

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la
recherche scientifique

Université Larbi Tébessi - Tébessa



Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie

Département : Mathématiques et Informatique

Mémoire de fin d'étude

Pour l'obtention du diplôme de **MASTER**

Domaine : Mathématiques et Informatique

Filière : Informatique

Option : **SYSTÈMES MULTIMÉDIA**

Thème

Reconnaissance Des Mots Arabes Manuscrits

Présenté Par

RIADH MEBARKIA

Devant le jury

Pr. Djeddi Chawki

Université Larbi Tébessi

Président

Dr. Daouadi Kheireddine

Université Larbi Tébessi

Examineur

Dr. ABBES Faycel

Université Larbi Tébessi

Encadreur

Date : 2022/ 2023

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Résumé

La reconnaissance des mots arabes manuscrits est un domaine de recherche vaste et complexe. Englobant divers aspects de la langue et ses subtilités. Ce domaine continue de se développer, avec un intérêt croissant des chercheurs pour comprendre les défis de la reconnaissance des mots arabes. Deux approches de reconnaissance sont proposées dans la littérature : l'approche analytique et l'approche globale.

Le système proposé est basé sur une approche de reconnaissance globale de l'écriture arabe manuscrite hors ligne dans un vocabulaire limité, des noms de villes/villages tunisiens, en l'absence de prétraitement, le système est composé de deux étapes : extraction des caractéristiques et classification, par l'utilisation des différents descripteurs (LBP, Obif, CLBP) qui permet de donner des représentations différentes des mots dont le but est d'améliorer le taux de reconnaissance des mots manuscrit Arabe basée sur le classificateur SVM.

Le système est testé sur la base du manuscrit Arabe IFN-ENIT. Elle donne des résultats encourageants et ouvre des perspectives dans le domaine de la reconnaissance du manuscrit Arabe.

Mots clés :

Reconnaissance manuscrite de l'arabe :

Classificateur SVM

Descripteurs LBP, CLBP, Obif

المخلص

يعتبر التعرف على الكلمات العربية المكتوبة بخط اليد مجال بحث واسع ومعقد. يشمل جوانب مختلفة من اللغة وتعميقاتها. يستمر هذا المجال في التطور مع زيادة اهتمام الباحثين بفهم تحديات التعرف على الكلمات العربية. تم اقتراح نهجين للتعرف: النهج التحليلي والنهج العام

يعتمد النظام المقترح على نهج الاعتراف العام بالنص العربي المكتوب بخط اليد بمفردات محدودة، أسماء البلدات / القرى التونسية، بالاستغناء عن خطوة المعالجة المسبقة، يتكون النظام من خطوتين: استخراج الميزة وتصنيفها، من خلال استخدام واصفات مختلفة (LBP, Obif, CLBP) تقدم تمثيلات مختلفة للكلمات التي تهدف إلى تحسين معدل التعرف على الكلمات المكتوبة بخط اليد العربية بناءً على مصنف SVM.

تم اختبار النظام على قاعدة بيانات العربية INF-ENIT حيث اعطي نتائج مشجعة ويفتح آفاقاً في مجال التعرف على خط اليد باللغة العربية.

الكلمات الدالة:

التعرف على كلمات اللغة العربية بخط اليد

مصنف SVM

الواصفات LBP ، CLBP ، Obif

Abstract

The recognition of handwritten Arabic words is a broad area of research. This thesis focuses on evaluating an offline handwriting recognition system specifically designed for Arabic words. The words used in this system are extracted from the IFN/ENIT database, which is a research-oriented database containing names of Tunisian cities/villages.

The system adopts a global recognition approach, where the entire word image is recognized without any segmentation of individual characters.

To develop the system, a preprocessing step is applied to the word images, which involves size normalization and the removal of diacritical marks, as well as white spaces in columns and rows.

For feature extraction, the LBP (Local Binary Patterns), CLBP (Compound Local Binary Patterns), OBIF (Oriented Basic Image Feature), and OBIF column descriptors are utilized. These descriptors are then fed into the SVM (Support Vector Machine) classifier.

The results show that the system achieves promising results, which indicates its performance compared to other handwritten Arabic word recognition systems.

Keywords:

Handwritten Arabic recognition:

SVM classifier

Descriptors LBP, CLBP, Obif

Remerciements

je remercie le bon Dieu tout puissant de m'avoir donné la force pour survivre, ainsi que l'audace pour dépasser toutes les difficultés et avoir éclairé le chemin pour la réalisation de cette thèse.

