités et difficultés liées à la reconnaissance de formules mathématiques arabes :

- Bidirectionnalité : la formule mathématique est écrite suivant ou à l'opposé de la

2.4 CONTRIBUTION 3 : RECONNAISSANCE DE FORMULES MATHÉMATIQUES ARABES

direction droite à gauche de l'écriture arabe. Par conséquent, l'interprétation de la formule dépend de la direction dans laquelle elle est écrite.

- Le texte en arabe et la formule mathématique interagissent mutuellement via des mots de liaisons comme *حيث*, *غير ذلك*, *مع*, *اذا كان*.
- Un grand nombre et une grande diversité de signes [27] :
 - Des lettres de plusieurs alphabets (Arabe, Latin, etc.), en plusieurs styles (Naskh, Koufi, etc. pour l'Arabe, Romain, italique, etc. pour le Latin). Ces alphabets peuvent figurer en minuscule, majuscule, gras, etc. Les lettres sont généralement privées des signes diacritiques ce qui restreint le nombre des lettres disponibles pour composer des symboles.
 - Des chiffres du Maghreb (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) et du Machrek (٠, ١, ٢, ٣, ٤, ٥, ٦, ٧, ٨, ٩).
 - Des symboles spécifiques en plusieurs formes et tailles. La forme des symboles est soit les mêmes symboles que ceux utilisés en Romain d'usage courant (exemples : +, -), soit les mêmes symboles moyennant une inversion du sens (exemples : < et >, ∈ et ∃, etc.). La taille des symboles extensibles est fonction du contexte. Elle peut dépendre de la taille de l'expression couverte (exemple : symbole racine) ou de sa position, en indice ou en exposant.
- Utilisation de noms arabes pour les fonctions.
- Deux types de symboles : littéraux et en miroir. Les symboles de sommation (voir la Figure 2.16(a)), de produit (voir la Figure 2.16(b)), la limite (voir la Figure 2.16(c)) et de factoriel (voir la Figure 2.16(d)) sont présentés de deux manières.

Figure 2.16 illustrates mathematical symbols in Arabic script, showing both the original (littéraux) and mirrored (en miroir) versions for four operations: (a) summation and product, (b) summation and product, (c) limit, and (d) factorial.

FIGURE 2.16 – Symboles littéraux et en miroir.

- la cursivité de lettres arabes est également mise à profit, dans quelques cas, pour définir des variables de noms composés de plus d'une lettre (exemple *نق* désigne le rayon de cercle).
- De grands opérateurs étirés à la largeur des limites inférieures et supérieures (voir la Figure. 2.17). Parfois, ces limites ne correspondent pas exactement au-dessus ou en dessous de l'opérateur surtout en cas du manuscrit (voir la Figure 2.18).
- Différents styles d'écritures ce qui peut entraîner des formes incohérentes pour un

FIGURE 2.17 – Larges opérateurs étirés

FIGURE 2.18 – Formule manuscrite avec de larges opérateurs étirés.

même caractère ou symbole (voir la Figure 2.19). En outre, des variations aléatoires dans les formes sont rencontrées dans les différentes instances de caractères écrits par une même personne. Comme la notation mathématique exige un haut niveau de précision typographique, toute modification dans l'écriture des caractères ou symboles risque d'affecter le sens de la formule.

FIGURE 2.19 – Une même formule écrite par différentes personnes.

- De nombreux symboles peuvent se chevaucher, surtout en manuscrit. Des symboles brisés peuvent aussi apparaître (voir la Figure 2.20).

FIGURE 2.20 – Formules avec (a) symboles qui se touchent, (b) symboles brisés.

2.4.2 Etat de l'art

Certains chercheurs [28], [29] se sont intéressés à la reconnaissance en temps réel des formules mathématiques en utilisant des tablettes graphiques, ce qui simplifie en partie le problème en restituant le sens du tracé. Mais, rares ceux qui se sont penchés sur l'imprimé et/ou le manuscrit en hors ligne en utilisant un scanner ou une caméra pour la saisie des formules. Sans préjuger ici de la difficulté d'un cas par rapport à l'autre,

nous pouvons seulement constater que dans le cas de l'écriture en ligne, les résultats sont souvent meilleurs pour des conditions similaires d'expérimentations. Cela vient des informations temporelles qui fournissent des connaissances précieuses sur la dynamique et la morphologie de l'écriture. S'ajoute à cela, la majorité des travaux sont effectués sur certains types de formules avec un style et typographie spécifique. Ils ne peuvent pas gérer toutes sortes de formules. Ils reconnaissent généralement des équations simples, comme fait par [30], mais pas des matrices ou des systèmes d'équations.

Comme discuté par Blosteïn et Grbavec[31], un certain nombre de systèmes de reconnaissance de formules mathématiques latines obtiennent la structure de la formule sans analyse. A la place de l'analyse, quelques règles codées procéduralement sont utilisées. En effet, la plupart des méthodes d'analyse structurelle des formules mathématiques sont basées sur certains types de syntaxe définies implicitement ou explicitement. Pour les formules simples, les deux manières devraient bien fonctionner. La situation change si on essaye de reconnaître des formules plus complexes. Au lieu d'ajouter des règles codées procéduralement au système et rester incertain de l'exactitude du module d'analyse structurelle, des règles explicites appliquées par un analyseur devraient fournir une forme plus claire et plus concise pour la vérification formelle. Le problème de création d'un système qui soit à la fois efficace et suffisamment souple pour reconnaître des formules complexes reste un problème ouvert.

L'objectif de ce travail est de proposer un système efficace et souple pour la reconnaissance hors ligne de formules mathématiques arabes, fondé sur un analyseur syntaxique. En s'inspirant du travail de Belaïd et al.[32], nous avons proposé un système s'appuyant sur un analyseur syntaxique ascendant-descendant. Le système proposé commence l'analyse par la recherche de l'opérateur le plus important dans la formule et tente de partitionner la formule en sous-expressions qui seront analysées de la même manière. L'analyseur fonctionne en conjonction avec un système de reconnaissance de symboles comme il sera expliqué ci-dessous. L'avantage du système proposé c'est sa grande souplesse qui se traduit par sa capacité de gérer une grande variation dans l'emplacement des symboles. De plus, il permet la reconnaissance de formules linéaires et bidimensionnelles à structure complexe, impliquant plusieurs symboles, opérateurs explicites et implicites, nom de fonctions, etc.

2.4.3 Système proposé, dirigé par la syntaxe

Le système fonctionne suivant le schéma de la Figure 2.22. Il repose sur deux étapes majeures : la reconnaissance des symboles et l'analyse structurelle. La première étape convertit l'image de la formule en un ensemble de caractères et symboles. La seconde étape analyse l'arrangement spatial des symboles afin de rétablir l'information contenue dans la formule. Nous nous intéressons aux formules écrites selon le style oriental (voir la

Figure 2.21) où des symboles spécifiques sont alors d'usage et l'écriture des expressions symboliques suit le sens de l'écriture de la langue naturelle, de droite à gauche. C'est un système d'écriture, en conformité avec les normes et les conventions adoptées, qui sont en vigueur dans les manuels scolaires des pays du Moyen Orient, en Libye et en Algérie.

$$\left. \begin{array}{l} \text{مجموع} \quad \text{ص} \quad \text{ب} \quad \text{س} \quad \text{إذا كان} \quad \text{س} \quad > \quad . \\ \text{ب} = 1 \\ \text{أ} \quad \text{ب} \quad \text{س} \quad \text{و} \quad \text{س} \\ \text{ظا} \quad \pi \end{array} \right\} = \text{ت (س)}$$

$$\text{إذا كان} \quad \text{س} \quad \ni \quad \text{م} \quad \text{غير ذلك (مع } \pi \approx 3,141)$$

FIGURE 2.21 – Style oriental où on utilise des alphanumériques locaux non seulement pour les nombre et les variables mais aussi pour les noms de fonction et les opérateurs mathématiques.

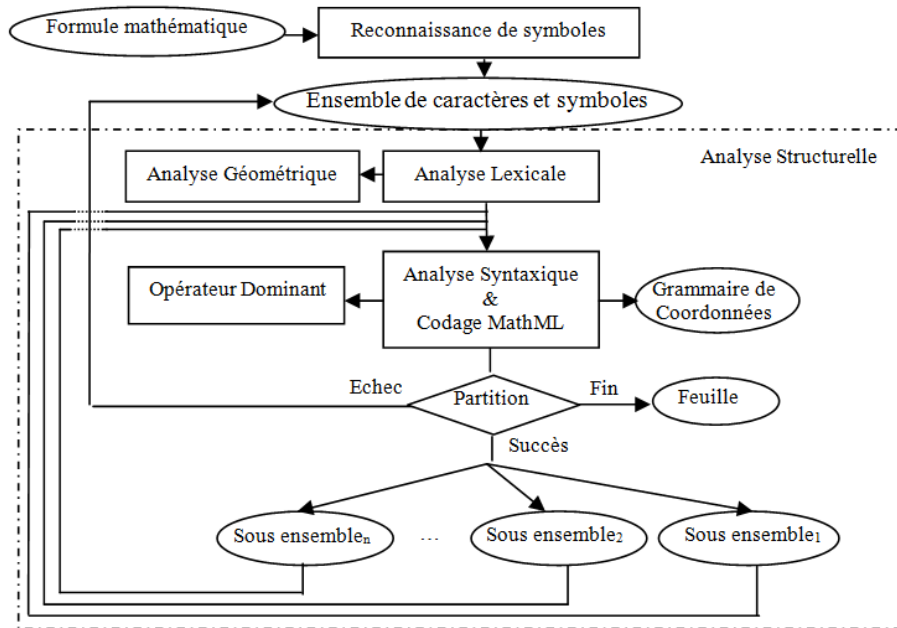


FIGURE 2.22 – Architecture du système proposé.

Pour reconnaître les symboles, nous avons eu recours au classifieur K^* en utilisant des descripteurs statistiques. Pour l'analyse structurelle, nous avons procédé par une analyse lexicale, géométrique et syntaxique à la fois ascendante et descendante en se basant sur une grammaire de coordonnées et les opérateurs dominants de la formule. Nous allons reprendre chacune de ces étapes, décrire son fonctionnement, les problèmes rencontrés ainsi que les propositions visant à leur résolution au moyen d'exemples de formules mathématiques.

Reconnaissance de symboles mathématiques isolés

Les caractères et symboles sont représentés par 30 descripteurs statistiques comme affiché dans la Figure 2.23. Cela inclut :

- L’histogramme de séquences de pixels noirs dans les directions horizontale, verticale, diagonale primaire et secondaire [33]. Ce descripteur peut effectivement discriminer les orientations, les surfaces et formes géométriques des symboles.
- Les moments de Hu et de Zernike qui sont utilisés pour capturer des descripteurs globaux d’une image.
- Co-occurrence à deux niveaux : il s’agit de compter le nombre de fois où une paire donnée de pixels se produit à une distance et une orientation fixe. Ce descripteur s’avert puissant pour extraire des informations de texture à partir d’une l’image en mettant en évidence la relation entre les pixels avoisinants [34].
- La portion de pixels blancs dans l’image.



```

Hu moments = [2.169548e-001 2.928672e-003 3.264625e-005]
Run length horizontal = [4 9 122 28 20]
Run length vertical = [6 6 32 105 33]
Run length positive diagonal = [11 47 93 96 23]
Run length negative diagonal = [45 37 26 32 132]
Central Zernikh moments = [6.568547e+001 4.667484e+001]
White pixel's portion = [5655]
Bi-level co-occurrence = [2208 2207 2116 2118]

```

FIGURE 2.23 – Vecteur des caractéristiques du nom de la fonction limite.

Nous avons créé une base de données composée de 50 classes de symboles avec 100 échantillons par classe. Pour la classification de symboles, nous avons testé avec plusieurs classifieurs (voir la Table 2.7) et nous avons trouvé que le meilleur classifieur est K^* . C’est un classifieur à base d’instances dans lequel la classe d’une instance de test est déterminée à partir des classe d’instances apprises qui lui est semblable selon une fonction de similarité à base d’entropie. L’intuition est que la distance entre les instances est définie comme la complexité de transformer une instance à une autre. L’utilisation de l’entropie comme une mesure de distance présente plusieurs avantages dont le plus saillant est la manipulation des attributs symboliques, des attributs à valeurs réelles et des valeurs manquantes. Pour plus d’informations sur K^* , se référer à [35]. Nous avons utilisé l’algorithme génétique pour la sélection des descripteurs pertinents et la validation croisée pour évaluer le classifieur. Nous avons observé une augmentation du taux de reconnaissance de 95.98% à 96.18% et une réduction dans le nombre de descripteurs : de

30 à 23. Nous avons également noté un abaissement dans le temps de classification : de 0.23s à 0.19s.

TABLE 2.7 – Evaluation de performance par classifieur.

Classifieur	F-Mesure(%)	Temps(s)
K^*	96.18	0.19
Arbre de décision	92.44	0.3
K-plus proches voisins	88.42	0.0037
Perceptron multi-couches	81.88	1.26
Naïf Bayes	72.92	0.0011

Bien que ce taux est assez élevé, l'échec du système de reconnaissance à distinguer entre certains symboles communs serait gênant pour les étapes ultérieures. En fait, certaines symboles distincts sont en étroite ressemblance morphologique tels le signe moins et la barre de fraction horizontale, les chiffres arabes deux et trois (٢, ٣), les lettres س et ص, la lettre ل et le chiffre un ١, le chiffre zéro et le point diacritique (٠, ٠), les parenthèses ouvrantes et fermantes, les opérateurs de comparaison (<, >), les chiffres arabes sept et huit (٧, ٨), etc. Nous estimons que certains de ces cas sont difficiles à discriminer par tout classifieur si le contexte n'est pas pris en compte. La résolution de tels cas de confusions pourrait se faire lors de l'analyse structurelle de la formule.

Analyse structurelle

L'analyse de la structure de la formule passe par une analyse lexicale, géométrique et syntaxique comme expliqué ci-après.

- **Analyse lexicale** : Cette analyse permet une abstraction supplémentaire dans le cadre de l'analyse structurelle, en créant des classes de symboles ayant un comportement identique (voir la Table 2.8). Un groupement vertical et/ou horizontal est nécessaire pour regrouper les lettres à leurs signes diacritiques : أ, ب, ت, etc., les chiffres en entiers ou en réels et les opérateurs composites : =, ≤, ≥, ⊆, ⊇. Pour évaluer l'efficacité de l'analyseur lexical à regrouper correctement ces symboles composites, nous avons généré une vérité de terrain, composée de 100 formules dont les symboles sont étiquetés à la main, et l'avons comparé aux résultats de groupement. Nous avons constaté que 99% des symboles composites ont été parfaitement regroupés.
- **Analyse géométrique** : A la suite de l'étape d'analyse lexicale, nous disposons d'un ensemble de symboles avec des attributs associés tels que son type lexical, son emplacement (coordonnées spatiales), son étendue (l'emplacement des ses opérandes) et son identité (résultat de la reconnaissance). Le but des analyses suivantes est de déterminer la structure hiérarchique des symboles, représentée souvent par un arbre d'analyse ou arbre de relations. Lors de l'analyse géométrique, nous avons

2.4 CONTRIBUTION 3 : RECONNAISSANCE DE FORMULES MATHÉMATIQUES ARABES

TABLE 2.8 – Etiquettes attribuées.

Type lexical	Désignation	Ensemble de symboles
SS	Symbole de Sommation	\neq, Σ
SP	Symbole de Produit	$\text{جد}, \Pi$
SI	Symbole d'Intégrale	\int
SR	Symbole de Racine	$\sqrt{2}$
BFH	Barre de fraction horizontale	$\frac{1}{1+2}$
DL	Délimiteurs	$(((((\dots))))))$
OP	Opérateurs	$-, =, <, <=, >, >=, \text{etc.}$
AR	Flèches	\rightarrow
Entier_non_signé	Entier non signé	$\text{٤١}, \text{٢٩٠٣}$
Réel_non_signé	Réel non signé	٠٣٦٠٥
Lettre	Lettre	$\text{ا}, \text{ب}, \text{etc.}$
FN	Fonction	$\text{طتا}, \text{طجا}, \text{جتا}, \text{طا}, \text{جتا}, \text{طا}, \text{جتا}, \text{حا}$
NF	Nouvelle Fonction	ق(س)
LM	Limite	نها

subdivisé l'espace entourant un symbole donné en plusieurs zones qui dénotent les relations spatiales que le symbole peut avoir avec son voisinage. Nous avons considéré dix relations spatiales : H (en haut), B (en bas), D (à droite), G (à gauche), EG (en exposant à gauche), ED(en exposant à droite), IG (en indice à gauche), ID (en indice à droite), I (inclus) et L (délimité). Notons que ces relations spatiales sont également utiles pour résoudre les cas de confusions entre symboles à morphologie similaire par la considération de leurs contexte. Par exemple, pour qu'un symbole soit considéré une barre de fraction horizontale au lieu d'un signe moins, il devrait avoir un numérateur et un dénominateur au-dessus et en dessous.

- **Analyse syntaxique** : Nous avons proposé une extension du formalisme grammair de coordonnées, initialement adopté par Anderson [36], dans le but de réduire les restrictions sur le positionnement de symboles et gagner en efficacité. Dans [36], Anderson a utilisé une approche descendante pour l'analyse des expressions mathématiques arithmétiques et matricielles latines les plus fréquemment utilisées. Pour améliorer cette analyse, nous avons eu recours à des connaissances sur les conventions de notation mathématique telles que la dominance et la priorité d'opérateurs pour éviter le retour arrière et gagner ainsi en efficacité. L'objet de cette analyse est donc d'introduire la connaissance a priori, déduite des règles de constitution des formules, afin d'améliorer la reconnaissance et de permettre au système de détecter avec, une grande certitude, les zones susceptibles d'être mal reconnues. Nous nous sommes inspirés aussi des travaux de [32] en adoptant une analyse syntaxique à la fois ascendante et descendante. L'idée est de subdiviser la formule en sous-expressions plus simples par l'opérateur dominant qui à partir duquel et de ses contextes (droit, gauche, haut, bas, en exposant, en indice, inclus et délimité), l'analyseur choisira la règle correspondante dans la grammaire. Ce processus se ré-

pète d'une manière récursive jusqu'à ce qu'ou chaque but soit atteint ou bien toutes les possibilités échouent. Dans ce qui suit, nous expliquons comment choisir l'opérateur dominant, introduire de la connaissance à priori et définir une formalisation grammaticale des formules mathématiques utiles à une meilleure reconnaissance.