*On profite de l'occasion pour remercier toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet de fin d'études, et surtout les professeurs **Aouine Mohammed** et **Abdeljalil Gattel**.*

*On tient à exprimer nos vifs remerciements pour notre grand et respectueux professeur **Abess faycel**, d'avoir accepté de nous encadrer pour notre projet de fin d'études, ainsi que pour son soutien, ses remarques pertinentes et son encouragement.*

*On tient aussi à remercier particulièrement les professeurs : **Djeddi Chawki** et **Daouadi Kheireddine**, qui ont accepté d'examiner et de juger notre travail.*

On les remercie pour leurs conseils, pour leurs disponibilités à tout moment, pour leurs gentillesse, et pour leurs aides illimitées.

On tient ensuite à remercier nos parents pour le soutien inconditionnel dont ils ont fait preuve depuis que notre projet est défini. Merci pour le soutien financier, moral, psychologique et matériel.

Enfin, on adresse nos plus sincères remerciements à tous nos proches et amis, qui nous ont toujours soutenues et encouragées au cours de la réalisation de ce mémoire.

Merci à tous et à toutes

Dédicace

Je dédie ce modeste travail :

À l'origine de mon hors de l'honneur, A la lumière de ma vie et la chose la plus précieuse que j'ai d'exister, c'est pourquoi mon succès : A ma mère.

A toi la bougie de ma vie, source de Tendresse infini que Dieu te garde et te protège : A mon père.

À mes chers frères et sœurs, pour leurs encouragements permanents, leur soutien moral, leur appui et leur encouragement,

À mes beaux-frères et leurs enfants

Pour leur présence dans ma vie

À mes amis et mes camarades

Sans oublier tous les professeurs

Et à tous ceux qui ont contribué de près ou de loin pour que ce projet soit possible,

Que ce travail soit l'accomplissement de vos vœux tant allégués, et le fruit de votre soutien infaillible,

Je veux dis merci.

Table des matières

Résumé	
المخلص	
Abstract	
Remerciements	
Dédicace	
Introduction générale	1
Chapitre 1	5
1.Introduction	5
2.Les caractéristiques spécifiques de l'écriture arabe	5
3.Différents aspects de reconnaissance de l'écriture :	6
3.1.Reconnaissance en-ligne	7
3.2.Reconnaissance hors-ligne	7
4.Approches de reconnaissance hors ligne	8
4.1.Approche globale.....	8
4.2.Approche analytique.....	8
5.Processus de reconnaissance hors ligne :	9
5.1.Acquisition.....	9
5.2.Phase de prétraitement	9
5.2.1.La binarisation.....	10
a)Seuillage global.....	10
b)Seuillage adaptif.....	11
5.2.2.Élimination de bruit	11
5.2.3.Le lissage.....	11
5.2.4.La squelettisation	12
5.2.5.La normalisation	12
a)Normalisation de taille	12
b)La normalisation de l'inclinaison (Redressement).....	13
5.3.Segmentation	13
5.3.1.Segmentation explicite	13
5.3.2.Segmentation implicite	13
5.4.Phase d'extraction des caractéristiques	14
5.5.Phase de classification	14
5.5.1.L'apprentissage.....	14

a)L'apprentissage supervisé.....	14
b)L'apprentissage non supervisé.....	15
5.5.2.Reconnaissance et décision	15
6. Les approches de reconnaissance	15
6.1.Approche statistique.....	15
6.2.Approche structurelle	15
6.3.Approche stochastique	16
6.4.Approche hybride	16
7.Méthodes de classification statistiques.....	16
7.1.Réseaux de neurones.....	16
7.2.Les Machines à Vecteur Support (SVM)	16
7.3.Le classifieur du K plus proches voisins	17
8.Post-traitements.....	17
9.Conclusion.....	18
Chapitre 2	19
1.Introduction	19
2.Caractéristiques d'une image numériques.....	19
2.1.LES PIXELS.....	19
2.2.Dimension	19
2.3.Résolution	20
2.4.Bruit	20
2.5.Histogramme.....	20
2.6.Texture.....	21
3.Extraction des caractéristiques	21
3.1.Sélection des caractéristiques	21
3.2.Types des caractéristiques	22
3.2.1.Caractéristiques topologiques ou métrique	22
3.2.2.Caractéristiques structurelles	22
3.2.3.Caractéristiques statistiques.....	22
3.2.4.Caractéristiques globales ou locales.....	23
3.2.5.Superposition des modèles et corrélation	23
4.Méthodes d'extractions des caractéristiques	23
4.1.LBP (Local Binary Pattern)	23
4.1.1.Principe de LBP	24
4.1.2.Méthodologie	24
4.2.CLBP (Compound Local Binary Pattern)	25
4.3.OBIF (Oriented Basic Image Feature)	27
4.3.1Oriented Basic Image Features (oBIFs).....	27

4.3.2.Caractéristiques de fond	28
4.4.Histogrammes oBIF column.....	29
5.Classification en SVM	31
5.1.Apprentissage statistique et SVM.....	32
5.2.Objectif de l'apprentissage statistique.....	32
5.3.Le sur-apprentissage et le sous-apprentissage.....	33
5.3.1.Sur-apprentissage	33
5.3.2.Sous-apprentissage	33
5.4.Principe de SVM.....	33
5.4.1.Préparation des données :	34
5.4.2.Sélection du noyau :	34
5.4.3.Détermination de l'hyperplan optimal :	34
5.4.4.Classification des nouvelles données	34
5.5.Types de SVM	34
5.5.1.SVM linéaire	34
5.5.2.SVM non linéaire.....	35
5.6.Fonctions noyaux (Kernel).....	35
5.7.Avantages et inconvénients des SVM	35
6.Conclusion	37
Chapitre3	38
1.Introduction	38
2.Description de différents travaux ultérieurs	38
2.1.La méthode proposée par M. Pechwitz et V. Maergner	38
2.2.La méthode proposée par P.Dreuw et al.....	38
2.3.La méthode proposée par A. Benouareth et al.....	38
2.4.La méthode proposée par R.A. Mohamad et al.....	39
2.5.La méthode proposée par J.H. AlKhateeb et al.....	39
2.6.La méthode proposée par J.H. Alkhateeb et al.....	39
2.7.La méthode proposée par H. Nemmour et Y. Chibani	39
2.8.La méthode proposée par M.T. Parvez et S.A. Mahmoud.....	40
2.9.La méthode proposée par H. Nemmour et Y. Chibani	40
2.10.La méthode proposée par Amrouch et al.....	40
2.11.La méthode proposée par Ghanim et al.....	40
3.Comparaison entre les travaux	40
4.Conclusion.....	42
Chapitre4	43
1.Introduction	44
2.Environnement de développement.....	44

2.1.Ressources matériels	Error! Bookmark not defined.
2.2.Ressource logiciel	Error! Bookmark not defined.
2.3.Base de données IFN/ENIT {v 2.0}.....	44
3.Schéma générale.....	45
4.Résultats et discussions.....	47
4.1.Expérimentation avec OBIF	47
4.1.1.Obif simple	47
4.1.2.Concaténations des vecteur obif	48
4.1.3.Avec normalisation	49
4.2.oBIF column.....	49
4.2.1.Avec normalisation	50
4.3.LBP	51
4.3.1.Concaténations des vecteur LBP.....	51
4.3.2.Avec normalisation	53
4.4.CLBP	53
4.4.1.Concaténations des vecteur LBP.....	54
4.4.2.Avec normalisation	55
5. 5. Résultats : Présentation Et Comparaison.....	Error! Bookmark not defined.
5.1.Comparez les résultats obtenus	55
5.2. Comparaison.....	Error! Bookmark not defined.
6.Conclusion.....	58
Conclusion générale	56
<i>Bibliographie.....</i>	<i>57</i>

LISTE DES FIGURES

Figure 1. Mots arabes incluant la Hamza, la Chadda et la Madda [25].	5
Figure 2. Lettres arabes ayant des points diacritiques. [1].	6
Figure 3. Classification des modes de reconnaissance des caractères.	7
Figure 4. Reconnaissance en-ligne. [23]	7
Figure 5. Reconnaissance hors-ligne.	8
Figure 6. schéma général d'un système de reconnaissance d'écriture manuscrite et chiffres [6].	9
Figure 7. effets de certaines opérations de prétraitement [24].	10
Figure 8. Principe de la binarisation par seuillage global [5].	11
Figure 9. Exemple d'élimination de bruit d'une image	11
Figure 10. Exemple de lissage [10].	12
Figure 11. Exemples de squelettisation [11].	12
Figure 12. Exemple d'un chiffre manuscrit normalisé. [12]	12
Figure 13. Les différents types de segmentation [2].	14
Figure 14. L'étape de la classification.	14
Figure 15. ensemble de pixels qui représente une image.	19
Figure 16. Exemple de dimension d'image numérique	19
Figure 17. Exemple de résolution d'image	20
Figure 18. Exemple de bruit dans l'image.	20
Figure 19. Exemple d'histogramme d'intensités de gris dans une image	21
Figure 20. Exemple de Calcul de base du LBP	24
Figure 21. Exemple Illustration du calcul du modèle binaire local (LBP) (a) exemple image, (b) patron binaire par seuillage et (c) matrice de poids	25
Figure 22. Illustrations de pixels locaux voisins.	25
Figure 23. LBP.	26
Figure 24. Calcul du CLPB (a): pattern d'un block 3x3 (b): la Différences locale (c): le Composant Signe (d): le Composant Magnitude.	26
Figure 25. Exemple de calcul des OBIF d'images manuscrites $\sigma = 8$ and $\epsilon 0.01$.	28
Figure 26. Configurations de concavité	29
Figure 27. Étiquettes de concavité pour l'image d'un mot manuscrit.	29
Figure 28. La division de l'image du chiffre en utilisant une grille uniforme de 2x1.	Error! Bookmark not defined.