– *Choix de l'opérateur dominant :*

1. Calculer pour chaque opérateur O combien de fois il a été dominé par les autres opérateurs. Un opérateur O_1 domine un opérateur O_2 si O_2 se trouve dans l'étendue de O_1 . Rappelons que l'étendue d'un opérateur correspond aux emplacements possibles de ses opérands. L'opérateur dominant sera donc le moins dominé.
 2. S'ils existent dans la même formule plusieurs opérateurs de même dominance, c'est l'opérateur le plus prioritaire qui l'emporte. Les opérateurs de comparaison ($=$, $<$, \leq , $>$, \geq , etc.) ont la plus haute priorité, puis les parenthèses qui ont une priorité supérieure à celle des opérateurs arithmétiques binaires ($+$ et $-$), puis le ($*$ et $/$), ensuite les opérateurs unaires ($+$, $-$ et \pm), après les indices et les exposants enfin les autres opérateurs, notamment les sommations, les produits, les intégrales, les racines, les fractions et les fonctions.
 3. S'il y a dans la formule plusieurs opérateurs de même dominance et de même priorité, l'opérateur le plus à droite sera l'opérateur de départ.
 4. Il faut vérifier aussi que l'opérateur dominant englobe, avec son étendue, la totalité de l'espace qu'occupe la formule. Au cas contraire, il s'agit d'une multiplication implicite entre deux sous-expressions et la formule sera partagée en deux, juste à la fin de l'étendue de l'opérateur dominant le plus à droite.
 5. S'il n'y pas d'opérateurs dans la formule, alors il s'agit d'une multiplication implicite entre variables, constantes entières ou réelles. Auquel cas, la formule sera partagée en deux selon les types lexicaux impliqués.
- *Procédure d'analyse :* Dans un contexte mathématique, des structures basiques sont très souvent utilisées ensembles de manière récursive pour former des structures plus complexes comme une fraction dans une intégrale avec la racine carrée d'un nombre. L'analyse de structure mathématique est assez complexe en raison de la multitude de combinaisons possibles de différentes structures qui peuvent avoir des relations horizontales, verticales et même diagonales. L'analyseur doit être capable de traiter un grand nombre de ces combinaisons. Pour cela, nous avons procédé par une analyse à la fois ascendante et descendante. L'analyseur ascendant cherche l'opérateur de départ. Ensuite, il sélectionne selon cet opérateur et son contexte, la règle correspondante dans la grammaire. Cette règle

2.4 CONTRIBUTION 3 : RECONNAISSANCE DE FORMULES MATHÉMATIQUES ARABES

donne des instructions à l'analyseur descendant pour partitionner la formule en sous-expressions qui seront analysées de la même façon. La stratégie du partitionnement de la formule est comme suit. Si la formule ou la sous-expression ne contient pas un opérateur de départ ou bien les conditions ne sont pas vérifiées, la règle ne s'applique pas. S'il existe une règle applicable, alors la division de la formule ou de la sous-expression se fait selon la position relative de l'ensemble de caractères et symboles par rapport à l'opérateur de départ sélectionné.

- *Grammaire de coordonnées* : Nous avons établi une trentaine de règles syntaxiques dont voici un exemple de règle relative à l'expression d'intégrale (voir la Table 2.9). Comme les mathématiques exploitent de manière intensive plusieurs modalités visuelles (formules, graphes, matrices, etc.) qui rendent particulièrement difficile pour une personne non voyante l'accès à ce savoir, nous avons cherché à utiliser une présentation des formules mathématiques qui lui soit réellement adaptée. Pour cela, nous avons procédé au codage MathML lors de l'analyse syntaxique des formules en insérant des actions sémantiques au niveau des règles. La Figure 2.24 montre un exemple illustratif d'analyse d'une formule.

TABLE 2.9 – Exemple de règle de la grammaire proposée et son codage MathML.

Règle	Contexte	Codage MathML
$R_{10} : E \rightarrow SI_S^N E^1 V \text{lettre}$	$S = \emptyset B(SI) R_S(SI) L_S(SI),$ $N = \emptyset A(SI) R_S(SI) L_S(SI),$ $L(SI) = E^1,$ $R(\text{lettre}) = E^1,$ $L(\text{lettre}) = V,$ $\text{lettre.code} = \langle d \rangle$	si($N \neq \emptyset$ and $S \neq \emptyset$) alors $E^0.\text{code} = V.\text{code} \langle mo \rangle \langle d \rangle \langle /mo \rangle$ $E^1.\text{code} \langle munsup \rangle S.\text{code} N.\text{code} \langle mo \rangle$ $\&\text{int}; \langle /mo \rangle \langle /munsup \rangle$ sinon si($S = N = \emptyset$) alors $E^0.\text{code} = V.\text{code} \langle mo \rangle \langle d \rangle \langle /mo \rangle E^1.\text{code} \langle$ $mo \rangle \&\text{int}; \langle /mo \rangle$ fin si

2.4.4 Expérimentations

Nous avons testé notre système sur 110 formules mathématiques arabes imprimées. Pour évaluer la performance du système, nous avons comparé sa sortie à une vérité de terrain. La Figure 2.25(b) présente le code MathML retourné par notre system pour la formule de la Figure 2.25(a). La Figure 2.25(c) affiche la vérité de terrain. Comme nous pouvons le constater, il y'a eu erreur de reconnaissance au niveau de la lettre arabe س qui a été confuse à la lettre ص .

Nous avons testé notre système sur différents types de formules : 1D (linéaires), $1D\frac{1}{2}$ (linéaires impliquant des indices, des exposants et des racines), 2D (des fractions) et $2D\frac{1}{2}$ (expressions d'intégrale, de sommation, de produits, etc.). La Table 2.10 présente

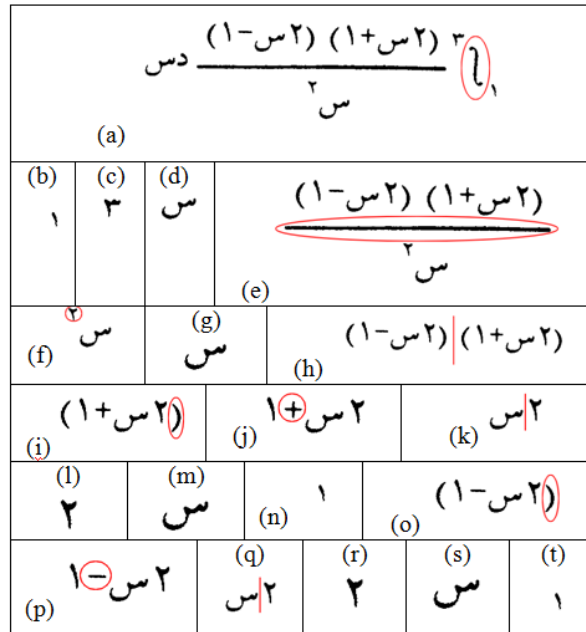


FIGURE 2.24 – Exemple d'analyse syntaxique d'une formule d'intégrale.

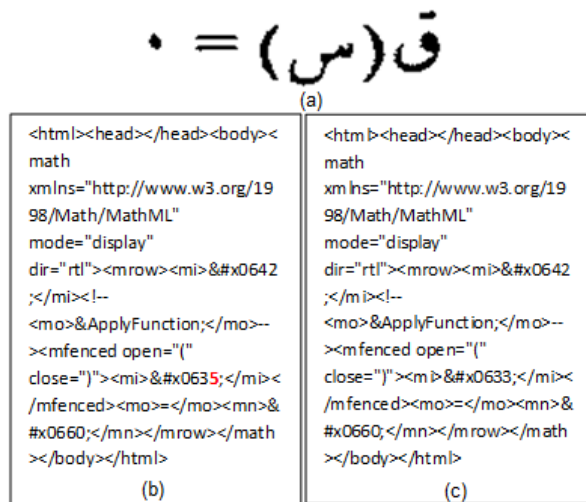


FIGURE 2.25 – Codage MathML retourné par notre système vs. vérité de terrain.

les résultats obtenus en se comparant à une vérité de terrain par calcul de la distance de Levenshtein. Un taux moyen de 91% est atteint.

En conclusion, l'objet de ce travail est de proposer des voies originales pour permettre une reconnaissance automatique de formules mathématiques arabes. Pour cela, nous avons présenté un système dirigé par la syntaxe qui reconnaît et retourne les résultats de la reconnaissance dans le format MathML. Un ensemble de règles de remplacement est défini par une grammaire de coordonnées pour analyser les formules. Cette grammaire

2.5 CONCLUSION

TABLE 2.10 – Taux de reconnaissance par type de formule.

Type de formule	Taux de reconnaissance
1D	92.3%
$1D\frac{1}{2}$	85.4%
2D	100%
$2D\frac{1}{2}$	100%

est employée en s'appuyant sur la reconnaissance de symboles et l'analyse de leur arrangement spatial et constitue une approche claire et bien structurée. Nous avons extrait des descripteurs statistiques et utilisé le classifieur K^* pour représenter et reconnaître les caractères et symboles mathématiques et un analyseur syntaxique ascendant-descendant qui repose sur la dominance d'opérateurs pour diviser récursivement la formule en sous formules plus simples.

2.5 Conclusion

Qu'il s'agisse d'écriture imprimée ou manuscrite, linéaire ou bidimensionnelle, pas de bonne reconnaissance sans caractérisation efficace. Dans ce domaine, les descripteurs peuvent être considérés comme un moyen permettant de caractériser et reconnaître l'écriture. Dès lors, il est nécessaire de définir de bons descripteurs lors du développement des systèmes de reconnaissance. Un bon descripteur doit posséder les propriétés suivantes : 1) bonne fidélité à la forme initiale, 2) bonne discrimination de formes différentes et 3) bonne adaptation aux opérations de reconnaissance de formes (invariance par translation, rotation et changement d'échelle). Les descripteurs sont généralement définies par expérience ou par intuition. Plusieurs types de descripteurs peuvent être extraits et combinés. Suivant les applications et techniques utilisées pour les systèmes, les descripteurs peuvent être très différents (globaux ou locaux, topologiques, structurels ou statistiques, de haut ou de bas niveau, etc.). Mais, l'objectif commun de tous les descripteurs est de représenter au mieux la forme de l'écriture afin de pouvoir distinguer si deux images appartiennent à deux classes différentes ou à la même classe. Ainsi, l'extraction des descripteurs doit être effectuée avec le plus grand soin, car les traitements ultérieurs ne vont plus manipuler l'image d'origine mais plutôt les résultats fournis par cette étape.

CHAPITRE 3

Contribution en identification du script et de sa nature

3.1 Introduction

Il existe de nombreux systèmes d'analyse de documents qui sont en mesure de gérer des documents écrits dans une seule langue. Mais il y'a une grande nécessité d'étendre ces systèmes pour gérer des documents multi-lingues. Pour développer de tels systèmes, le problème de l'identification du script doit être abordé. L'identification de la langue de l'écriture et de sa nature s'avère un problème complexe qui n'est pas encore entièrement résolu. Les travaux faits sur l'identification du script dépendent principalement des descripteurs extraits des images de documents au niveau du bloc, ligne ou mot. Notons que la plupart des systèmes existants identifient le script au niveau du bloc. Comme ils sont basés sur l'aspect visuel global du bloc textuel, ils sont généralement incapables de faire face aux variations dans le style de l'écriture, la taille des caractères, l'espacement entre les lignes ou mots, etc. Lorsque l'identification s'effectue par mots et non pas par lignes ou blocs, il serait possible d'analyser plus de documents où des mots en différentes langues peuvent être dispersés partout dans le document. Mais cela nécessite une analyse plus fine de chaque mot.

Nous proposons, dans le cadre de la thèse d'Asma Saïdani, l'étude de choix de descripteurs et classifieurs pour caractériser l'Arabe et le séparer du Latin. Il s'agit de discriminer entre les mots arabes et latins de natures imprimée ou manuscrite. Une première approche de cette étude nous a conduits à analyser et exploiter des méthodes d'extraction

3.2 SIMILARITÉS ET DIFFÉRENCES ENTRE L'ARABE ET LE LATIN

de descripteurs et de classification, connues de l'état de l'art. Une seconde approche de nos recherches a été d'exploiter la différence d'orientation des écritures arabe et latine. Des primitives spectrales (à base d'histogramme) locales utilisées pour extraire une information de direction des traits sont en particulier étudiées. Nous commençons par décrire les spécificités des scripts arabe et latin. Ensuite, nous étudions les descripteurs proposés dans la littérature. Puis, nous présentons les étapes d'extraction des descripteurs et de classification de mots du système proposé. Nous clôturons ce chapitre par comparaison de nos résultats à ceux de travaux connexes.

3.2 Similarités et différences entre l'Arabe et le Latin

La Figure 3.1 illustre les différents types d'écriture arabe et latine, manuscrite et imprimée.

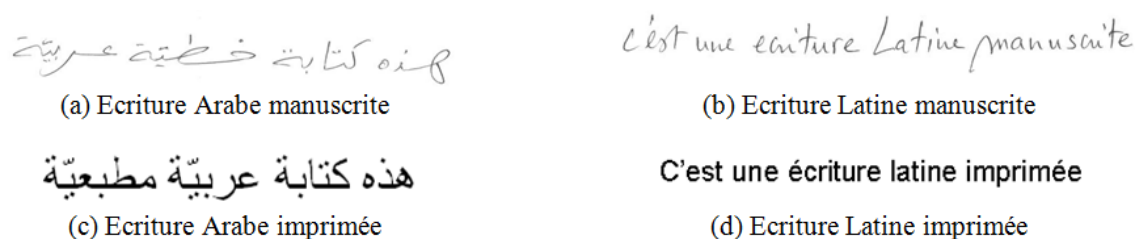


FIGURE 3.1 – Types d'écritures arabe et latine.

Ces écritures ont les points suivants en commun :

- **Présence de lignes d'écriture** (voir la Figure 3.2).



FIGURE 3.2 – Lignes d'écriture.

- **La bande centrale** est généralement la plus chargée en terme de densité d'informations en pixels (voir la Figure 3.3). Elle correspond aux ligatures horizontales, aux lettres centrales (sans extensions) et aux boucles.



FIGURE 3.3 – Bande centrale.

- **Régularités et singularités** : l'écriture arabe comme latine, présente des singularités au-dessus de la ligne de base, tandis que les régularités sont au-dessous (voir la Figure 3.4).



FIGURE 3.4 – Régularités et singularités par rapport à la ligne de base.

- **Variabilité inter-scripteurs et intra-scripteur** : L'écriture dépend du style de l'écriture du scripteur (voir la Figure 3.5) et de son état émotionnel (joie, nervosité, peur, etc.).

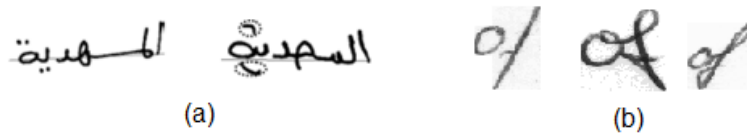


FIGURE 3.5 – Le même mot écrit par différents scripteurs

En revanche, l'écriture arabe se distingue de celle latine en plusieurs points :

- **Orientation de l'écriture** : Les mots en Arabe sont écrits de droite à gauche.
- **Projection horizontale** : Comme illustré dans la Figure 3.3, le profil de la projection horizontale de textes arabes a un seul pic autour du centre de la ligne textuelle. Mais, le profil de la projection horizontale de textes latins présentent généralement deux pics majeurs. En effet, la distribution spatiale des caractères latins est répartie dans les zones moyenne et supérieure. Seuls quelques caractères minuscules sont répartis dans la zone inférieure.
- **Alphabet** : L'alphabet arabe comporte 28 lettres sans considérer la variation de leur forme selon leur position, les éléments de voyellation et le contexte phonétique.
- **Formes de lettre** : La forme d'une lettre varie selon sa position dans le mot : initiale, médiane, finale ou isolée. Par exemple, la lettre ح a quatre formes différentes : isolée dans صباح, au début dans حلیم, au milieu dans رحمان et à la fin dans فتح. Toutes les lettres arabes sont les mêmes. Il n'y a pas de distinction entre lettres imprimées et cursives comme c'est le cas pour le Latin. Certaines lettres se connectent d'une seule côté et d'autres des deux côtés. Dans une même position, la forme de certaines lettres varie aussi en fonction de son contexte phonétique.
- **Points diacritiques** : Il existe des lettres différentes qui ont la même forme, mais qui se distinguent par la position et le nombre de points qui leur appartiennent. La moitié des lettres arabes contient des points diacritiques (un, deux ou trois). Ces points sont situés au-dessus ou au dessous de la lettre, mais jamais les deux simultanément.

3.3 ETAT DE L'ART

- **Mot** : Les lettres sont reliées ensemble pour former les mots de manière unique.
- **Écriture semi cursive** : L'écriture arabe est semi-cursive qu'elle soit imprimée ou manuscrite. Les lettres sont liées généralement entre elles. Le mot peut être composé d'un ou plusieurs PAWs. Chaque PAW est une séquence de lettres complètement cursives. Les PAWs sont séparés par quelques blancs et ils ne sont pas composés nécessairement par le même nombre de lettres (voir les Figure 3.6).

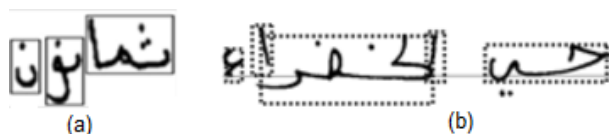


FIGURE 3.6 – PAWs en écriture arabe.

- **Ligatures** : Les lettres d'un PAW, peuvent être ligaturées horizontalement ou verticalement (voir la Figure 3.7). Dans certaines fontes ou styles d'écritures, deux, trois ou même quatre lettres peuvent être ligaturées verticalement.



FIGURE 3.7 – Ligatures horizontales et verticales des lettres en mots arabes.

- **Elongations** : En Arabe, le même mot n'a pas une longueur fixe puisque différents nombre d'élongations peuvent apparaître entre les lettres (voir la Figure 3.8).



FIGURE 3.8 – Elongations de l'écriture.

Pour l'écriture arabe manuscrite, d'autres caractéristiques interviennent :

- **Fusion de points diacritiques** : Deux ou trois points diacritiques peuvent facilement être agglomérées en deux ou même en un seul point (voir la Figure 3.5).
- **Chevauchement de PAWs** : Les PAWs peuvent se chevaucher comme illustré dans la Figure 3.6(b).

3.3 Etat de l'art

Les systèmes d'identification du script et de sa nature sont résumés dans la Table 3.1. Ils concernent principalement la différenciation d'écritures arabe et latine de natures imprimée et manuscrite.

TABLE 3.1 – Synthèse des systèmes d'identification

Ref.	Script	Nature	Base	Niveau	Taux d'identification
[37]	Arabe/Latin	Imprimé/Manuscrit	400 Blocs	Bloc/Ligne	88.5%
[38]	Arabe/Latin	Imprimé/Manuscrit	800 Documents	Bloc	84.75%
[39]	Arabe/Latin	Imprimé/Manuscrit	800 Mots	Mot	97.5%
[40]	Arabe/Anglais	Imprimé	1976 Lignes, 8320 Mots	Ligne/Mot	99.7%(Ligne), 96.8%(Mot)
[41]	Farsi_Arabe	Imprimé/Manuscrit	32006 Mots	Mot	97.1%
[42]	Latin	Imprimé/Manuscrit	50 Documents	Ligne	98.2%
[43]	Bangla/Anglais	Manuscrit	1200 Blocs	Bloc	95%
[44]	Arabe/Latin	Imprimé/Manuscrit	400 Blocs	Bloc	95%
[45]	Latin	Imprimé/Manuscrit		Mot	>80%
[46]	Arabe/Latin	Imprimé	Apprentissage : 3383 Mots, Test : 846 Mots	Mot	94.32%

3.4 Approche proposée

Différents types de descripteurs, proposés dans la littérature et dans cette étude, sont extraits et testés pour illustrer leurs propriétés et performances. L'analyse des résultats obtenus nous a conduit à exploiter la différence d'orientation des écritures arabe et latine. Des descripteurs à base d'histogramme des orientations de gradients sont alors proposés pour extraire l'information sur la direction des traits.

3.4.1 Etude de descripteurs

Ci-dessous la liste de descripteurs étudiés :

- **Variance de la projection verticale** : C'est une mesure d'homogénéité du profil de la projection. Ce descripteur permet d'en savoir plus sur l'étalement horizontal du mot [45]. Comme mentionné par Baāti et al. dans [38], en raison des chevauchements entre les lettres ou PAWs des mots manuscrits, les profils de projection ont des vallées et des pics lisses provenant de petites variances en comparaison aux mots imprimés.
- **Largeur, hauteur, ratio d'aspect, surface et densité des composantes connexes** : Comme souligné par Ben Jlaïel et al. dans [44], les tailles des composantes connexes des mots imprimés sont plus conformes, entraînant des variances d'hauteurs et de largeurs plus petites. Selon nous, ce type de descripteurs reflète les différences d'apparence visuelle et de structure. Ils s'utilisent lorsque les formes à étudier sont très irrégulières. Mais ils sont invariants en translation uniquement. Ils peuvent être utiles pour discriminer entre mots imprimés et manuscrits latins uniquement, mais non pour l'Arabe en raison de son aspect cursif et la possibilité d'élongation.
- **Longueur de séparateurs entre paires de composantes connexes successives** : En Latin imprimé, les composantes connexes sont séparées régulièrement

3.4 APPROCHE PROPOSÉE

comme noté par Ben Jlaiel et al. [44], ce qui n'est pas le cas dans le manuscrit.

- **Analyse des profils des composantes connexes** : L'analyse du profil inférieur des composantes connexes a été faite par Zhou et al. [43] dans le but de mesurer la discontinuité de l'écriture et séparer entre Bangla et le Latin. Sachant que l'Arabe et le Latin manuscrits ainsi que l'Arabe imprimé sont tous cursifs, alors que le Latin imprimé ne l'est pas, nous avons mesuré la discontinuité de l'écriture, dans un travail antérieur [47]. Nous avons constaté que l'écriture arabe est moins discontinue que celle du Latin. De plus, elle n'a pas des liens élevés entre ses lettres, comme c'est le cas du Latin (*o, v, etc.*) qui augmentent les différences entre les coordonnées des pixels du profil inférieur des mots latins.
- **Ratio d'aspect des boucles** : Le script imprimé est une succession de composantes connexes pouvant contenir des boucles dont les surfaces sont régulières à la différence de celles de l'écriture arabe ou latine manuscrite qui dépendent du style de l'écriture [37]. La largeur du rectangle englobant la boucle divisée par son hauteur est prise ainsi comme descripteur, généralisé en terme de moyenne et écartype.
- **Distribution des pixels** : Ce descripteur indique s'il y a une grande différence de répartition des pixels entre la moitié supérieure et la moitié inférieure du mot. Il est particulièrement utilisé pour discriminer entre le manuscrit et l'imprimé Latin dans [42].
- **Descripteurs issus du profil de la ligne de base** : Dans les mots imprimés, les positions des ascendants et descendants sont déterminées en fonction de la ligne de base. Mais, en cas du manuscrit, les mots ne sont pas toujours écrits selon une seule ligne de base et les positions des ascendants et descendants varient selon le style de l'écriture. Pour différencier entre des mots imprimés et manuscrits on a utilisé, dans [41], des descripteurs issus du profil de la ligne de base comme la position de la ligne de base, le nombre n de sous lignes de base, la distance d_1 du haut de l'image à la ligne de base, la distance d_2 du bas de l'image à la ligne de base et le nombre p des pixels sur la ligne de base. La ligne de base est estimée par le pic de l'histogramme horizontal du mot (voir la Figure 3.9).

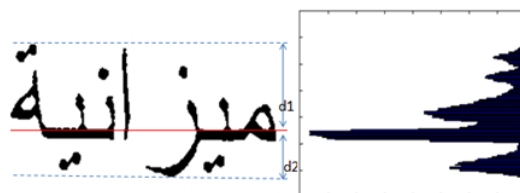


FIGURE 3.9 – Descripteurs issus du profil de la ligne de base.

- **Histogramme à base de plage** : Ce descripteur tient compte du nombre de pixels noirs successifs qu'on peut trouver dans plusieurs directions (horizontale, verticale

et/ou diagonale). Zheng et al. [48] ont utilisé ce descripteur pour la classification imprimé/manuscrit des caractères chinois. Notons que ce descripteur peut être utilisé pour souligner la différence entre les longueurs des graphèmes. En considérant ce descripteur pour la discrimination de mots arabes et latins de natures imprimée et manuscrite, nous avons déduit que les histogrammes à base de plages pour les mots de même nature (imprimée ou manuscrite), qu'ils soient arabes ou latins, sont similaires. Nous avons constaté aussi que les histogrammes de mots manuscrits sont décalés à droite ce qui signifie que les segments verticaux (associés aux hampes et jambages) sont plus étendus en manuscrit qu'en imprimé.

- **Histogramme de transitions** : Il s'agit de compter le nombre de fois que la valeur du pixel passe de 0 à 1 en suivant un axe donné (horizontal et vertical). Ce descripteur a été utilisé par Mozaffari and Bahar dans [41], pour distinguer entre les mots manuscrits et imprimés écrits en Farsi_Arabic. Il permet de mesurer la complexité des graphèmes.
- **Co-occurrence à deux niveaux** : Le nombre de fois qu'un motif apparaît est calculé. Un motif est une paire de pixels blanc - blanc, blanc - noir, noir - blanc, noir - noir selon une distance entre pixels et un axe donné. Nous avons retenu que les paires de pixels noirs successifs que nous considérons porteuses d'informations. Les directions choisies sont les axes horizontal, vertical et les deux diagonales. Nous avons constaté que ce descripteur permet de distinguer entre l'imprimé et le manuscrit qu'il s'agit de mots arabes ou latins, principalement selon l'axe horizontal et les deux diagonales.
- **Profil haut/bas** : Comme noté par Kavallieratou et al. [42], la hauteur des caractères imprimés est plus ou moins stable au niveau de lignes textuelles. En revanche, la distribution de la hauteur des caractères manuscrits est variable. Ces remarques s'appliquent également pour la hauteur du corps principal des caractères ainsi que la hauteur des ascendants et descendants. Pour caractériser un mot à la base de son profil haut et bas, nous avons calculé : le ratio entre la zone de l'ascendant et celle du corps principal du caractère, le ratio entre la zone du descendant et celle du corps principal du caractère et le ratio entre la surface et la valeur maximale de l'histogramme horizontal du profil haut/bas. La Figure 3.10 présente les résultats obtenus pour le mot arabe imprimé ميزانية.
- **Grandeurs physiques du mot** : Dans [41], Mozaffari et Bahar ont noté que les tailles des mots imprimés sont plus conformes que celles des mots manuscrits. Ainsi, des descripteurs en rapport avec les grandeurs physiques du mot tels que la densité de pixels noirs, la largeur, la hauteur, le ratio d'aspect et la surface sont considérés.
- **Recouvrement** : Dans l'écriture manuscrite, les rectangles englobant les composantes connexes ont tendance à se chevaucher. Cela ne se produit pas en imprimé.

3.4 APPROCHE PROPOSÉE

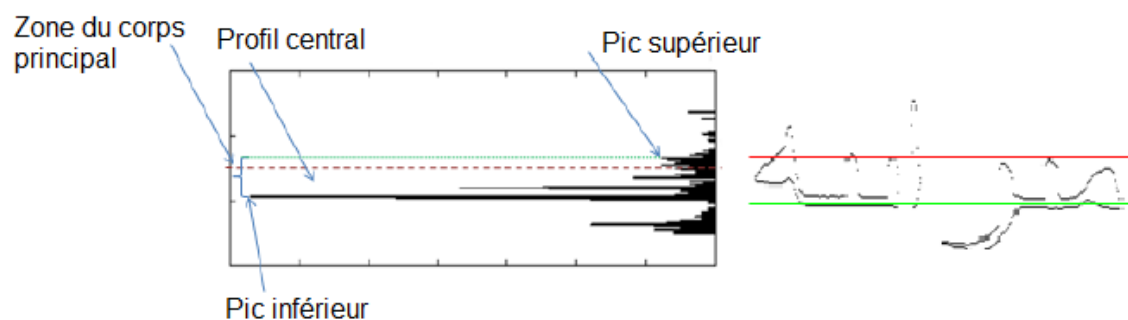


FIGURE 3.10 – Profil haut/bas du mot arabe imprimé : ميزانية.

- **Moments** : De simples propriétés de l'image du mot peuvent être déterminées via les moments comme la surface, son centroïd et des informations concernant son orientation. Nous avons considéré en particulier, les moments centrés qui traduisent l'influence qu'ils exercent en fonction de leur éloignement par rapport au centre de gravité.
- **Filtres de Gabor** : Dans [49], les filtres de Gabor ont été appliqués et 16 primitives de texture ont été extraites pour identifier le script (Anglais ou Chinois) des mots imprimés. Les résultats obtenus confirment la capacité des filtres de Gabor de capturer les caractéristiques du script. Mais, ils sont très sensibles au bruit et s'appliquent sur de grandes images. Ces filtres ont été utilisés aussi par Baâti et al. [38] pour discriminer entre blocs textuels imprimés et manuscrits écrits en Arabe et en Latin. Notons que les mots possèdent des structures géométriques très marquées. Une étude des transformées existantes pour les images des mots montre que peu d'entre elles (les ondelettes, les filtres de Gabor, etc.) permettent de représenter efficacement ces structures géométriques des images. Comme notifié par [9], malgré leurs facilité d'utilisation et rapidité d'exécution, ces filtres s'avèrent insuffisants pour la localisation des fréquences extrêmes. De telles fréquences renseignent sur les contours des dessins des caractères.
- **La transformée en pyramides orientables** : Cette transformée permet d'analyser le contenu de l'image dans de nombreuses directions et à différents niveaux de résolution. Pour cela, elle utilise des filtres orientables dans différentes directions en assurant ainsi une invariance par rapport à la rotation. Par cette transformée, l'image est subdivisée en collection de sous bandes localisées à différentes échelles et orientations [50]. Des descripteurs extraits des sous bandes de la pyramide ont servi, dans [39] à classifier les scripts. Le vecteur des caractéristiques inclut la moyenne, l'écartype, etc. calculés à partir de la matrice de co-occurrences. Bien qu'efficace, l'analyse par filtres orientables présente une importante redondance d'informations dans les différents niveaux des échelles de décomposition, ce qui pénalise considé-

ablement son coût de codage.

- **Primitives structurelles** : Quelques caractéristiques structurelles ont été déjà utilisées dans [46] pour identifier les scripts arabe et latin imprimés. Ce type de descripteurs ont l'avantage non seulement de représenter les mots avec une grande tolérance aux distorsions et aux variations de styles, mais aussi de refléter les aspects intuitifs de l'écriture, comme les boucles, les ascendants, les descendants, etc. Nous proposons, dans cette étude, d'autres primitives structurelles notamment :
 - **Position des points diacritiques** : Cette primitive est proposée en raison de l'absence des points diacritiques en bas des mots latins, ce qui n'est pas le cas pour les mots arabes (voir la Figure 3.11).



FIGURE 3.11 – Positions des points diacritiques.

- **Position des boucles** : Les boucles dans les mots arabes sont généralement présentes au niveau de la bande centrale à l'exception de la lettre ذ dans laquelle les boucles débordent légèrement en dessus et en dessous. En revanche, il y a beaucoup de lettres latines dont les boucles se trouvent au dessus ou en dessous de la bande centrale (voir la Figure 3.12).

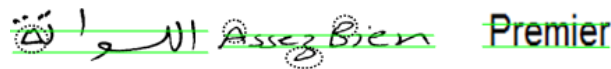


FIGURE 3.12 – Positions des boucles.

- **Forme des jambages** : L'écriture arabe, qu'elle soit imprimée ou manuscrite, est caractérisée par des jambages allongés. Dans l'écriture latine, les jambages sont plutôt étendus (voir la Figure 3.13).

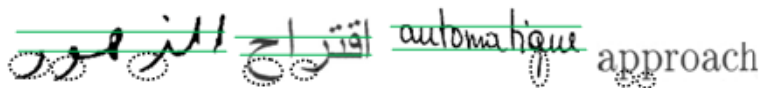


FIGURE 3.13 – Formes des jambages.

En combinant tous ces descripteurs, un ensemble de 126 descripteurs est formé. Comme certains de ces descripteurs pourraient être inutiles, voire redondants au système d'identification, leur pertinence doit être analysée. Pour cela, nous avons appliqué plusieurs méthodes de sélection des descripteurs. Cela a permis de réduire le nombre des

3.4 APPROCHE PROPOSÉE

descripteurs à 25% (de 126 à 58) tout en améliorant la performance en taux d'identification du script et de sa nature. La Table 3.2 présente les résultats obtenus avec différentes méthodes de sélection de descripteurs en utilisant le classifieur AODEsr. Elle indique aussi le temps de classification.

TABLE 3.2 – Test avec différentes méthodes de sélection de descripteurs.

Méthode de sélection	Descripteurs sélectionnés	Taux d'identification (%)	Temps(s)
Algorithme génétique	58	98.72	0.03
Analyse en composantes principales	48	95.69	0.02

Les premières expériences ont été menées en utilisant deux bases de données publiques : la base IAM (de mots latins manuscrits) et la base IFN/ENIT (de mots arabes manuscrits). Pour les mots arabes et latins imprimés, nous avons construit nos propres bases de données. Les mots sont extraits de magazines, journaux et autres documents de différents styles, fontes et tailles de polices. Une résolution de 300 points par pouce est utilisée pour numériser les mots. L'ensemble d'apprentissage et de test inclut 1720 échantillons (430 par classe) avec un nombre égal de mots arabes imprimés (AI), arabes manuscrits (AM), latins imprimés (LI) et latins manuscrits (LM).

Comme notre base de départ est assez réduite en nombre, nous avons utilisé la validation croisée comme méthode d'estimation de fiabilité du classifieur fondé sur une technique d'échantillonnage. Le classifieur est alors bâti sur l'échantillon d'apprentissage et validé sur l'échantillon de test. Les mots sont partagés en dix ensembles disjoints que nous les apprenons sur neuf ensembles puis nous validons le classifieur sur l'ensemble restant et nous répétons cette opération dix fois en calculant la moyenne des taux d'identification obtenus de chaque itération.

Les résultats dans la Table 3.3 montrent que le taux d'identification moyen est environ 98.72% pour les mots manuscrits et imprimés qu'ils soient écrits en Arabe ou en Latin. Notons qu'en utilisant les descripteurs sélectionnés, les mots arabes et latins imprimés ont été identifiés de manière fiable avec un taux d'identification dépassant les 99%.

TABLE 3.3 – Taux moyen d'identification par classe.

	PA	HA	PL	HL	Moyenne
Taux d'identification (%)	99.1	98.4	99.7	97.8	98.7

En analysant la matrice de confusion (voir la Table 3.4), nous avons remarqué que la plupart des erreurs d'identification sont dues aux confusions entre mots latins et arabes manuscrits à cause de leur nature cursive. Nous avons noté aussi que des confusions entre Latin imprimé et manuscrit sont dues à certains styles d'écriture dans lesquels il n'y a pas de ligatures entre lettres.

TABLE 3.4 – Matrice de confusion.

	AI	AM	LI	LM
AI	430	0	0	0
AM	3	420	0	7
LI	0	0	428	2
LM	5	4	1	420

Nous avons comparé également les performances de trois classifieurs typiques : un classifieur bayésien (AODEsr), k plus proches voisins et l’arbre de décision (voir la Table 3.5). Nous avons constaté que le classifieur bayésien, avec les descripteurs sélec-

TABLE 3.5 – Evaluation de performance par classifieur.

Classifieur	F-Mesure(%)	Temps (s)
Classifieur bayésien	98.72	0.03
Arbre de décision	85.98	0.14
k plus proches voisins	97.5	0.01

tionnés, capture une grande partie des différences entre mots arabes et latins, imprimés ou manuscrits fournissant ainsi une bonne solution au problème d’identification du script et de sa nature.

Dans le but d’améliorer le taux d’identification, nous avons observé les cas de confusion, surtout entre scripts manuscrits. Nous avons remarqué qu’il est possible de les résoudre en exploitant avantageusement la différence d’orientation entre scripts arabe et latin (voir la Figure 3.14). Pour cela, nous avons eu recours à une méthode d’analyse de texture, ensuite aux descripteurs de formes à base de d’histogramme des orientations de gradients qui tiennent compte de cet aspect.

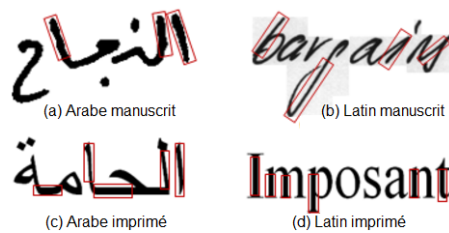


FIGURE 3.14 – Différences d’orientation dans les écritures arabe et latine de natures imprimée et manuscrite.

3.4.2 Méthode des longueurs de plages de niveaux de gris

Cette méthode analyse la texture du mot caractérisée par l’arrangement plus ou moins régulier de ses primitives dans différentes directions. Elle consiste à compter le nombre de plages d’une certaine longueur j , de niveau de gris i dans une direction q donnée. Nous

3.4 APPROCHE PROPOSÉE

avons considéré que les pixels noirs. A chaque direction correspond donc un vecteur (voir la Figure 3.15). De ces vecteurs, nous avons extrait les descripteurs suivants :

- Le “poids” des plages courtes, traduit en Anglais par Short Run length Emphasis (SRE) rend compte de la finesse d’une texture. Plus l’image contient des plages de petite taille, plus SRE est élevé.
- Le “poids” des plages longues, traduit en Anglais par Long Run length Emphasis (LRE) rend compte de la grossièreté de la texture de l’image. Plus l’image contient des plages de grande taille, plus le LRE est élevé.
- La distribution des niveaux de gris ou Grey Level Distribution (GLD) permet d’appréhender l’uniformité des plages dans les différents gris. GLD augmente avec le nombre de plages ayant des niveaux de gris identiques.
- La distribution des longueurs de plages ou Run Length Distribution (RLD) mesure la non uniformité de la répartition des longueurs de plages, indépendamment de leur niveau de gris. RLD est minimum lorsque les plages sont également distribuées entre les longueurs.

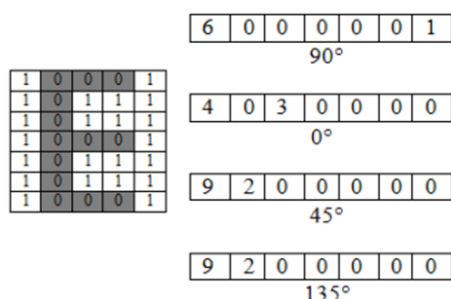


FIGURE 3.15 – Représentation d’une image par les longueurs de plages de pixels noirs.

En comparant les performances d’AODEsr, SVM et 1-PPV, combinés aux descripteurs issus de l’analyse de texture du mot, nous avons trouvé que le classifieur 1-PPV aboutit au meilleur résultat (voir la Table 3.6).

TABLE 3.6 – Evaluation par classifieur.