Figure 29. Les étapes de l'histogramme des column oBIF (A) Image originale de l'écriture manuscrite (B) Calcul des oBIF pour les paramètres d'échelle $\sigma = 8$ et $\sigma = 4$ (C) Les oBIF à deux échelles sont croisés pour former des column à chaque emplacement (D) L'his 'histogramme est calculé avec c non plat.	31
Figure 30. Exemple de classification en SVM [48].	32
Figure 31. Les différents composants de base d'une SVM [50].....	34
Figure 32. Schéma général du système de reconnaissance de mot manuscrit.	46

Liste des symboles

SVM : Machine à Vecteur du Support.

LBP: Local Binary Pattern.

OBIF: Oriented Basic Image Feature.

HOG : Histogramme des Gradients Orientés.

CLBP: Compound Local Binary Pattern.

RDF : Reconnaissance de Forme.

REM : Reconnaissance de l'Écriture Manuscrite.

SEB : Suppressions des Espaces Blancs.

SPD : Suppressions des Points Diacritiques.

SREM : Système de Reconnaissance de l'Écriture Manuscrite.

LISTE DES TABLEAUX

Tableau 1. Différentes formes des caractères arabes selon leurs positions.[2]	6
Tableau 2. Classifieurs utilisés dans les systèmes de reconnaissance de mots arabes manuscrits	51
Tableau 3. Résultats de classification pour obif en quatre section (50,100,150,200).....	55
Tableau 4. Résultats de classification pour obif.....	56
Tableau 5. Résultats de classification pour obif Avec normalisation	57
Tableau 6. Résultats de classification pour obif column	57
Tableau 7. Résultats de classification pour obif column Avec normalisation	58
Tableau 8. Résultats de classification pour LBP (P, R)	58
Tableau 9. Résultats de classification pour LBP avec concaténation.....	59
Tableau 10. Résultats de classification pour LBP Avec normalisation	59
Tableau 11. Résultats de classification pour CLBP.....	60
Tableau 12. Résultats de classification pour CLBP avec concaténation.....	60 - 61
Tableau 13. Résultats de classification pour CLBP Avec normalisation	61
Tableau 14. Comparez les résultats obtenus.....	61
Tableau 15. Comparer les résultats obtenus avec d'autres travaux.....	62

Introduction Général

Introduction générale

Le système de reconnaissance automatique de l'écriture arabe manuscrite fait face à plusieurs problèmes qui mettent en évidence la complexité de la morphologie et de la cursivité de l'écriture arabe. Ces problèmes ont été largement étudiés ces dernières années, en particulier dans des domaines tels que l'automatisation du tri du courrier postal, le traitement des chèques, le traitement de formulaires et l'indexation automatique des manuscrits anciens.

Dans le domaine de la reconnaissance de l'écriture arabe manuscrite, on distingue deux types de systèmes de reconnaissance :

- Les systèmes de reconnaissance hors ligne : qui se concentrent sur la reconnaissance de l'écriture présente sur un support classique tel que le papier.
- Les systèmes de reconnaissance en ligne : qui sont réalisés à partir d'un stylo et d'une tablette électronique pouvant être assimilée à un écran, comme le papier électronique.

Dans notre travail, nous nous intéressons uniquement au système de reconnaissance hors ligne.

Généralement, il existe deux grandes approches : l'approche analytique et l'approche globale.

- L'approche analytique est basée sur la segmentation du mot : Elle consiste à diviser le mot manuscrit en parties inférieures appelées graphèmes. La principale difficulté de cette approche réside dans l'obtention robuste des bonnes hypothèses de segmentation. Cette approche est la seule applicable dans le cas de grands vocabulaires.
- L'approche globale se base sur une description unique de l'image du mot, considérée comme une entité indivisible. Cette approche ne peut être envisagée que pour de petits vocabulaires (quelques centaines de mots). L'approche globale, qui est beaucoup plus simple que l'approche analytique, permet de concevoir des systèmes de reconnaissance aux performances très intéressantes.

En général, le schéma principal des systèmes de reconnaissance des mots arabes cursifs comprend cinq phases importantes : le prétraitement, la segmentation, l'extraction des caractéristiques, la reconnaissance et le post-traitement.