1-PPV	99.03%
SVM	96.03%
AODEsr	98.56%

3.4.3 Descripteurs à base d'histogramme des orientations de gradients

Un histogramme des orientations de gradients (HOG) est une caractéristique utilisée en vision par ordinateur pour la détection d'objets. Il s'est montré particulièrement efficace pour la détection de personnes [51]. L'idée importante derrière le descripteur HOG est que l'apparence et la forme locale d'un objet dans une image peuvent être décrites par la distribution de l'intensité du gradient ou la direction des contours. Ceci peut être fait en divisant l'image en des régions adjacentes de petite taille, appelées cellules, et en calculant pour chaque cellule l'histogramme des directions du gradient ou des orientations des contours pour les pixels à l'intérieur de cette cellule. La combinaison des histogrammes forme alors le descripteur HOG (voir la Figure 3.16).

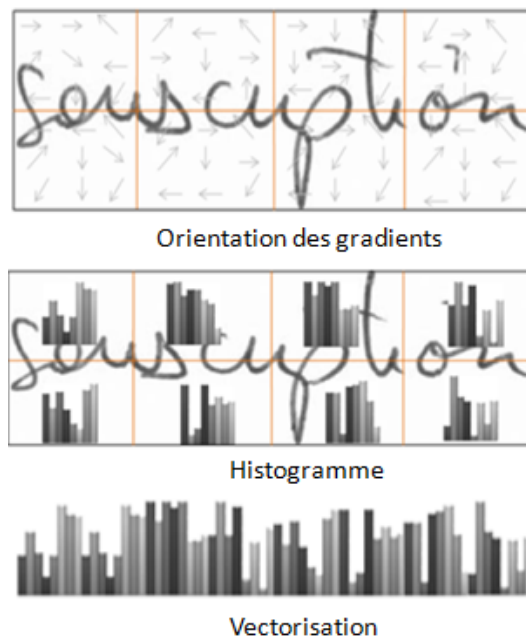


FIGURE 3.16 – Calcul du descripteur HOG.

Descripteur HOG basique

Pour construire ce descripteur, nous avons suivi les étapes suivantes :

- **Calcul de l'image gradient** : Le gradient sert à calculer la variation de l'intensité des pixels dans différentes orientations. Il correspond à la première dérivée (contour horizontal ou vertical) de l'image. La méthode la plus courante pour calculer le gradient consiste à appliquer un filtre dérivatif 1-D centré, dans les directions horizontale et verticale (voir la Figure 3.17).

3.4 APPROCHE PROPOSÉE

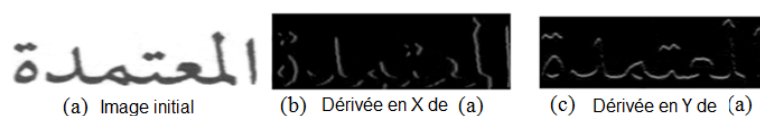


FIGURE 3.17 – Calcul de l'image gradient.

- **Construction de l'histogramme** : Elle est faite dans des cellules carrées de petite taille (de 4×4 à 12×12 pixels). Chaque pixel de la cellule vote alors pour une classe de l'histogramme, en fonction de l'orientation du gradient à ce point. Le vote du pixel est pondéré par l'intensité (ou magnitude) du gradient en ce point. Les histogrammes sont uniformes de 0° à 180° (cas non signé) ou de 0° à 360° (cas signé).
- **Formation et normalisation des blocs** : Pour de meilleurs résultats, les histogrammes locaux sont normalisés en contraste, en calculant une mesure de l'intensité sur des zones plus larges que les cellules, appelées des blocs, et en utilisant cette valeur pour normaliser toutes les cellules du bloc. Cette normalisation permet une meilleure résistance aux changements d'illuminations et aux ombres.

Le descripteur HOG maintient quelques avantages clés par rapport aux autres descripteurs. Il est facile à mettre en oeuvre, caractérisé par une grande robustesse face à la variabilité intra-classe et assez rapide pour la classification en temps réel. Etant basé sur une description locale de l'image, HOG offre une analyse plus fine du mot. Au lieu d'avoir un descripteur unique pour tout le mot, on aura un ensemble de descripteurs qui représentent ses éléments constitutifs. Ce descripteur est invariant aux changements d'échelle ce qui signifie que les mots peuvent être classés sans qu'il soit nécessaire de passer par une étape de normalisation de la taille. Ainsi le descripteur HOG est devenu l'un des descripteurs les plus populaires de représentations de bas niveau d'image en vision par ordinateur. Toute amélioration, aussi petite qu'elle soit, de sa capacité à représenter des images serait utile. Dans la suite, nous examinons plusieurs façons d'améliorer HOG au minimum coût de performance.

Descripteur HOG basique amélioré

En remarquant que la distribution des pixels diffère dans différentes parties du mot, nous avons divisé différemment l'image en une grille de cellules rectangulaires : 1) trois bandes horizontales égales (voir la Figure 3.18), 2) bandes supérieure, centrale et inférieure (voir la Figure 3.19), 3) cellules sans chevauchement (voir la Figure 3.20) et 4) cellules en chevauchement (voir la Figure 3.21). Nous avons calculé un HOG pour chaque cellule séparément, ensuite nous avons enchaîné les HOGs pour obtenir un descripteur composite.

Généralement la zone relative à la bande centrale du mot présente une plus grande

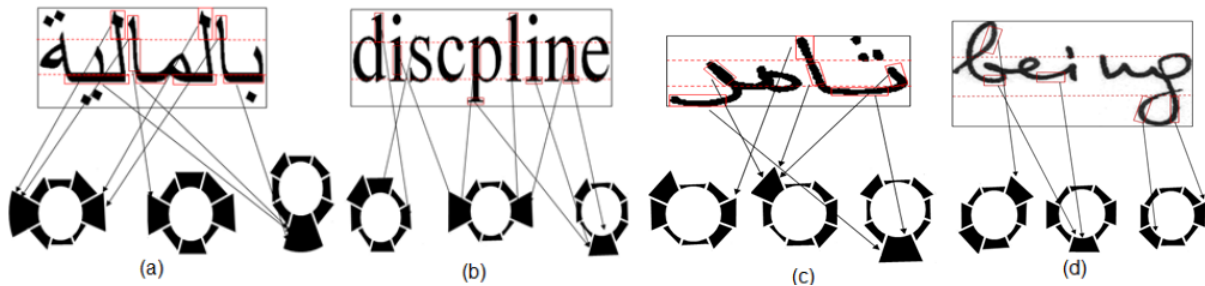


FIGURE 3.18 – HOGs, selon trois bandes horizontales égales d’un (a) Mot arabe imprimé, (b) Mot latin imprimé, (c) Mot arabe manuscrit et (d) Mot latin manuscrit.

variabilité, la densité d’information y est très importante car la morphologie de la chaîne dans cette bande est complexe à cause des boucles, quand elles existent, des ligatures horizontales et verticales. Pour cela, nous avons divisé l’image du mot en trois cellules par rapport à la bande centrale du mot. Nous avons utilisé la méthode de projection horizontale pour extraire la bande centrale et déduire les bandes inférieure et supérieure du mot. Comme la bande centrale contient aussi la plus grande quantité d’information, nous l’avons segmenté en cinq cellules égales (voir la Figure 3.19).

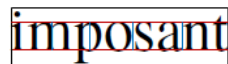


FIGURE 3.19 – Division de la bande centrale en 5 cellules.

Pour la disposition de cellules sans chevauchement, nous avons également essayé de varier le nombre de cellules dans la direction horizontale n_x et verticale n_y en fonction du nombre total de cellules $n_x \times n_y$ et de voir son impact sur le taux d’identification du script. Les dispositions possibles de six cellules sont affichées dans la Figure 3.20.

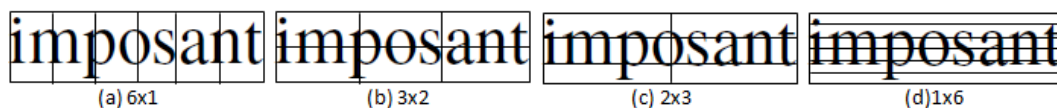


FIGURE 3.20 – (a) 6 colonnes \times 1 ligne, (b) 3 colonnes \times 3 lignes, (c) 2 colonnes \times 3 lignes et (d) 1 colonne \times 6 lignes.

3.4 APPROCHE PROPOSÉE

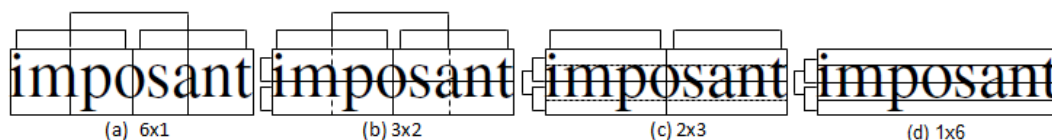


FIGURE 3.21 – Disposition de 6 cellules en chevauchement.

Pyramide d'histogramme des orientations de gradients (PHOG)

PHOG considère la propriété spatiale de la forme locale lorsque l'image est représentée par un HOG. L'information spatiale est représentée par un carrelage de l'image en cellules à plusieurs résolutions en se basant sur une mise en correspondance de pyramide spatiale [52].

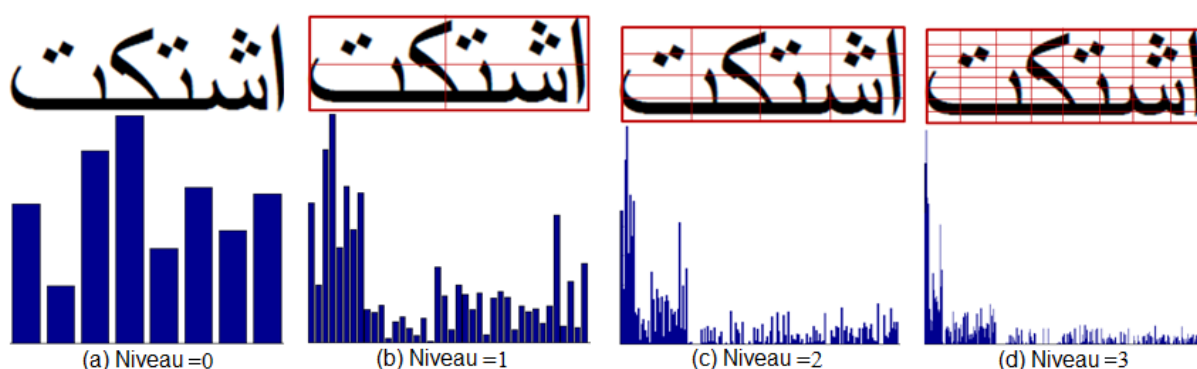


FIGURE 3.22 – Illustration de PHOG à chaque niveau de résolution.

Au départ, nous avons calculé le HOG pour la totalité de l'image (voir la Figure 3.22(a)). L'histogramme est divisé en n barres (ici $n = 8$). Chaque barre de l'histogramme représente le nombre des pixels qui ont une orientation à l'intérieur d'une certaine plage angulaire. Ensuite, l'image est divisée en quatre zones et un HOG est calculé pour chacune d'entre elles (voir la Figure 3.22(b)). Ce processus est répété récursivement jusqu'à une profondeur L (voir la Figure 3.22(c) et (d)). Par conséquent, l'image du mot est divisée en cellules à plusieurs niveaux de la pyramide. Le niveau 0 est représenté par un n -vecteur correspondant aux n barres de l'histogramme. Le niveau 1 est représenté par un $4n$ -vecteur et le descripteur PHOG de l'ensemble image est un vecteur de dimension égale à $K * \sum_{l=0}^L 4^l$. Les histogrammes du même niveau sont enchaînés dans un vecteur. Le descripteur PHOG final pour une image est une concaténation de tous les vecteurs à chaque résolution (niveau) de la pyramide. Dans notre cas, $n = 8$ et $L = 3$. Par conséquent, notre descripteur de PHOG est de longueur $8 * (1 + 4 + 16 + 64) = 680$. Il s'agit ainsi d'une analyse récursive multi-échelle, souple au niveau de l'analyse directionnelle.

La Figure 3.23 montre les descripteurs du HOG (au premier niveau) de mots arabes et latins manuscrits. Comme nous pouvons le constater, des images de mots arabes ont des HOGs similaires. De même pour les mots latins. En revanche, les images de mots latins ont des HOGs différents de ceux de mots arabes. Chaque barre est d'une largeur $\frac{\pi}{4}$ radians. Notons que HOG affiche l'orientation prédominante des traits de lettres. Par exemple, l'histogramme de mots manuscrits arabes ont des pointes importantes sur le secteur $[\frac{\pi}{2}, \frac{3\pi}{4}]$ ce qui respecte l'orientation de l'écriture arabe : de droite à gauche où les lettres sont généralement inclinées à gauche. Au contraire, les histogrammes de mots manuscrits latins ont des pointes importantes au secteur $[\pi, \frac{5\pi}{4}]$ qui respecte également l'orientation de l'écriture latine : de gauche à droite où les lettres sont plutôt inclinées à droite. De cette même figure, nous avons constaté que HOG est capable de discriminer entre les scripts, mais pas entre leurs natures. Il a fallu raffiner encore l'analyse en faisant recours au PHOG.

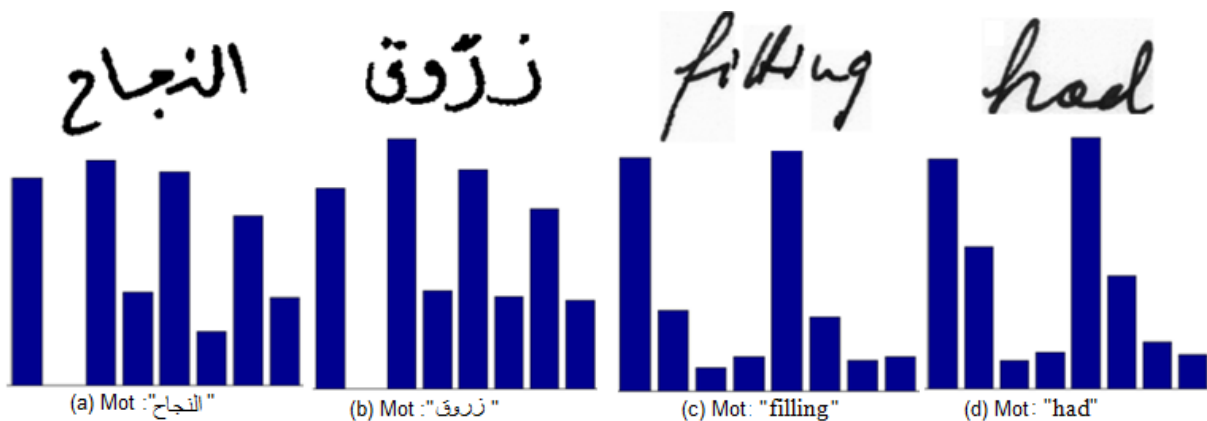


FIGURE 3.23 – Descripteurs HOG à 8 barres d'images de mots manuscrits.

Notons que le caractère multi-résolutionnel de ce descripteur est capable de délivrer des informations précises, à la fois en espace et en fréquence sur l'image, et d'analyser l'écriture à différents échelles. Notons que la magnitude ou l'intensité et l'orientation des gradients sont calculées jusqu'ici qu'au seul niveau de pixel et les relations entre les paires de pixels ne sont pas prises en compte. Nous pensons que si certaines relations entre les pixels sont considérées, les résultats seront meilleurs. Cette information peut être extraite à partir d'une matrice de co-occurrence comme indiqué ci-dessous.

Matrice de co-occurrences des orientations de gradients (CO-MOG)

Nous proposons d'utiliser CO-MOG pour capturer l'information spatiale en comptant la fréquence de co-occurrence de gradients orientés entre paires de pixels. Ainsi des

3.4 APPROCHE PROPOSÉE

emplacements relatifs sont mémorisés. Les emplacements relatifs sont reflétés par la distance entre deux pixels comme indiqué dans la Figure 3.24(a). Nous considérons le pixel jaune au centre. Les pixels bleus voisins sont des pixels à différentes distances. Chaque pixel bleu voisin forme une paire d'orientation avec le pixel jaune du centre et vote à la matrice de co-occurrence comme illustré sur la Figure 3.24(a). La fréquence de la co-occurrence du gradient orienté est capturée à chaque distance via une matrice de co-occurrence comme illustré dans la Figure 3.24(b). Notons que HOG est un cas particulier de CO-HOG lorsque la distance est nulle puisque seul le pixel en question est considéré. Par rapport à HOG, CO-HOG capture plus l'information spatiale locale tout en maintenant les avantages du HOG.

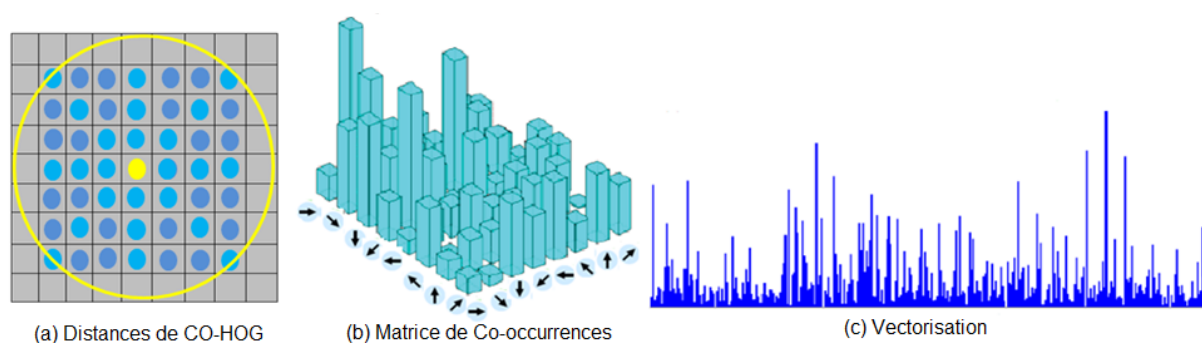


FIGURE 3.24 – (a) la distance qui peut être utilisée dans CO-MOG, (b) co-occurrence d'images de mots à une distance donnée, (c) vectorisation et concaténation des matrice de co-occurrences pour former le descripteur CO-MOG [53].

Plusieurs tests ont été effectués pour déterminer quelle distance et quelles orientations sont les plus appropriées. Notons qu'en écriture arabe ou latine (en particulier de nature imprimée), il y a beaucoup de lettres qui sont principalement composées de traits verticaux (Exemple : les lettres arabes : ل , et ل et les lettres latines : 'l', 't', 'I', etc.). En Arabe imprimé, les lettres sont horizontalement ligaturées et les ligatures peuvent être plus ou moins longues. Ainsi, pour ce type de lettres et ligatures horizontales, de nombreux pixels voisins ont une même orientation. Dans ces cas, une distance "lointaine" pourrait suffire puisqu'il n'y a pas de variations morphologiques. Par contre, dans le cas d'une écriture cursive (arabe ou latine manuscrite), où les lettres ne sont pas nécessairement rectilignes (Exemple : 'r', 'e', 'c', 'o', etc.), l'utilisation d'une distance "proche" décrira mieux les formes de lettres (jambages, boucles, angles, etc.). Nous avons trouvé qu'une distance égale à cinq est satisfaisante et qu'il suffira aussi de considérer que le voisinage haut de droite à gauche des pixels, soit quatre angles comme le montre la Figure 3.25. Ainsi, de l'image d'un mot, quatre matrices de co-occurrence sont calculées et par conséquent, la taille du vecteur des caractéristiques est $64 * 4 = 256$.

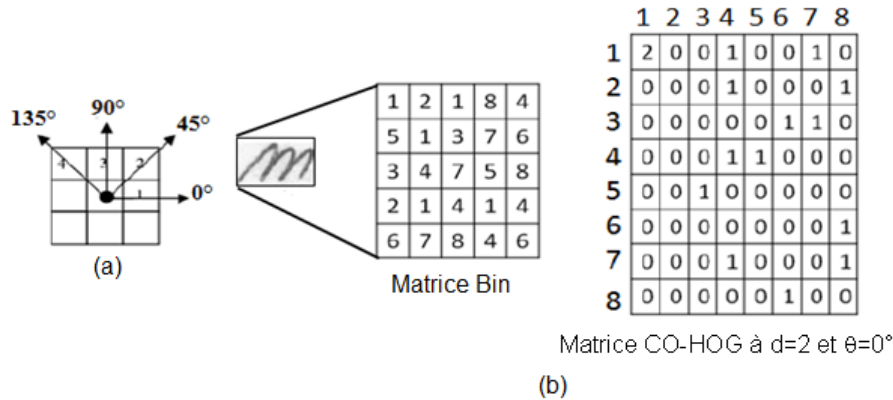


FIGURE 3.25 – (a) Du pixel central ; pixel 1 représente 0° à $d = 1$; pixel 2 représente 45° ; pixel 3 représente 90° et pixel 4 représente 135° à $d = 1$; (b) Matrice CO-MOG calculée à $d = 2, \theta = 0^\circ$ pour un voisinage 8 à 8 autour du pixel en question.

3.4.4 Expérimentations

Nous avons étudié les effets des différents choix d'implémentation, concernant les dispositions de cellules, le nombre de descripteurs et secteurs d'orientation et les distances CO-HOG, sur les performances du système d'identification. Nous avons utilisé une base de 24000 échantillons (6000 par classe). Comme le nombre des descripteurs à base de HOG est relativement élevé, nous avons eu recours à l'algorithme génétique pour sélectionner les descripteurs pertinents. Nous avons comparé aussi les performances de différents classifieurs : AODEsr, k -Plus Proches Voisins (k -PPV) et Support Vector Machine (SVM).

De ces expérimentations, nous avons constaté que pour le même nombre de descripteurs ($n = n_x * n_y$ avec $n = 8$ barres ou secteurs), le meilleur arrangement de cellules est toujours lorsque n_x ou n_y sont différents de 1. Ainsi, il faut éviter un arrangement horizontal ou vertical de cellules. Nous avons noté que les arrangements sans chevauchement de cellules sont meilleurs que ceux avec chevauchement. Nous avons remarqué aussi que plus le nombre de cellules augmente, plus le taux d'identification est meilleur. Notons que CO-MOG a abouti à un taux d'identification de 99.85%, ce qui est nettement supérieur aux taux obtenus par HOG basique, HOG avec les arrangements multiples de cellules, selon la bande centrale, avec ou sans chevauchement et PHOG (voir la Table 3.7). Il est clair aussi qu'une meilleure classification est faite par k -PPV.

Nous avons également testé l'effet du nombre de secteurs d'orientation sur le taux d'identification, en utilisant PHOG et AODEsr, et constaté que le meilleur nombre de secteurs est égale à 8 (voir la Table 3.8). Nous avons testé aussi avec différentes distances (2, 4, 6, 8, 10 et 12) au niveau de CO-HOG pour décider de la meilleure distance qui considère tout changement morphologique de lettres. Des expérimentations ont montré

3.4 APPROCHE PROPOSÉE

TABLE 3.7 – Résultats par descripteur et classifieur.

	HOG	PHOG	CO-MOG
Nombre de descripteurs	64	680	324
Descripteurs sélectionnés	38	268	248
Taux d'identification par SVM	80.56%	95.85%	96.43%
Taux d'identification par k -PPV	82.87%	95.64%	99.85%
Taux d'identification par AODEsr	93.47%	97.47%	99.71%

TABLE 3.8 – Nombre de bins d'orientation vs. Taux d'identification.

Nombre de secteurs d'orientation	5	7	8	10	12	18
Nombre de descripteurs	340	425	680	850	1020	1530
Descripteurs sélectionnés	120	201	268	374	654	847
Taux d'identification	94.54%	96.54%	97.47%	97.34%	97.38%	97.40%

qu'une distance $d = 5$ est la meilleure.

De la matrice de confusion, issue du CO-HOG et de k -PPV (voir la Table 3.9), nous avons recensé quelques cas de confusion entre mots latins et arabes manuscrits due à leur aspect cursif. Dans la Figure 3.26(a), un mot latin manuscrit a été confus à un mot latin imprimé parce les hampes et les jambages des lettres sont droits. Dans la Figure 3.26(b), un mot latin manuscrit a été classé comme mot arabe manuscrit car dans certains styles d'écritures, les lettres sont inclinées vers la gauche, ce qui est généralement caractéristique de l'écriture arabe manuscrite. Dans la Figure 3.26(c), un mot arabe imprimé a été confus à un mot latin manuscrit en raison des jambages orientés vers la droite.

TABLE 3.9 – Matrice de confusion.

	AI	AM	LI	LM	Taux d'identification
AI	5982	0	18	0	99.70%
AM	0	6000	0	0	100%
LI	16	0	5983	1	99.71%
LM	1	0	0	5999	99.98%
				Moy.	99.85%

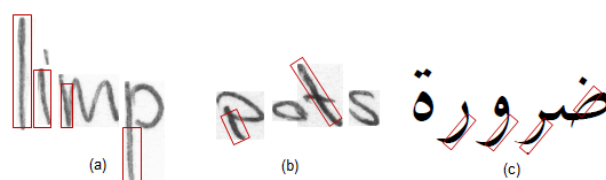


FIGURE 3.26 – Quelques cas de confusion observés.

Nous avons finalement comparé notre système, à base de CO-HOG, au système de

Benjelil et al. [39] qui traite de la même problématique mais qui utilise la transformée en pyramides orientables. Nous avons implémenté cette transformée [54] et l’a testé sur une base de mots commune. La Table 3.10 résume les résultats obtenus.

TABLE 3.10 – Notre système versus quelques travaux similaires

Système	Vecteur de caractéristiques	Classifieur	Base	Taux d’identification
[39]	48	<i>k</i> -PPV	800	97.5%
[54]	48	<i>k</i> -PPV	24000	98.94%
[55]	64 par fenêtre	GMMs	20000	99.10%
Notre système	226	<i>k</i> -PPV	20000	99.76%
Notre système	248	<i>k</i>-PPV	24000	99.85%

Notons qu’en testant avec la transformée en pyramides orientables [54], comme fait par [39], mais sur une plus grande base (de 800 à 24000 mots), le taux d’identification a été augmenté de 97.5% à 98.94%. L’utilisation des descripteurs CO-MOG avec le classifieur *k*-PPV réalise un taux d’identification de 99.85%, ce qui est nettement le meilleur parmi les taux affichés par les autres systèmes.

3.5 Conclusion

Dans ce travail, nous nous sommes intéressés à l’identification du script et de sa nature au niveau de mot. Des expériences ont été menées sur des mots manuscrits à styles d’écriture variés et des mots imprimés avec diverses polices, tailles et fontes, écrits en Arabe et en Latin dans le but de tester, proposer et sélectionner les meilleurs descripteurs et classifieurs. Nous avons constaté qu’une représentation locale du mot permet d’avoir une vue plus approfondie de la façon dont ses primitives sont formées. L’avantage d’utiliser une méthode locale est d’accorder de l’énergie aux petits détails locaux d’une image évitant ainsi le bruit engendré par un traitement global. Pour cela nous avons opté pour des descripteurs locaux en se basant sur des primitives directionnelles qui donnent la direction générale du tracé, longueurs et angles surtout que l’arabe se distingue du latin par l’orientation de l’écriture.

3.5 CONCLUSION

CHAPITRE 4

Contribution en segmentation d'anciens manuscrits arabes

4.1 Introduction

Une bonne partie du patrimoine culturel et scientifique de la Tunisie est conservée sous forme d'anciens manuscrits dans les archives nationales de la Tunisie. Le traitement automatique de ces documents en vue de leur restauration, indexation et exploitation offre un avantage certain. Cependant, on est confronté à de nombreuses difficultés dues au mauvais état de conservation de ces manuscrits et à la complexité de leur contenu ce qui mettent en échec les algorithmes classiques de segmentation. Le problème étudié, dans ce travail, concerne la segmentation en lignes et en mots d'anciens manuscrits arabes.

En effet, la segmentation de ces manuscrits est nécessaire et constitue un défi important car si elle échoue, handicape les étapes d'extraction et de reconnaissance de mots. La segmentation s'avère être plus complexe lorsqu'il s'agit de manuscrits arabes où les espacements interlignes sont parfois très réduits ce qui génère des chevauchements entre les mots des lignes successives ou bien entre deux mots de la même ligne voir même entre deux lettres du même mot. La segmentation des zones de connection constitue le principal défi pour l'extraction correcte des lignes et des mots.

La méthode de segmentation, proposée dans ce travail et faisant l'objet de la thèse de Nabil Aouadi, est destinée à des documents comme ceux de la Figure 4.1 où nous montrons juste un extrait. Il s'agit d'anciens manuscrits arabes mono-orientés. Les lignes textuelles sont horizontales et elles sont écrites selon des lignes de support. Ces manuscrits

4.2 ETAT DE L'ART

correspondent à des pages de registres de dénombrement du 19ème siècle.

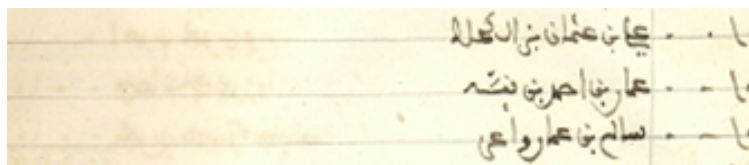


FIGURE 4.1 – Extrait d'une page de registre de dénombrement du 19ème siècle.

La segmentation en lignes et en mots des pages de ces registres, auxquels nous nous intéressons, est rendue délicate par :

- la présence de lignes successives qui peuvent se connecter par les ascendants et les descendants et qui sont très accentués dans l'écriture arabe, en vue de lui donner un attrait calligraphique,
- la présence de points et de signes diacritiques,
- la diversité des formes des lettres et la variabilité de liaison entre les lettres (ligatures horizontale et/ou verticale).

L'approche de segmentation proposée débute par localisation et extraction des zones de connections. Les lignes connectées sont ensuite séparées et l'approche se termine par une phase finale d'extraction des mots dans les lignes. Ce chapitre, objet de notre dernière contribution, est organisé de la manière suivante. Une première section présente un bref état de l'art. Une deuxième section décrit le système de segmentation proposé en insistant sur le choix de descripteurs et classifieurs pour cette tâche. Une troisième section discute des résultats de segmentation des zones de connection entre lignes. Une quatrième section explique l'étape de segmentation des lignes en mots par repérage de mots de liaison. Le chapitre se termine par une conclusion et des perspectives qui montreront un prolongement possible du système proposé.

4.2 Etat de l'art

A cause de la sinuosité des lignes et de leur multi-orientation, les méthodes de segmentation traditionnelles ont échoué à segmenter les anciens manuscrits en lignes et en mots. Partant de ces constatations, certains chercheurs proposent de nouvelles approches. Parmi les modèles, utilisés dans le domaine d'extraction des lignes, nous citons : le modèle de contour actif paramétrique. Dans [56], on propose une approche pour la détection et la séparation de lignes connectées des documents manuscrits arabes multi-orientés. En raison de la multi-orientation, on a utilisé un maillage automatique de l'image qui permet de déterminer progressivement et localement les lignes. Le maillage est initialisé avec une petite fenêtre où la taille est corrigée par extension jusqu'à ce que suffisamment de lignes aient été trouvées.

Pour la segmentation des zones de connections entre les mots d'une même ligne peu de travaux ont été proposés. Selon le nombre de lettres connectées et la nature des connections, les zones de connections peuvent être classées en trois catégories : 1) simple touche (voir la Figure 4.2(a)), 2) multi-touche où la connection est entre plus de deux lettres (voir la Figure 4.2(b)) et 3) chevauchement (entre lignes, voir la Figure 4.2(c) ou entre mots, voir la Figure 4.2(d)).

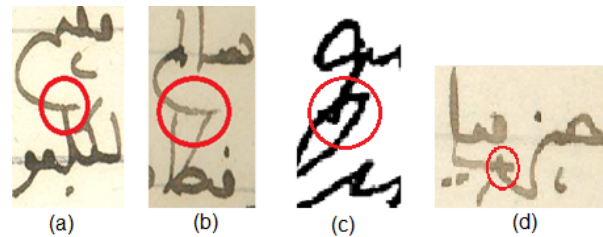


FIGURE 4.2 – (a) Simple touche, (b) Multi-touche, (c) Chevauchement, (d) Mots horizontalement interconnectés.

Les approches de segmentation des zones de connections étudiées sont résumées dans la Table 4.1. La plupart d'entre elles se font sans reconnaissance en exploitant des informations structurelles pour identifier les parties qui composent la zone de connection. Les approches qui segmentent une zone de connection par coupure horizontale [57], trouvent des difficultés pour localiser là où il faut couper. L'endroit de la coupure est choisi selon une analyse structurelle ou à base de règles. Le résultat d'un tel procédé n'est pas toujours correct car la connection entre lettres est généralement plus complexe qu'une simple touche horizontale. D'autres approches [58], [7], [59], qui se basent sur des informations de contour, s'avèrent être plus efficaces car elles tiennent compte de l'irrégularité de ces connections, mais elles se sont appliquées sur un ensemble réduit de connections. D'autres méthodes dépendent fortement des seuils qui les limitent à un ensemble réduit de styles d'écritures. Nous pensons également que les travaux à base de squelette de la zone de connection ne pourraient être efficaces que si le squelette est correctement extrait. Les approches à base de reconnaissance ont été testées sur des bases plus larges comme dans [60] et ont abouti à de bons résultats.

4.3 Système proposé

La première étape dans le système proposé est l'extraction des lignes de texte. En cas de lignes accolées, il faut passer par les étapes d'extraction et de segmentation des zones de connections. A la fin, il s'agit de segmenter les lignes en mots par repérage de mots de liaison comme expliqué plus loin.

4.3 SYSTÈME PROPOSÉ

TABLE 4.1 – Synthèse des travaux sur la segmentation.

Catégories	Réf.	Approches	Script	Résultats
Sans reconnaissance	[57]	A base de projection	Latin	-
	[61]	Contraintes géométriques	Latin	-
	[62]	Lignes de base	Latin	-
	[63]	Enveloppe convexe	Arabe	-
	[64]	Distance du centre de gravité	Arabe	-
	[58]	Méthode statistique	Latin	-
	[65]	Connaissances structurelles	Latin	-
	[66]	Algorithme à base de plages	Latin	-
	[58]	Variation angulaire	Arabe	94%
	[7]	Décomposition de forme à base du contour	Arabe	71.2%
A base de reconnaissance	[59]	Tracé du contour	Latin	82%
	[60]	Dictionnaire, contexte de forme et transformation TPS	Arabe	71.4%
	[67]	Dictionnaire et arbre de décision	Latin	90%

4.3.1 Extraction des zones de connections

D'abord, nous avons suivi les lignes de support déjà présentes dans les images des documents, comme le montre la Figure 4.3, pour extraire les lignes de base.

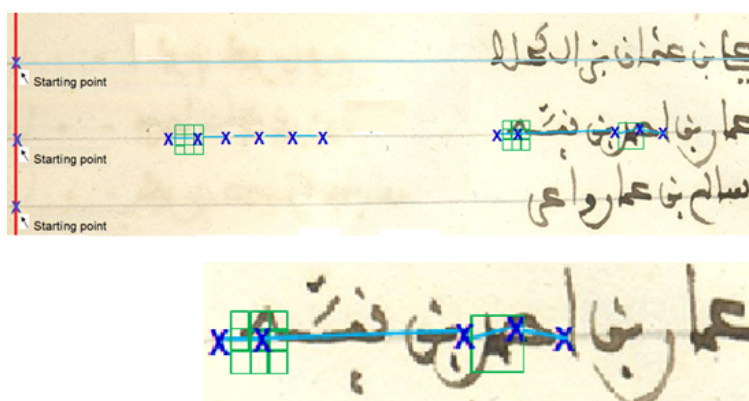


FIGURE 4.3 – Extraction des lignes de base.

Ensuite, en suivant la ligne de base, la méthode extrait les lignes de texte simplement par association des composantes connexes du même alignement. Si parmi ces composantes, il y'en a une qui fait partie simultanément à deux alignements, celle-ci est considérée cas de connection verticale. Auquel cas, ces lignes sont squelettisées et le point de jonction où les deux caractères sont connectés est identifié du côté de la ligne séparatrice situé à mi-chemin des deux lignes de base des deux lignes textuelles inter-connectées. Finalement, la connection est délimitée autour du point de jonction (voir la Figure 4.4).

Pour les connections horizontales, la méthode adoptée repose sur la morphologie de l'écriture arabe en repérant et en filtrant des points de jonction entre caractères

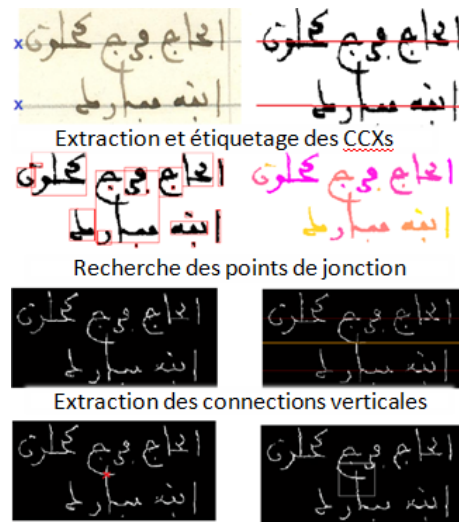


FIGURE 4.4 – Extraction des connexions verticales.

par analyse de la convexité des courbes qui leurs sont rattachées. Pour cela, il a fallu d'abord considérer la partie squeletisée en dessous de la ligne de base et trouver toutes les points de jonction. Ensuite, il a fallu trouver toutes les branches attachées à chaque point de jonction, écarter les petites branches (relatives aux signes diacritiques ou au bruit) et analyser la convexité des branches restantes. Pour les branches satisfaisant une contrainte de convexité, nous avons retenu que celles se terminant près de la ligne de base. Finalement, les branches semi-convexes sont fusionnées si nécessaire (voir la Figure 4.5).

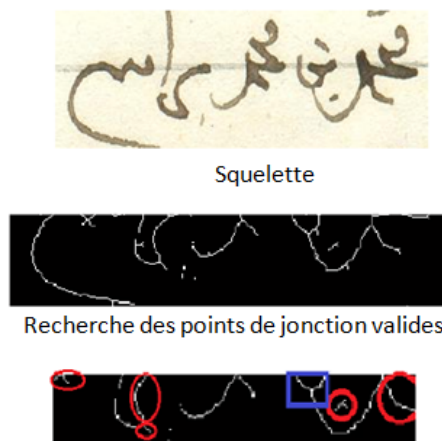


FIGURE 4.5 – Extraction des connexions horizontales.