Notre travail se concentre sur la reconnaissance hors ligne des mots arabes manuscrits en utilisant une approche globale avec un vocabulaire limité, en utilisant la base de données IFN/ENIT qui contient des mots arabes manuscrits représentant les noms des villes/villages tunisiens.

Problématique et objectifs

La problématique de la reconnaissance automatique de l'écriture arabe manuscrite réside dans les défis spécifiques liés à la langue arabe elle-même. Bien que de nombreux chercheurs aient tenté de développer des systèmes de reconnaissance des mots arabes, il n'existe toujours pas de système parfait en raison de plusieurs raisons, notamment :

- Variation des styles d'écriture : La langue arabe présente une grande diversité de styles d'écriture en raison des différences régionales, des contextes culturels et des préférences individuelles. La variabilité des styles d'écriture rend difficile la conception d'un système qui puisse reconnaître avec précision tous ces styles.
- Cursivité de la langue arabe : L'écriture arabe est essentiellement cursive, ce qui signifie que les lettres sont liées entre elles dans un mot. Cette cursivité complexifie la reconnaissance automatique, car les formes des lettres peuvent changer en fonction de leur position dans le mot et de leurs connexions avec les lettres adjacentes.
- Complexité de la morphologie des caractères : Les caractères arabes ont une morphologie complexe. Chaque lettre peut prendre différentes formes en fonction de sa position dans le mot et de ses relations avec les lettres environnantes. La variabilité morphologique complique la sélection de primitives (caractéristiques) pertinentes pour décrire les caractères arabes de manière efficace.
- Chevauchement des mots : Dans l'écriture arabe manuscrite, il est courant que les mots se chevauchent, ce qui rend la segmentation précise des mots plus difficile. La reconnaissance automatique doit être capable de gérer ce chevauchement et de segmenter correctement les mots pour une analyse précise.
- Variations individuelles : Chaque individu a une écriture manuscrite spécifique qui lui est propre, avec des différences dans la forme des lettres, la taille et la disposition des mots. Ces variations individuelles ajoutent une complexité supplémentaire à la

reconnaissance automatique, car les modèles doivent être adaptés à chaque style d'écriture spécifique.

Les objectifs de la recherche dans ce domaine visent à développer des méthodes et des algorithmes robustes qui surmontent ces problèmes, permettant ainsi une reconnaissance automatique plus précise et fiable de l'écriture arabe manuscrite.

Nous nous sommes fixé comme objectif d'avoir un système efficace capable de reconnaître la majorité des mots avec un taux de reconnaissance accru, pour cela nous proposons d'utiliser deux types de descripteurs : LBP (Local Binary Patterns), CLBP (Composite Local Binary Patterns) et OBIF (Oriented Basic Image Feature) et des descripteurs de column OBIF, utilisés comme classificateur SVM pour identifier ces mots, extraits de la base de données IFN/ENIT.

Présentation du mémoire

Cette thèse est subdivisée en quatre chapitres comme suit :

Chapitre1. La reconnaissance de l'écriture arabe manuscrit

Dans ce chapitre, nous avons abordé le sujet de la reconnaissance des caractères en se concentrant particulièrement sur les particularités de l'écriture arabe, ses styles divers, les aspects variés de la reconnaissance des caractères et les différentes étapes du processus de reconnaissance.

Chapitre 2. Extraction des caractéristiques

Ce chapitre se concentre sur l'extraction des caractéristiques pour la reconnaissance des mots arabes manuscrits. Il présente différentes méthodes utilisées pour extraire des informations visuelles à partir des images des mots arabes manuscrits, telles que la texture, les contours et les orientations. La méthode d'extraction des caractéristiques choisie pour le système de reconnaissance est détaillée, en expliquant les principes et les étapes de traitement des images. Les concepts d'apprentissage et de classification sont également abordés, expliquant comment entraîner un modèle à partir des données d'entraînement et utiliser un classifieur pour prendre des décisions de classification. Ce chapitre fournit une compréhension approfondie des méthodes d'extraction des caractéristiques pour la reconnaissance des mots arabes manuscrits, ainsi que des concepts clés liés à l'apprentissage et à la classification.

Chapitre 3. Etat de l'art

Ce chapitre a pour objectif de présenter un état de l'art des travaux existants dans la reconnaissance des mots arabes manuscrits. Il vise à recenser et à comparer les différentes approches et méthodes utilisées, en se basant sur les résultats obtenus par chaque étude.

La comparaison entre les travaux se fera en fonction de divers critères tels que le taux de reconnaissance, la taille de la base de données utilisée, les caractéristiques extraites, les algorithmes de classification employés, etc.