4.3 SYSTÈME PROPOSÉ

4.3.2 Segmentation des zones de connection

L'objectif est de séparer une zone de connection en deux morceaux afin que chaque partie rejoigne son mot d'origine. Ci-après, nous discutons de la méthode que nous avons proposée pour la segmentation des zones de connections [68], [69] et [70]. Elle repose sur deux étapes majeures : 1) la reconnaissance et la transformation et 2) la segmentation proprement dite. L'étape de reconnaissance et de transformation sélectionne, à partir de plusieurs modèles prédéfinis celui qui se rapproche le plus à la zone de connection à segmenter selon une mesure de distance. A ce niveau, nous avons proposé d'ajuster le calcul de la distance, donné par [6] en trouvant un appariement optimal entre pixels de la zone de connection et ceux du modèle. Nous avons estimé par la suite une transformation de l'alignement et calculé la distance entre les deux formes comme étant une somme pondérée des erreurs d'appariement entre les points correspondants, avec le coût de la transformation et d'alignement. Nous avons montré qu'un tel ajustement dans la mesure de distance permet de réduire la complexité d'appariement et aboutir à un taux meilleur de segmentation. L'étape de segmentation se sert finalement des points intermédiaires et la répartition des composantes du modèle pour segmenter la zone de connection.

Etape de reconnaissance et de transformation

Cette étape nécessite un dictionnaire qui regroupe des classes de modèles de connections préalablement segmentées où à chaque modèle est associé deux parties issues d'une segmentation correcte de ce modèle. Ensuite, pour trouver le modèle le plus proche d'une connection à segmenter, parmi des modèles stockés dans le dictionnaire, nous avons eu recours au descripteur de contexte de forme (shape context, en Anglais). Il s'agit d'extraire d'une image des points de contour et d'obtenir pour chacun de ces points le contexte de forme en déterminant la distribution relative des points les plus proches au moyen d'un histogramme de distribution de coordonnées log-polaires (voir la Figure 4.6). C'est un histogramme de coordonnées relatives des points de contour par rapport à des points p_i , les points de référence. Ainsi, une forme est représentée par l'ensemble de points échantillonnés des contours externes et internes, $\mathcal{C} = p_1, p_2, \dots, p_n$ où $p_i \in \mathcal{R}^2$ et n est le nombre de points de contour. Pour un point p_i , les coordonnées relatives des $n - 1$ autres points sont déterminées. Les coordonnées relatives sont les coordonnées du point dans un système de coordonnées log-polaire utilisant p_i comme origine où r_q est la distance entre q et p_i et θ_q est l'angle entre le vecteur $\overrightarrow{p_i q}$ et l'axe horizontal.

$$q = (\log(r_q), \theta_q), \forall q \neq p_i \text{ et } q \in \mathcal{C} \quad (4.1)$$

Le contexte de forme h_i du point p_i est défini par un ensemble de barres où chaque barre (bin en Anglais) représente le nombre de points de contours associés au secteur de

piste correspondant (voir la Figure 4.6(b), (d) et (f)). Les cellules sombres représentent les valeurs les plus élevées et on a utilisé 5 pistes pour $\log r$ et 12 secteurs pour θ .

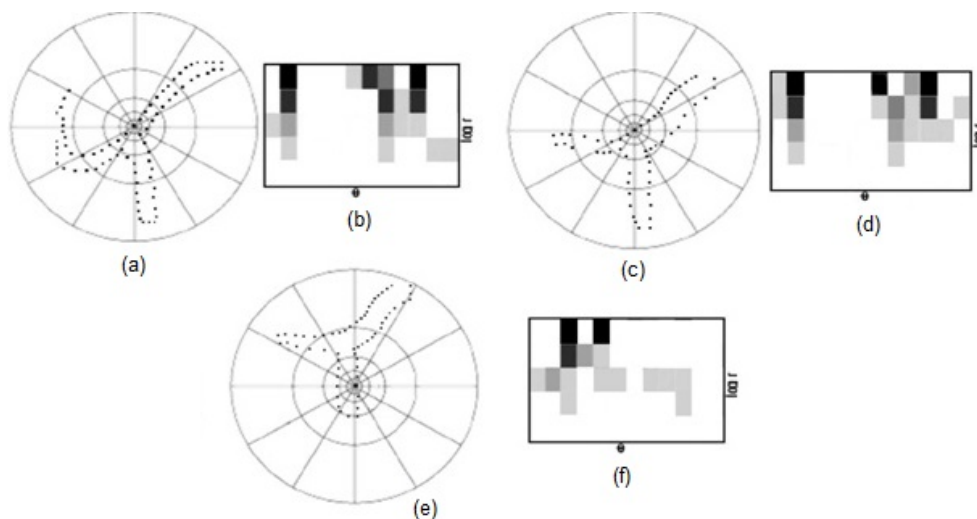


FIGURE 4.6 – (a) Contexte de forme d'un point d'une première zone de connection, (b) l'histogramme log-polaire correspondant, (c) Contexte de forme d'un point d'une seconde zone de connection, (d) l'histogramme log-polaire similaire à (b), (e) Contexte de forme d'un autre point de la seconde zone de connection, (f) l'histogramme polaire assez différent de celui de (b) et (d).

Comme le contexte de forme est un descripteur qui contient des informations de la configuration locale associée à chaque point de contour, il semble bien adapté aux zones de connections. Cependant, une telle représentation par des contextes de forme de presque tous les points de contour engendre une grande complexité dans le calcul du descripteur et lors de la recherche et l'appariement avec le modèle le plus proche. Pour cela, nous avons opté pour un appariement en bipartie par région au lieu d'un appariement point à point comme expliqué plus loin. Ci-après les phases suivies en étape de reconnaissance et transformation.

- **Apprentissage de modèles** : Notre système est entraîné à partir de classes de modèles de connections, de taille 120×120 pixels, préalablement segmentés en deux parties A et B . Ces modèles sont stockés dans un dictionnaire avec un élément représentant chaque classe qui va servir à une recherche rapide de modèles appropriés.
- **Appariement de formes** : Cette étape a pour but de trouver des correspondances entre la zone de connection et les modèles par calcul de la distance (D_{sc}). Pour appairer deux formes, notre système compare les histogrammes de contexte de forme de leurs points de contour. Pour extraire les points de contour, nous avons appliqué le filtre de Canny [71] suivi d'un processus d'échantillonnage pour obtenir

4.3 SYSTÈME PROPOSÉ

un nombre fixe de points de contour ce qui facilite l'appariement. La Figure 4.7(a) et (b) montre respectivement un exemple de zone de connection et l'échantillonnage de ses points de contour.

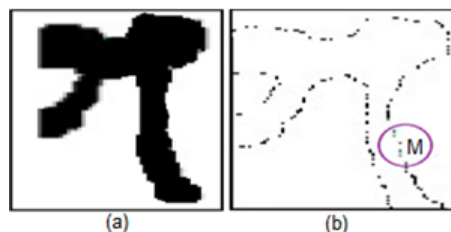


FIGURE 4.7 – (a) Zone de connection (b), Echantillonnage de ses points de contour.

Pour calculer le contexte de forme pour N points du contour et obtenir le descripteur de forme, nous avons combiné les vecteurs de tous les contextes de formes des points. Il s'agit d'une matrice $N * K$ où K est le produit de θ et les "bin" de distance (soit $5*12=60$). En utilisant deux descripteurs de formes (un pour une zone de connection d'entrée et l'autre pour un modèle stocké dans le dictionnaire), le système calcule, pour chaque couple de points (p_i, q_j) des deux formes, le coût d'appariement (voir Eq. 4.2) et obtient une matrice de coûts C de taille $N * N$ où $h_i(k)$ et $h_j(k)$ désignent les histogrammes au niveau de p_i et q_j .

$$C_{ij} = C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^K \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)} \quad (4.2)$$

L'objectif à ce niveau est de trouver l'ensemble des couples (p, q) qui minimise le coût total.

- **Points atypiques et nouvelle forme d'appariement en bi-partie :** Notons que même si le contexte de forme est un descripteur riche, la présence de points atypiques (outliers, en Anglais) est un problème qui peut affecter le résultat d'appariement surtout lorsqu'il s'agit de formes fines. Pour cela, nous avons défini une relation de voisinage entre les points de la même forme. Ainsi, pour chaque point M , nous avons considéré ses voisins lors de son appariement (voir la Figure 4.7(b)) et ce dans l'objectif d'écarter les points ne pouvant pas avoir de bonnes correspondances. Cela diffère de l'appariement point à point, proposé par Belongie et al. [6] qui tient compte de toutes les paires de points minimisant le coût. En effet, nous ne considérons que les points bien appariés (WMP, Well Matched Pixels, en Anglais), compte tenu de leur voisinage. Ainsi, pour chaque point, nous associons un sous ensemble de pixels définissant un disque de rayon r autour de lui comme le montre la Figure 4.7(b). Un appariement entre deux points n'est considéré valide que s'il existe un nombre suffisant de leurs voisins qui peuvent aussi s'apparier ensemble.

- **Estimation de la transformation d'alignement :** L'estimation de la transformation plane entre deux formes selon les résultats d'appariements est nécessaire pour deux raisons : 1) calculer la distance et 2) transformer les parties séparées du modèle de manière à s'aligner à la zone de connection et avoir ainsi des caractéristiques géométriques proches. Il existe plusieurs techniques d'interpolation. Nous avons utilisé la transformation TPS (Thin-Plate Spline, en Anglais) [72] qui est couramment utilisée pour des transformations souples de coordonnées. Cette interpolation TPS minimise l'énergie de flexion (voir la Figure 4.8).

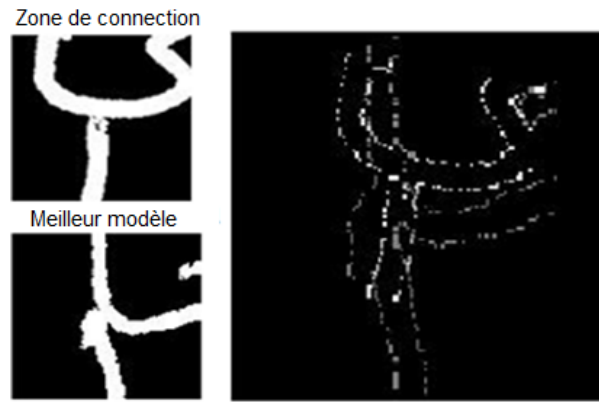


FIGURE 4.8 – Interpolation de la zone de connection en entrée et le meilleur modèle après déformation par la transformation TPS pour calculer D_{be} .

- **Calcul de distance :** Pour mesurer la distance entre deux formes, nous avons impliqué trois paramètres : 1) D_{sc} , la distance de contexte de forme, 2) N_{match} , le ratio du nombre de points bien appariés (WMP) par rapport au nombre total des points du contour N et 3) D_{be} , l'énergie de flexion nécessaire pour l'alignement du modèle à la zone de connection. D_{sc} et D_{be} ont été déjà utilisées par [6] (voir Eq. 4.3).

$$D = D_{sc} + 0.3D_{be} \quad (4.3)$$

Nous avons ajusté cette équation pour ne considérer que les points bien appariés en impliquant le ratio de points bien appariés N_{match} pondéré par un coefficient, déterminé d'une manière empirique. Ainsi, la mesure de distance que nous proposons est définie dans Eq. 4.4 comme suit :

$$D = D_{sc} + 0.3D_{be} - \text{coeff} * N_{match} \quad (4.4)$$

La Figure 4.9 montre un exemple où la formule (Eq. 4.3) adoptée par [6] ne reconnaît pas le meilleur modèle alors qu'en utilisant l'Eq. 4.4, tel que nous proposons, notre système réussit à l'identifier.

4.3 SYSTÈME PROPOSÉ

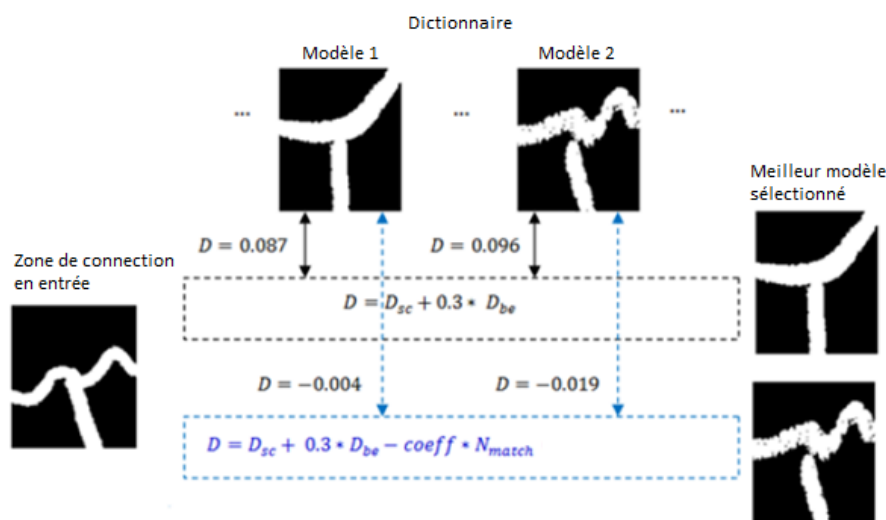


FIGURE 4.9 – Comparaison des résultats lors de la sélection du meilleur modèle.

Etape de segmentation

Maintenant que nous disposons du modèle le plus proche à la connection à segmenter ainsi que les paramètres TPS estimés qui ont servi à aligner le modèle à la connection, il est possible de se servir des parties A et B du modèle pour segmenter la connection. Soient (P_a, P_b) les contours des parties séparées du modèle, après leurs transformations TPS et (C_a, C_b) leurs points intermédiaires respectifs, définis pour caractériser la distribution d'un ensemble de points (voir l'exemple de la Figure 4.10). Rappelons que pour un ensemble de points P auquel nous associons une métrique D (distance Euclidienne ou Manhattan, etc.), le point intermédiaire M est le point le plus accessible, celui qui minimise la somme de toutes les distances de l'ensemble.

Ces points intermédiaires sont à peu près ceux des parties qui composent la zone de connection à segmenter. Pour délimiter ces parties au niveau de la zone de connection, le système associe ses pixels au point intermédiaire le plus proche. Pour les pixels de la zone commune, c'est à dire à égale distance des deux points intermédiaires, l'affectation des pixels aux deux parties de la zone de connection se fait après ajustement des points intermédiaires. Pour cela, notre système détermine d'abord les points d'intersection (I_a, I_b) entre le segment L , reliant les deux points intermédiaires initiaux, et (P_a, P_b) . Le système ajuste les points intermédiaires initiaux en trouvant leurs symétriques les plus proches des points d'intersection (voir C_b et C'_b dans la Figure 4.10(a)) et ainsi l'affectation des pixels de la zone commune (en vert) se fait selon les points intermédiaires ajustés (voir la Figure 4.11(b)). L'avantage de cette méthode de segmentation est son indépendance du script puisque nous n'avons pas besoin d'étudier les connections d'une langue spécifique. Mais, il est nécessaire d'utiliser un ensemble approprié d'apprentissage. De plus, nous

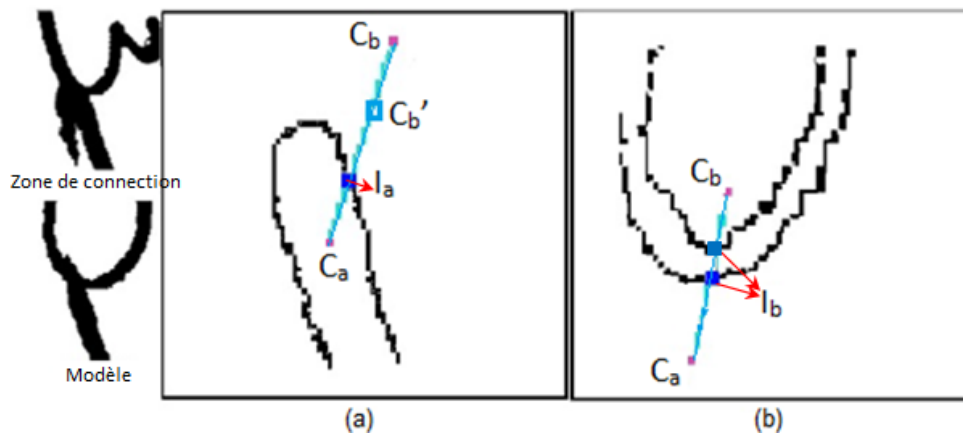


FIGURE 4.10 – (a) Contour de la partie A après transformation *TPS*, (b) Contour de la partie B après transformation *TPS*.

n'avons pas besoin de trouver le modèle qui s'aligne parfaitement à la zone de connexion en question pour pouvoir la segmenter correctement. Il suffit de trouver un modèle proche et le transformer de manière à s'approcher plus de la zone de connexion.

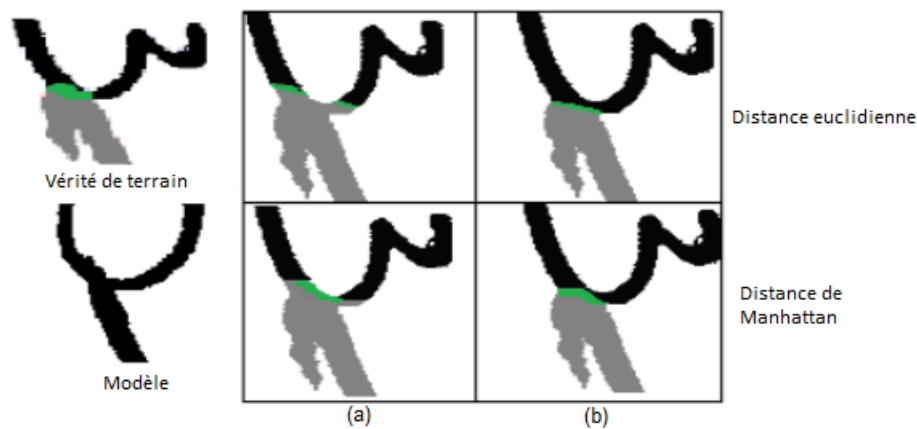


FIGURE 4.11 – Résultats de la segmentation en utilisant deux métriques différentes : (a) sans ajustement de points intermédiaires, (b) avec ajustement des points intermédiaires ajustés.

Experimentations

Pour évaluer l'extraction des zones de connexion, nous avons eu recours à 1500 lignes textuelles impliquant 500 et 600 connexions verticales et horizontales. Nous avons réussi à extraire correctement 94% des connexions entre lignes successives et 94.2% des connexions entre mots de même lignes. Quant à l'évaluation de la segmentation des zones

4.3 SYSTÈME PROPOSÉ

de connection, nous avons utilisé 3000 connections dont 670 connections horizontales et 2330 connections verticales. Certaines connections sont extraites manuellement d'images d'anciens manuscrits arabes. D'autres sont choisies parmi celles de la base de données proposée par Kang and Doermann [60]. Les valeurs des paramètres utilisés sont : $N = 200$ le nombre de points de contour échantillonnés, $Neighbors = 30$ le nombre de de voisins de chaque point, $Threshold_Neighbors = 10$ le nombre minimum de voisins bien appariés et $Coeff = 0.2$ le coefficient utilisé lors du calcul de distance, déterminé empiriquement.