Chapitre 4. Résultats expérimentaux

Ce chapitre présente l'environnement de travail ainsi que les résultats obtenus pour le système de reconnaissance de mots manuscrits. Il offre aux lecteurs une vision globale du contexte et des performances du système.

Le travail se termine par une conclusion générale

Chapitre 1

La reconnaissance de l'écriture arabe
manuscrite

1. Introduction

La reconnaissance de l'écriture manuscrite, et en particulier de l'écriture arabe, est une tâche complexe et difficile pour les systèmes de reconnaissance automatique de textes écrits. Ce chapitre vise à analyser les différentes méthodes et outils utilisés pour la reconnaissance de l'écriture arabe manuscrite, en passant en revue les caractéristiques de la langue arabe, les approches de reconnaissance et les étapes du processus de reconnaissance, de l'acquisition à la post-traitement. La reconnaissance de l'écriture manuscrite présente un intérêt important dans de nombreux domaines d'application, tels que la lecture de chèques, la reconnaissance de formulaires administratifs, le tri de courrier, l'indexation et l'archivage de documents, la reconnaissance de numéros minéralogiques, la recherche d'informations dans une base de documents manuscrits, et même l'aide à la lecture pour les personnes aveugles.

2. Les caractéristiques spécifiques de l'écriture arabe

L'écriture arabe se distingue des autres langues par sa structure et la liaison entre les caractères formant un mot, ce qui la rend difficile à reconnaître pour les logiciels de reconnaissance. [1]

- La plupart des lettres arabes sont liées les unes aux autres.
- Les caractères arabes peuvent contenir un trait vertical (ظ), oblique (ك) ou en zigzag(أ).
- Les caractères arabes n'ont pas de taille fixe.
- Six lettres ne s'attachent jamais à la lettre suivante : "أ", "ز", "ر", "د", "ذ", "و".
- Certains mots ne se distinguent que par des signes diacritiques, notamment fatha (◌َ), damma (◌ِ) et kasra (◌ِ).
- D'autres types de signes diacritiques incluent la Hamza (ء), la Chadda (ّ) et la Madda (◌◌).



Figure 1. Mots arabes incluant la Hamza, la Chadda et la Madda [25].

- Dans l'alphabet arabe, 15 lettres parmi les 28 possèdent un ou plusieurs points et dans des positions différentes. [1]

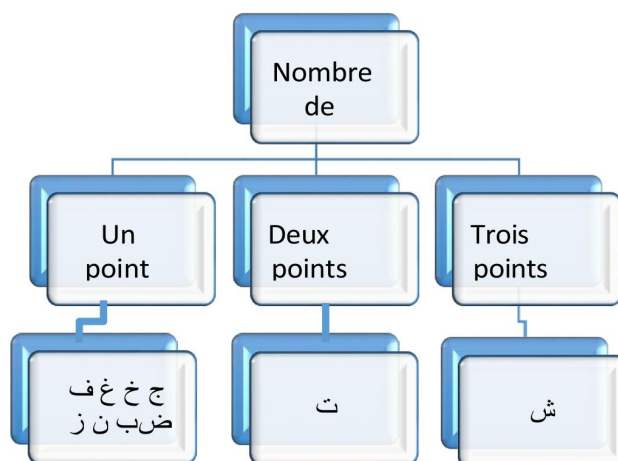


Figure 2. Lettres arabes ayant des points diacritiques. [1]

- L'alphabet arabe comporte 28 lettres fondamentales, chacune des lettres arabes se déclinesous plusieurs formes suivant sa place dans le mot

Caractère.	Au début	Au milieu	Fin	Isolé	Car.	Au début	Au milieu	Fin	Isolé
alif	أ	لا ^ا	لا	إ	dhad	ض	ض	ض	ض
ba	ب	ب	ب	ب	tad	ط	ط	ط	ط
tâ	ت	ت	ت	ت	thad	ظ	ظ	ظ	ظ
thâ	ث	ث	ث	ث	ayn	ع	ع	ع	ع
jim	ج	ج	ج	ج	ghayn	غ	غ	غ	غ
ha	ح	ح	ح	ح	fa	ف	ف	ف	ف
kha	خ	خ	خ	خ	kaf	ق	ق	ق	ق
del	د	د ^ا	د	د	kef	ك	ك	ك	ك
dhel	ذ	ذ ^ا	ذ	ذ	lam	ل	ل	ل	ل
ra	ر	ر ^ا	ر	ر	mim	م	م	م	م
zei	ز	ز ^ا	ز	ز	noun	ن	ن	ن	ن
sin	س	س	س	س	ha	ه	ه	ه	ه
shin	ش	ش	ش	ش	wew	و	و ^ا	و	و
sad	ص	ص	ص	ص	ya	ي	ي	ي	ي

Tableau 1 Différentes formes des caractères arabes selon leurs positions.[2]

3. Différents aspects de reconnaissance de l'écriture :

La reconnaissance de l'écriture manuscrite peut être divisée en reconnaissance en ligne et reconnaissance hors ligne :

Le choix entre la reconnaissance en-ligne et la reconnaissance hors-ligne dépend des besoins et des objectifs de l'application. Chaque mode de reconnaissance présente des caractéristiques, des outils spécifiques et des algorithmes d'extraction qui leur sont propres, et qui peuvent être plus adaptés à certains domaines et utilisations que d'autres.