Pour évaluer les résultats de la segmentation, nous avons calculé un score d'appariement (MatchScore, en Anglais) pour chaque partie [73] en se basant sur la comparaison entre les parties de la zone de connection retournées par le système et ceux de la vérité de terrain. Soit (G_A, G_B) les ensembles de tous les points des parties de la vérité de terrain, (R_A, R_B) l'ensemble des points des parties retournées par le système et $T(s)$ une fonction qui compte les éléments de l'ensemble s . MS_i représente les résultats d'appariement entre les parties de la vérité de terrain et celles retournées par le système.

$$\text{MatchScore} = \frac{2 * MS_A * MS_B}{MS_A + MS_B} \text{ avec } MS_i = \frac{T(G_i \cap R_i)}{T(G_i \cup R_i)} \quad (4.5)$$

La méthode de segmentation proposée est testée avec la distance Euclidienne, de Manhattan et Canberra. Les résultats affichés dans la Table 4.2 sont les taux de segmentation pour un score d'appariement supérieur à 0.9. Nous estimons que les résultats obtenus sont encourageants. Par la distance Euclidienne, la segmentation de la zone de connection est similaire à la coupe selon l'orthogonale de la ligne reliant les centres de gravité des parties du modèle. La distance de Manhattan semble ainsi plus appropriée pour la segmentation des zones de connection (voir la Figure 4.12).

TABLE 4.2 – Résultats des tests.

Distance	Euclidienne	Manhattan	Canberra
Taux de segmentation	92.6%	94%	92%

Pour nous comparer à des travaux connexes, nous avons mis en oeuvre la méthode de segmentation proposée par Ouwayed et Belaïd [74]. Notons que cette dernière atteint un taux de segmentation de 76% et que les erreurs enregistrées proviennent le plus souvent des zones de connection entre mots de la même ligne. A noter également que la présence de boucles dans l'image rend difficile la recherche du point de jonction principal même dans le cas de connections entre lignes successives. En se comparant aussi au travail de Kang et Doermann [60], dont le taux de segmentation est 71.4%, notre système réussit à segmenter correctement les zones de connections, qu'elles soient entre lignes ou entre mots dans 94% des cas.

Il importe de noter que notre méthode de segmentation n'est pas fortement dépendante de l'étape de la reconnaissance puisqu'elle réussit à bien segmenter la zone de

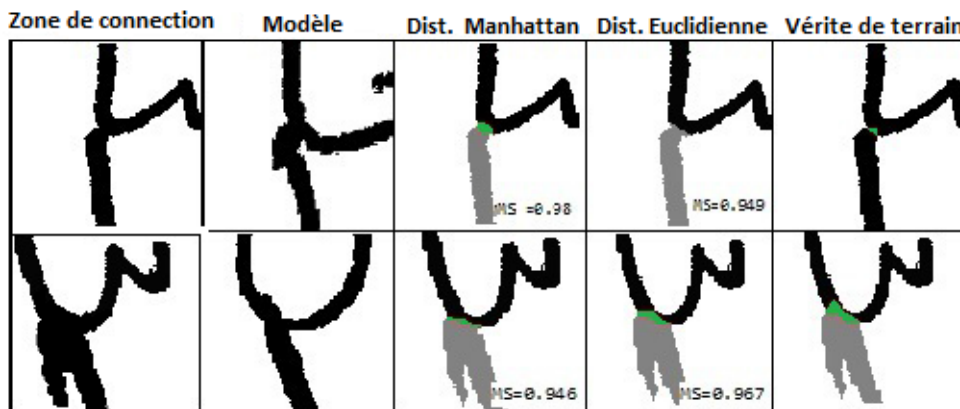


FIGURE 4.12 – Exemples de segmentation pour un Matchscore supérieur ou égal à 0.9.

connection même si ce n'est pas le meilleur modèle qui a été sélectionné car notre méthode s'appuie aussi sur la phase de transformation et l'ajustement des points intermédiaires lors de l'étape de segmentation. Les quelques erreurs de segmentation sont essentiellement dues à la déviation de la transformation. En outre, le descripteur de contexte de forme s'est avéré robuste aux déformations, bruit et aux points atypiques.

4.3.3 Segmentation des lignes en mots

Pour segmenter les lignes en mots au niveau des pages du registre de dénombrement, nous avons eu recours à la Transformée Généralisée de Hough (TGH) pour le repérage de mots de liaison comme : *بن, ابنه, اخوه, الحاج, الشيخ, العدل, etc.* (voir la Figure 4.17). Ces mots de liaison sont utilisés pour séparer entre noms et prénoms des personnes.

Rappelons que la transformée standard de Hough est une technique classique de détection de formes paramétriques telles que les droites, les cercles, les ellipses. Cette transformée a été généralisée pour la reconnaissance de formes quelconques. La méthode de segmentation en mots proposée se base sur le repérage de certains mots, que nous pouvons considérer comme formes sans représentation analytique simple par la TGH. Cette technique repose sur une similarité de formes, évaluée à partir d'une analyse très fine de l'orientation du tracé de l'écriture. Rappelons que dans la TGH, on recherche la présence d'une courbe, caractérisée par un certain nombre de paramètres. Chaque point de l'image analysée "vote" pour l'ensemble des jeux de paramètres générant des courbes auxquelles il appartient. La présence d'une courbe recherchée est caractérisée par un jeu de paramètres ayant un score élevé. Un tableau de référence définit la correspondance entre l'espace image de la forme recherchée et l'espace paramétrique (espace de Hough). La TGH agit sur des points caractéristiques de l'image, ici sur les points de contour. Elle permet, comme expliqué ci-dessous, en phase de modélisation ou d'apprentissage

4.3 SYSTÈME PROPOSÉ

de décrire chaque forme par le tableau de référence. En phase de reconnaissance, elle exploite les tableaux de références, issus de la phase d'apprentissage, pour générer les espaces de votes qui permettent de faire la classification.

- **Modélisation par la TGH** : La Figure 4.13 illustre les paramètres de la TGH pour une forme type donnée où O est le centre de gravité, MT est la tangente au contour en M , G est la normale à MT (direction du gradient), $d = MO$, α est l'angle sous lequel est vu le centre de gravité et β est l'angle de la normale au contour avec l'horizontale. Le modèle est la liste des triplets β, α, d . Plusieurs points

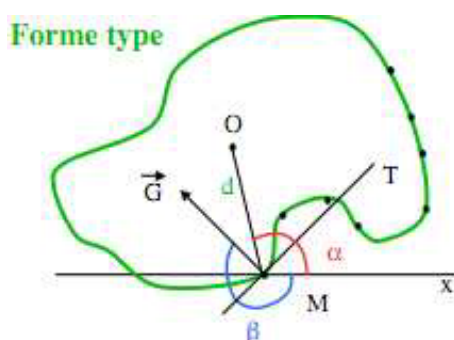


FIGURE 4.13 – Paramètres de la TGH.

du contour-modèle peuvent avoir le même angle β , avec des valeurs différentes de α_i et d_i comme le montre la Figure 4.14. Pour représenter le modèle, on regroupera les triplets β, α, d ayant la même valeur β .

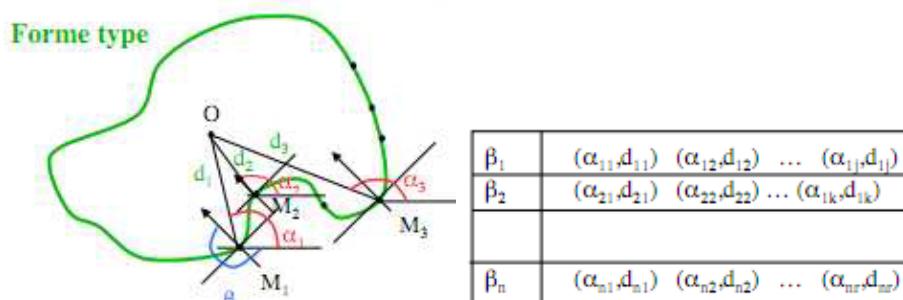


FIGURE 4.14 – Représentation du modèle.

- **Reconnaissance** : Pour reconnaître une forme, on calcule, pour chaque point P , l'angle b que fait la normale au contour en P avec l'horizontale. On estime la direction du contour à l'aide des deux points adjacents à P sur le contour. On parcourt la forme modèle et on cherche les triplets b, α, d . Ce sont les points de contours du modèle qui ont une normale parallèle à celle de P . On peut donc dire que la portion de contour centrée sur P vote pour un centre de gravité dont les

coordonnées sont données par α et d , sachant que l'espace des paramètres est celui des coordonnées du centre de gravité O (voir Eq. 4.6) et que si beaucoup de points de la forme votent pour ce même point, la forme sera reconnue.

$$x_0 = x + d \cos \alpha \text{ et } y_0 = y + d \sin \alpha \quad (4.6)$$

- **Repérage de mots par la TGH :** Nous avons choisi d'appliquer la THG sur des portions de mots pour deux raisons : 1) c'est plus simple et plus rapide que sur un mot entier et 2) beaucoup de mots ont des portions en commun (voir la Figure 4.15). Pour modéliser les portions de mots, lors de la phase d'apprentissage, nous avons retenu pour chacun d'eux son centre de gravité O comme point de référence. Ensuite, nous avons calculé, pour quelques points contour du modèle, les triplets β, α, d . Nous avons regroupé les triplets du modèle ayant la même valeur β et nous les avons rangé dans un tableau de références, dont les lignes sont indicées par les différentes valeurs de β . Nous avons construit ainsi un tableau de références correspondant à chaque modèle de portion de mot. Nous avons procédé ensuite à la construction d'un dictionnaire. Ce dernier comprend pour chaque mot, ses portions et pour chaque portion son tableau de références ainsi que des valeurs servant au regroupement des portions de mots en mots comme le seuil de Hough s -*hough* et la distance d -*cluster*.

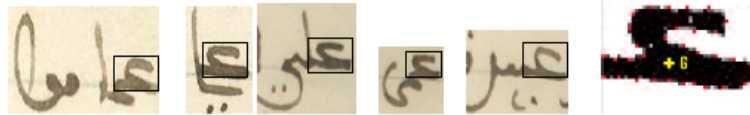


FIGURE 4.15 – Même portion appartenant à différents noms propres.

Pour repérer un mot, étant donné ses PAWs, nous avons extrait des points caractéristiques de chacun d'eux. Nous avons calculé les angles β_i correspondants. Se servant des β_i comme index dans les tableaux de références, nous avons estimé la position du point de référence pour chaque modèle du PAW inclus dans le dictionnaire. Nous avons entassé au fût et à mesure les résultats obtenus dans les accumulateurs des modèles. Un processus de recherche de maximum parmi l'ensemble complet des accumulateurs nous a permis de retenir la meilleure proposition. Si le nombre de points votants dépasse un seuil s -*hough* alors le PAW en question sera localisé. Avec la TGH il est possible ainsi d'identifier les portions relatifs aux mots et leurs positions dans l'image par localisation de leurs points de référence. Pour reconnaître les mots en entier, nous avons considéré les groupes de points votants des PAWs et les distances qui les séparent. Si une distance entre groupes de votes, relatifs aux PAWs, est inférieure à une distance d -*cluster*, alors les PAWs correspondants seront regroupés pour reconstituer le mot (voir La Figure 4.16). La

4.4 CONCLUSION

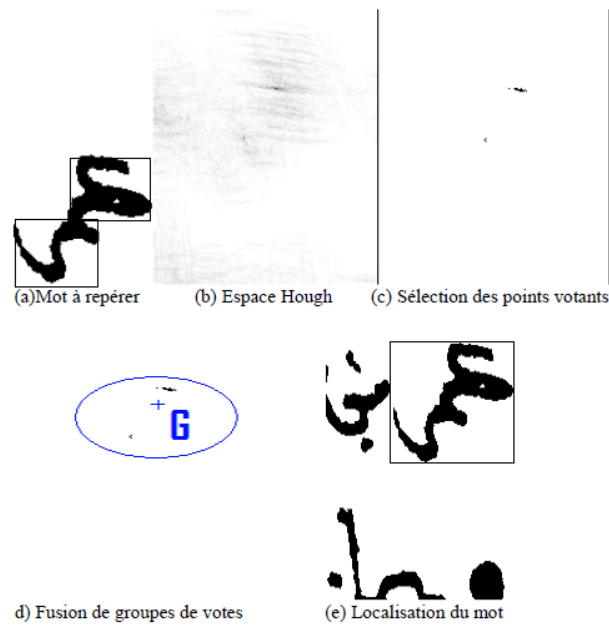


FIGURE 4.16 – Repérage du nom propre : عمر par la TGH.

Figure 4.17 montre en (a) le résultat de repérage du mot de liaison بن par la TGH et (b) son utilisation pour la segmentation en mots. Les résultats semblent être satisfaisants.

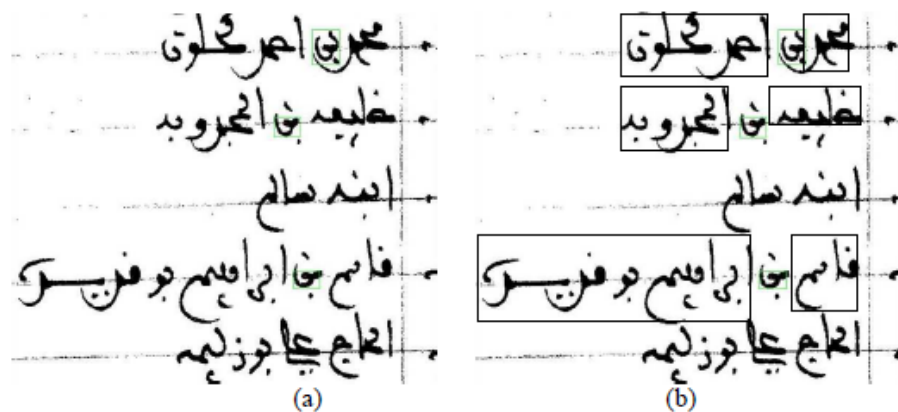


FIGURE 4.17 – (a) Repérage du mot de liaison بن, (b) Segmentation de la ligne en mots.

4.4 Conclusion

Dans ce travail, nous avons proposé des méthodes d'extraction et de segmentation des zones de connections dans des anciens manuscrits arabes. La méthode d'extraction

a l'avantage de traiter différents types de connections qu'elles soient entre lignes ou mots successifs. La méthode de segmentation se base principalement sur les étapes de reconnaissance et de transformation en s'appuyant sur le descripteur de contexte de forme là où il s'agit de trouver le meilleur modèle, stocké dans un dictionnaire, et de l'aligner à la zone de connection en question. Ensuite, les points intermédiaires des parties transformées du modèle sélectionné sont utilisés pour la segmentation proprement dite de la zone de connection. La méthode de segmentation proposée a été essentiellement testée sur des anciens manuscrits arabes. Mais, nous l'avons appliquée aussi sur quelques manuscrits latins et s'est montrée efficace. Des expériences menées sur un ensemble de zones de connections confirment l'efficacité du système de segmentation proposé. A la fin, nous avons proposé une méthode de segmentation des lignes en mots en se basant sur la TGH qui a l'avantage d'être simple, rapide et robuste au bruit.

4.4 CONCLUSION

CHAPITRE 5

Conclusion générale et perspectives

Dans cette habilitation, nous avons essayé de couvrir le domaine d'analyse et reconnaissance de l'écrit et de documents selon différents aspects de l'écriture (imprimée/manuscrite, linéaire/bidimensionnelle, arabe/latine, vocabulaire large/réduit, etc.). Pour bien traiter ces différents aspects, il a fallait choisir les bons descripteurs et classifieurs. L'objectif étant la recherche de descripteurs les plus adéquats pour avoir des classifieurs les plus puissants et résoudre des applications les plus variées qui touchent à divers aspects de l'écriture. De la synthèse des travaux encadrés, nous sommes parvenus aux constatations suivantes :

Pour la reconnaissance d'un vocabulaire large de mots arabes, les spécificités de la langue de l'écriture influent sur les choix de la classe d'approche à adopter, des descripteurs à extraire et de la méthode de reconnaissance à utiliser. De plus, les spécificités linguistiques du vocabulaire peuvent, à leur tour, influencer la décision d'intégrer ou non des connaissances morphologiques dans le processus de la reconnaissance. Pour cela, nous avons analysé les caractéristiques de l'écriture et étudié la morphologie de l'Arabe, le processus de dérivation des mots décomposables en morphèmes et leurs effets sur la structure du mot. Cette morphologie complexe fait que les mots arabes se comptent aux millions ce qui rend leur traitement automatique irréaliste. Pour cela, nous avons proposé une approche, inspirée du système neuronal cognitif de Mc Clelland et Rumelhart, permettant une représentation du mot par couches, allant du local (couche des primitives) au global (couche des mots possibles) et inversement. Nous nous sommes limités au traitement d'un vocabulaire canonique de mots décomposables vu que la plupart des mots arabes

sont dérivés de racines trilitères saines. Nous avons proposé de combiner trois classifieurs neuro-linguistiques : des réseaux de neurones transparents pour reconnaître les racines, les schèmes et les éléments de conjugaison. Le mot est ensuite reconstitué à partir des entités reconnues. Cette répartition permet de mettre en facteur plusieurs composants communs des mots et par conséquent une économie dans la représentation d'un vocabulaire large. Le réel défi de ce travail a porté sur une collaboration intelligente de ces trois classifieurs linguistiques neuronaux pour assurer l'apprentissage et la reconnaissance d'un maximum de mots. Des perspectives de ce travail représentent des champs fertiles pour de nouveaux travaux de recherches : 1) traiter un vocabulaire plus large de mots décomposables dérivés de racines saines non seulement tri-consonantiques mais aussi quadri-consonantiques, 2) reprendre l'expérimentation du système sur une base de mots arabes manuscrits en utilisant des modèles graphiques probabilistes qui sont plus adaptées aux variations de styles d'écriture et 3) étendre le système à la reconnaissance de phrases en tirant profit de l'analyse morphologique des mots.

Pour résoudre le problème de distorsions non linéaires au niveau de l'écriture arabe manuscrite par des modèles graphiques probabilistes, l'étude de plusieurs modèles de Markov cachés a montré que ces derniers savent bien s'adapter à la variabilité des observations, notamment les distorsions élastiques. Cependant ces modèles sont essentiellement mono-dimensionnels et l'écriture a une nature réellement bidimensionnelle. Des extensions à l'aspect bidimensionnel de l'écriture ont été proposées dans ce travail. Dans cette optique, des modèles probabilistes s'appuyant sur les réseaux bayésiens dynamiques ont été conçus, implémentés et testés. Notre approche est basée sur le couplage de deux modèles de type Markov caché, en faisant apparaître des liens supplémentaires entre variables. Les modèles ont été obtenus à partir des primitives structurelles, extraites de bandes verticales horizontales du mot. Cela nous a permis d'obtenir une bonne modélisation des images des mots manuscrits, due au regroupement des informations concernant les lignes et les colonnes. Les résultats expérimentaux sur la base IFN-ENIT montrent que cette voie est très prometteuse dans le domaine de la reconnaissance de l'écrit. Dans l'avenir, nous projetons d'examiner d'autres architectures à base de modèles graphiques probabilistes ou combiner avec d'autres types de modèles notamment les réseaux de neurones profonds pour une meilleure performance de reconnaissance.