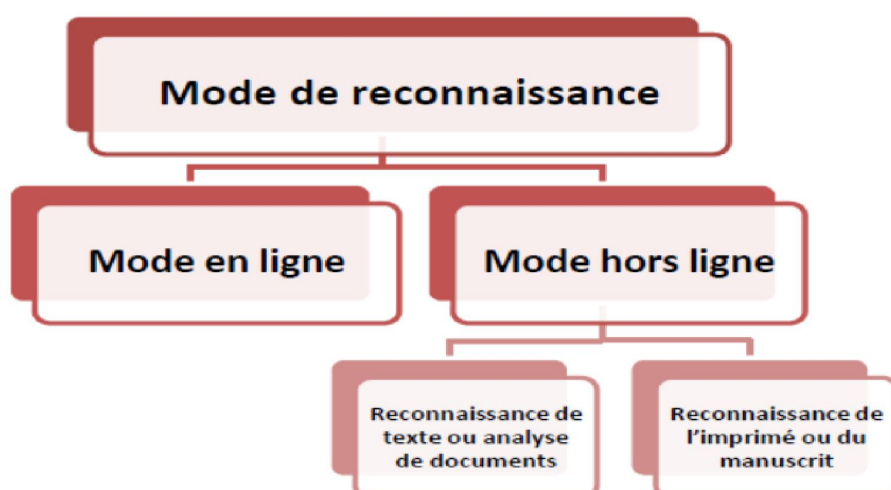


Figure 3. Classification des modes de reconnaissance des caractères.

3.1. Reconnaissance en-ligne

Il s'agit d'un mode qui fonctionne en temps réel (lors de la saisie de contenu). Dans une application de reconnaissance d'écriture manuscrite en ligne, l'utilisateur écrit sur une table spéciale. Ces systèmes sont utilisés dans une variété d'appareils électroniques, tels que les ordinateurs personnels ou les tablettes. Les données sont ensuite présentées sous la forme d'une série de points ordonnés. Grâce à un dispositif spécifique (comme illustré à la Figure 4), le signal d'écriture est enregistré directement lors de sa génération, puis le système reconnaîtra l'écriture et enverra le résultat à l'écran.



Figure 4 .Reconnaissance en-ligne.[23]

3.2. Reconnaissance hors-ligne

La reconnaissance hors-ligne des mots arabes manuscrits est un processus complexe qui implique la numérisation de l'image, le prétraitement de l'image, la segmentation des mots, l'extraction de caractéristiques, la reconnaissance, et la correction des erreurs. Les algorithmes de reconnaissance tels que les réseaux de neurones artificiels sont utilisés pour identifier les mots manuscrits avec une précision élevée. L'ensemble du processus nécessite une expertise technique et des compétences en traitement d'images et en apprentissage automatique pour atteindre une précision élevée dans la reconnaissance des mots arabes manuscrits.