Pour la reconnaissance hors-ligne de formules mathématiques arabes, les approches existantes sont empruntées le plus souvent des recherches en Latin. Des méthodes structurelles ont été proposées, mais elles sont restreintes à des formules assez simples. La plupart reposent sur l'analyse syntaxique ascendante ou descendante par respect à une grammaire. D'autres utilisent des graphes. Toutes ces techniques semblent trouver des limites du fait de la rigidité de leur action et la lenteur du processus d'analyse. Nous avons proposé un système dirigé par la syntaxe qui soit à la fois efficace et flexible, capable de gérer une grande variation dans l'emplacement des symboles et reconnaître des formules à

structure assez complexes : des structures bidimensionnelles récursives, impliquant même des opérateurs implicites. Pour la reconnaissance des symboles, nous avons extrait des descripteurs statistiques et utilisé un simple classifieur. Pour l'analyse structurale de la formule, nous avons implémenté un analyseur syntaxique ascendant-descendant qui repose sur une grammaire de coordonnées et la dominance d'opérateurs. Dans le système proposé, il a été possible d'utiliser le contexte pour aider à corriger des symboles en confusion. De plus, il retourne les résultats de la reconnaissance en MathML, ce qui facilite le traitement automatique, la recherche et la réutilisation des formules mathématiques. En perspectives, nous envisageons de traiter des formules manuscrites, bruitées et de structures plus complexes pour confirmer l'efficacité et la robustesse du système proposé. Nous projetons aussi d'élaborer des tests de performance du système sur un nombre plus grand de symboles et de formules.

Pour la discrimination entre mots arabes et latins de natures imprimée et manuscrite, la tâche est complexe à cause de la variation de fontes, tailles, styles et scripts et l'identification au niveau du mot où la quantité d'information, pourrait être insuffisante. Nous avons commencé par l'étude des descripteurs, proposées dans la littérature et utilisé quelques méthodes de sélection pour réduire leur nombre, en éliminant les descripteurs redondants et non pertinents, et améliorer la performance du classifieur. Nous avons présenté aussi une méthode alternative pour la discrimination des mots en adaptant le concept d'histogramme des orientations de gradients. La motivation derrière l'utilisation de ce type de descripteurs vient de leurs capacité à détecter l'orientation de l'écriture qui semble être une caractéristique permettant de séparer l'Arabe du Latin. Pour cela, nous nous sommes basés sur la morphologie de ces deux scripts pour localiser les orientations dominantes. Les résultats expérimentaux montrent que le processus d'identification est robuste et fiable. Notons que ces descripteurs peuvent être généralisés pour inclure autres langues romaines et anglo-saxon au lieu de considérer seulement l'Anglais ou le Français et d'autres langues qui utilisent des scripts arabes tels que le Perse et l'Urdu. Dans l'avenir, nous prévoyons d'extraire et/ou combiner avec de nouveaux descripteurs, de tester d'autres classifieurs en utilisant des bases de mots plus larges.

Pour la segmentation d'anciens manuscrits arabes, les connections entre lignes ou mots successifs compliquent considérablement la segmentation de ce type de documents. Nous avons noté aussi que dans l'écriture arabe, ces situations existent fréquemment à cause des caractères ascendants et descendants et la présence massive de symboles diacritiques qui génère souvent des fausses lignes. Pour cela, il a fallu d'abord extraire les zones de connection entre lignes ou mots successifs, puis les segmenter par une méthode qui se base sur le descripteur de contexte de forme et une fonction de déformation. Nous avons proposé également une méthode de segmentation des lignes en mots par repérage de mots de liaison grâce à la transformée généralisée de Hough. Nous avons abouti à des résultats satisfaisants. Dans le futur, nous proposons de tester le système proposé sur

des manuscrits écrits dans d'autres scripts pour confirmer son aspect générique.

Pour l'évaluation de nos systèmes, nous nous sommes attachés à utiliser des bases de données publiques pour comparer leurs performances à celles d'autres systèmes. Mais, même si les bases utilisées dans nos travaux sont standards dans le domaine, la comparaison des résultats n'est pas toujours évidente, du fait qu'elle suppose des protocoles d'évaluation identiques. Ainsi, nous projetons de participer à des campagnes d'évaluation sur des bases de données communes.

En conclusion, les méthodes d'extraction de descripteurs et de classification sont nombreuses et variées. Chacune d'elle possède des propriétés qui lui sont propres et ne peut s'appliquer qu'à certains contextes, réunissant certaines conditions. Avant de porter son choix sur une méthode plutôt qu'une autre, il est important de connaître tous les tenants et les aboutissants de leur utilisation. Pour ce faire nous avons étudié, testé et combiné différentes méthodes d'extraction et de classification. De tout ces travaux, il m'est possible d'affirmer qu'il ne s'agit pas simplement d'inventer de nouvelles méthodes, mais de faire en sorte que leurs particularités se voient à l'image.

Bibliographie

- [1] Ben Amor N. and Ben Amara Esskouri N. Analyse texturale de l'écriture arabe multifonte de gabor aux contourlets. Actes CIFED, (2008).
- [2] Belaïd A. Ben Amara N. and Ellouze N. Utilisation des modèles markoviens en reconnaissance de l'écriture arabe : Etat de l'art. Actes de CIFED, (2000).
- [3] Cheriet M. Visual recognition of arabic handwriting : challenges and new directions. In Proc. of the 2006 conference on Arabic and Chinese handwriting recognition, SACH'06, pp. 1_21, Berlin, Heidelberg, 2008. Springer_Verlag.
- [4] Kanoun S. Identification et analyse de textes arabes par approche affixale. PhD thesis, Université des Sciences et techniques de Rouen, France, 2002.
- [5] Belaïd A. Reconnaissance structurelle de caractères manuscrits et de formules mathématiques. Thèse de Doctorat en Informatique, Nancy, (1979).
- [6] Malik J. Belongie S. and Puzicha J. Shape matching and object recognition using shape context. IEEE transactions on pattern analysis and machine intelligence, pp. 509_522, (2002).
- [7] Cao H. Prasad R. Kang L., Doermann D. S. and Natarajan P. Local segmentation of touching characters using contour based shape decomposition. In Proc. of 10th IAPR International Workshop on Document Analysis Systems, Gold Coast, Queensland, Australia, pp. 460_464, (2012).
- [8] Ren J. AlKhateeb J. H., Pauplin O. and Jiang J. Performance of hidden markov model and dynamic bayesian network classifiers on handwritten arabic word recognition. Knowledge-based systems, 24(5), pp. 680_688, July (2011).
- [9] Ben Amor N. and Essoukri Ben Amara N. Analyse texturale de l'écriture arabe multifonte de gabor aux contourlets. Actes de CIFED, pp. 67_72, (2008).
- [10] Darwish K. Building a shallow arabic morphological analyzer in one day. In Proc. of the ACL_02 workshop on Computational approaches to semitic languages (SE-

BIBLIOGRAPHIE

- MITIC), pp. 1_8, Stroudsburg, PA, USA, 2002 Association for Computational Linguistics.
- [11] McClelland J. L. and Rumelhart D. E. Distributed memory and the representation of general and specific information. *Journal of Experimental Psychology : General*, pp. 159_188, (1985).
 - [12] McCulloch W. S. and Pitts W. A logical calculus of the ideas immanent in nervous activity. MIT Press, Cambridge, MA, USA, pp. 15_27, (1988).
 - [13] Aljlal M. and Frieder O. On arabic search : improving the retrieval effectiveness via a light stemming approach. In *Proc. of the International Conference on Information and Knowledge Management (CIKM)*, pp. 340_347, ACM, November 4_9, 2002.
 - [14] Alimi A. M. Kanoun S. and Lecourtier Y. Affixal approach for arabic decomposable vocabulary recognition : A validation on printed word in only one font. In *Proc. of 8th International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1025_1029, Seoul, Korea, 29 August_1 September 2005. IEEE Computer Society.
 - [15] Ben Hamadou A. Vérification et correction automatiques par analyse affixale des textes écrits en langage naturel. PhD thesis, Faculté des Sciences de Tunis, Tunisie, 1993.
 - [16] Kammoun W. and Ennaji A. Reconnaissance de textes arabes à vocabulaire ouvert. *Actes Conférence Internationale Francophone sur l'Écrit et le Document (CIFED)*, France, 06, 2004.
 - [17] Cheriet M. Côté M., Lecolinet E. and Suen C. Y. Building a perception based model for reading cursive script. In *Proc. of International Conference on Document Analysis and Recognition (ICDAR)*, pp. 898_901, 1995.
 - [18] Belaïd A. Snoussi Maddouri S., Amiri H. and Choisy C. Combination of local and global vision modelling for arabic handwritten words recognition. In *Proc. of the Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pp. 128_132, Washington, DC, USA, 2002, IEEE Computer Society.
 - [19] Essoukri Ben Amara N. Masmoudi Touj S. and Amiri H. A hybrid approach for off_line arabic handwriting recognition based on a planar hidden markov modeling. In *Proc. of the 9th International Conference on Document Analysis and Recognition*, Vol. 02, pp. 964_968, Washington, DC, USA, IEEE Computer Society, (2007).
 - [20] I. Khoury A. Giménez and A. Juan. Windowed bernoulli mixture hmms for arabic handwritten word recognition. In *International Conference on Frontiers in Handwriting Recognition, ICFHR 2010*, Kolkata, India, pp. 533_538. IEEE Computer Society, 16-18 November (2010).

- [21] S. Kanoun A. M. Alimi F. Slimane, R. Ingold and J. Hennebert. A new arabic printed text image database and evaluation protocols. In International Conference on Document Analysis and Recognition, pp. 946_950, (2009).
- [22] Likforman-Sulem L. and SigelleSrihari M. Recognition of degraded characters using dynamic bayesian networks. Pattern Recognition, 41(10), pp. 3092_3103, (2008).
- [23] AlKhateeb J. H. Offline handwritten arabic digit recognition using dynamic bayesian network. In Proc. of ICCIT, pp. 176_180, (2012).
- [24] Pauplin O. and Jiang J. A dynamic bayesian network based structural learning towards automated handwritten digit recognition. In Proc. of HAIS the 5th international conference on Hybrid Artificial Intelligence Systems, pp. 120_127, (2010).
- [25] Jayech K. Mahjoub M. A., Ghanmy N. and Miled I. Multiple models of bayesian networks applied to offline recognition of arabic handwritten city names. Computer Vision and Pattern Recognition, (2013).
- [26] Essoukri Ben Amara N. Masmoudi Touj S. and Amiri H. Arabic handwritten words recognition based on a planar hidden markov model. IAJIT, 2(4), pp. 318_325, (2005).
- [27] Lazrek A. Aspects de la problématique de la confection d’une fonte pour les mathématiques arabes. Cahiers GUTenberg no 3940, Mai (2001).
- [28] Smirnova E. and Watt S. Aspect of mathematical expression analysis in arabic handwriting. In Proc. of ICDAR, Vol. 2, pp. 1183_1187, (2007).
- [29] El-Sheikh T. S. Recognition of handwritten arabic mathematical formulas. UK IT conference, pp. 344_351, (1990).
- [30] Khalifa M. and Bing Ru Y. A hybrid segmentation system of offline arabic mathematical expression recognition. Canadian Journal on Image Processing and Computer Vision, Vol. 2, No. 4, pp. 30_35, (2011).
- [31] Blostein D. and Grbavec A. Recognition of mathematical notation. Handbook of character recognition and document image analysis, Eds. H. Bunke and P.S.P. Wang, world scientific publishing company, pp. 557_582, (1997).
- [32] Belaïd A. and Haton J. P. A syntactic approach for handwritten mathematical formula recognition. IEEE Trans. Pattern Analysis and machine intelligence, vol. 6, no. 1, pp. 105_111, (1984).
- [33] Phillips I. Wang Y. and Haralick R. Document zone content classification and its performance evaluation. Pattern Recognition, pp. 57_73, (2006).
- [34] Zheng Y. Machine printed text and handwriting identification in noisy document images. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 3, pp. 337_353, (2004).

BIBLIOGRAPHIE

- [35] Leonard E. John Cleary G. and Trigg. K^* : An instance-based learner using an entropic distance measure. In Proc. of 12th International Conference on Machine Learning, pp. 108_114, (1995).
- [36] Anderson R. H. Two-dimensional mathematical notation. In Proc. of Syntactic Pattern Recognition Applications, K.S. Fu, Ed. Springer Verlag, New York , pp. 147_177, (1977).
- [37] Ennaji A. Kanoun S., Moalla I. and Alimi A. M. Script identification for arabic and latin printed and handwritten documents. In Proc. of DAS, pp. 159_165, (2000).
- [38] Kanoun S. Baâti K. and Benjlaiel M. Différenciation d'écriture arabe et latine de natures imprimée et manuscrite par approche globale. Actes CIFED, (2010).
- [39] Mullet R. Benjelil M. and Alimi A. Language and script identification based on steerable pyramid features. In Proc. of ICFHR, Bary_Italy, pp. 712_717, 18_20 September (2012).
- [40] Elgammal A. M. and Ismail M. A. Techniques for language identification for hybrid arabic_english document images. In Proc. of DRR, pp. 1100_1104, (2001).
- [41] Mozaffari S. and Bahar P. Farsi/arabic handwritten from machine_printed words discrimination. In Proc. of ICFHR, pp. 694_699, (2012).
- [42] Kavallieratou E. and Stamatatos S. Discrimination of machine_printed from handwritten text using simple structural characteristics. In Proc. of ICPR, pp. 437_440, (2004).
- [43] Lu Y. Zhou L. and Tan C. Bangla/english script identification based on analysis of connected components profiles. In Proc. of DAS, pp. 243_254, (2006).
- [44] Alimi A. M. Ben Jlaiel M., Kanoun S. and Mullet R. Three decision levels strateg for arabic and latin texts differentiation in printed and handwritten natures. In Proc. of ICDAR, pp. 1103_1107, (2007).
- [45] Conici A. Faria da Silva L. and Sanchez A. Automatic discrimination between printed and handwritten text in documents. In Proc. of SIBGRAPI, XXII Brazilian Symposium, pp. 261_267, (2009).
- [46] Snoussi Maddouri S. Haboubi S. and Ellouze N. Discrimination between arabic and latin from bilingual documents. In Proc. of CCCA, (2011).
- [47] Saïdani A. Kacem A. and Belaïd A. A system for an automatic reading of student information sheets. In Proc. of ICDAR, pp. 1265_1269, (2011).
- [48] Liu C. Zheng Y. and Ding X. Single character type identification. In Proc. of DRR, pp. 49_56, (2002).
- [49] Ma H. and Doermann D. Word level script identification for scanned document images. In Proc. of ICDAR, (2004).

- [50] Simoncelli E. P. and Freeman W. T. The steerable pyramid : a flexible architecture for multi_scale derivative computation. In Proc. of CIP, pp. 444_447, (1995).
- [51] Dalal N. and Triggs B. D. Histograms of oriented gradients for human detection. In Proc. of Comput. Vision pattern recognition, Vol. 1, pp. 886_893, (2005).
- [52] Schmid C. Lazebnik S. and Ponce J. Beyond bags of features : Spatial pyramid matching for recognizing natural scene categories. In Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 2169_2178, (2006).
- [53] Ito S. Watanabe T. and Yokoi K. Co-occurrence histograms of oriented gradients for human detection. Information and Media Technologies, Vol. 5, no. 2, pp. 659_667, (2010).
- [54] Nguyen D. Simplified steerable pyramid. available at :www.mathworks.com/matlabcentral/fileexchange/36488-simplified-steerable-pyramid, (2012).
- [55] Kanoun S. Mezghani A., Slimane F. and Margner V. Printed/handwritten arabic script identification using local features and gmms. Actes de CIFED. (2014).
- [56] Ouwayed N. and Belaïd A. Détection et séparation de lignes connectées dans les documents multi-orientés. Actes Colloque International Francophone sur l'Écrit et le Document, CIFED, pp. 415_426, (2010).
- [57] Likforman-Sulem L. and Faure C. Une méthode de résolution des conflits d'alignements pour la segmentation des documents manuscrits. Traitement du Signal, vol. 12, No. 6, pp. 541_549,(1995).
- [58] Pratikakis I. Ouloudis G. L., Gatos B. and Halatsis C. Text-line and word segmentation of handwritten documents. Pattern Recognition, vol. 42, No. 12, pp. 3169_3183, (2009).
- [59] Pal U. Alaei A. and Nagabhushan P. A new scheme for unconstrained handwritten text-line segmentation. Pattern Recognition, vol. 44, No. 4, pp. 917_928, (2011).
- [60] Kang L. and Doermann D. Template based segmentation of touching components in handwritten text-lines. In Proc. of International Conference on Document Analysis and Recognition, Beijing China, pp. 417_422, (2011).
- [61] Vassilis K. Vassilis P., Themis S. and George C. Handwritten document image segmentation into text-lines and words. Pattern Recognition, vol. 43, No. 1, pp. 369_377, (2010).
- [62] Camillerapp J. Lemaitre A. and Couâsnon B. A perceptive method for handwritten text segmentation. In Proc. of Document recognition and retrieval XVIII - Electronic Imaging, San Francisco - United States, pp. 1_10, (2011).

BIBLIOGRAPHIE

- [63] Doermann D. S. Kumar J., Kang L. and Abd-Almageed W. Segmentation of handwritten text-lines in presence of touching components. In Proc. of International Conference on Document Analysis and Recognition, Beijining China, pp. 109_113, (2011).
- [64] Kang L. Kumar J., Abd-Almageed W. and Doermann D. S. Handwritten arabic text line segmentation using affinity propagation. In Proc. of 9th IAPR International Workshop on Document Analysis Systems, New york, pp. 135_142, (2010).
- [65] Takru K. and Leedham G. Separation of touching and overlapping words in adjacent lines of handwritten text. In Proc. of Eighth International Workshop on Frontiers in Handwriting Recognition, Ontario, Canada, pp. 496_501, (2002).
- [66] Uma Devi R. S. Rohini S. and Mohanavel S. Segmentation of touching, overlapping, skewed and short handwritten text-lines. International Journal of Computer Applications, vol. 49, No 19, pp. 24_27, (2012).
- [67] Viard-Gaudin C. Piquin P. and Barba D. Coopération des outils de segmentation et de binarisation de document. in proc. of Colloque National sur l'Ecrit et le Document, Rouen, pp. 283_292, (1994).
- [68] Kacem A. Aouadi N. and Belaïd A. Segmentation of touching component in arabic manuscripts. In Proc. of ICFHR, Crete Island, Greece, (2014).
- [69] Amiri S. Aouadi N. and Kacem A. Segmentation of connected component in arabic handwritten documents. In Proc. of CIMTA, Elsevier, Vol. 10, pp. 738_746, (2013).
- [70] Amiri S. Aouadi N. and Kacem A. Segmentation of touching and overlapping components in arabic handwritten documents. In Proc. of PSIVT, (2013).
- [71] Canny J. A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell., vol. 8, No. 6, pp. 679_698,(1986).
- [72] Bookstein F. L. Principal warps : Thin-plane spline and the decomposition of deformations. IEEE transactions on pattern analysis and machine intelligence, (1989).
- [73] Stamatopoulos N. Gatos B. and Louloudis G. Icdar handwriting segmentation contest. In Proc. of ICDAR, (2009).
- [74] Ouwayed N. and Belaïd A. Separation of overlapping and touching lines within handwritten arabic documents. In Proc. of the 13th International Conference on Computer Analysis of Images and Patterns, Münster (North Rhine-Westphalia), Germany, pp. 237_244, (2009).