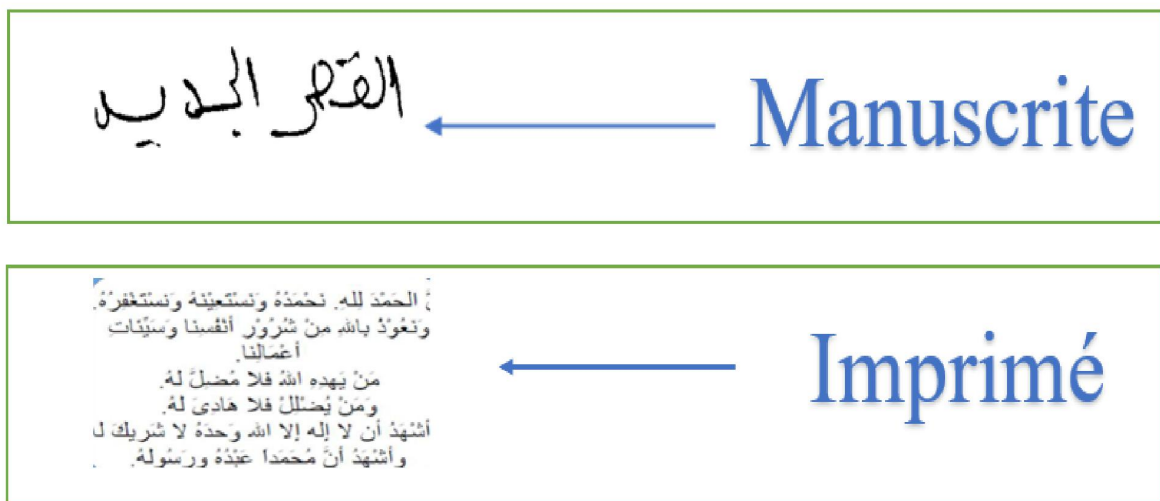


Figure 5.Reconnaissance hors-ligne

4. Approches de reconnaissance hors ligne

En reconnaissance hors-ligne des mots manuscrits et chiffres il existe deux approches :

- Une approche globale.
- Une approche analytique.

4.1. Approche globale

L'approche globale se base sur une description unique de l'image du mot, vue comme une entité indivisible [4].

Disposant de beaucoup d'informations, en effet, la discrimination des mots proches est très difficile, et l'apprentissage des modèles nécessite une grande quantité d'échantillons qui est souvent difficile à réunir. Cette approche est souvent appliquée pour réduire la liste des mots candidats dans le contexte d'une reconnaissance à vocabulaire réduits.

4.2. Approche analytique

L'approche analytique pour la reconnaissance de l'écriture manuscrite consiste à segmenter l'image en isolant les caractères pour en extraire les fragments morphologiques significatifs. Cette segmentation permet d'identifier les entités qui sont ensuite utilisées pour reconnaître les mots. Cependant, cette approche peut générer plusieurs types d'erreurs. Le processus de reconnaissance est donc un processus de relaxation qui alterne entre deux phases : la segmentation et l'identification des segments [5].

5. Processus de reconnaissance hors ligne :

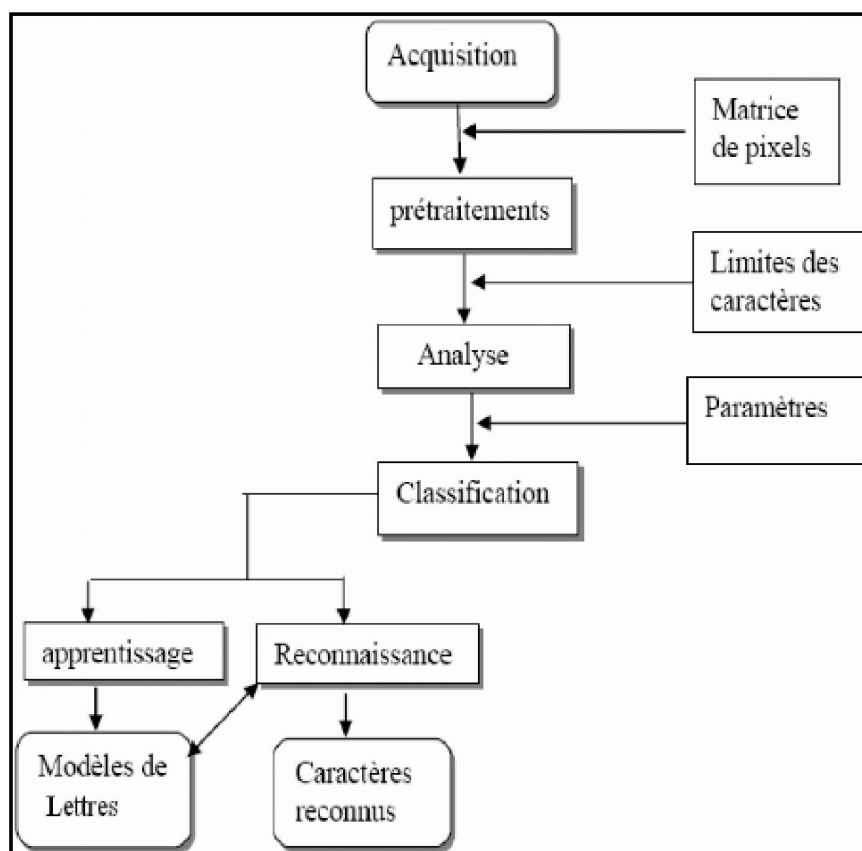


Figure 6. Schéma général d'un système de reconnaissance d'écriture manuscrite et chiffres [6].

5.1. Acquisition

L'étape d'acquisition est une phase préliminaire qui vise à obtenir les données à partir d'un document papier en utilisant un périphérique de numérisation pour le convertir en une image numérique (bitmap). Cette étape est cruciale car elle implique la préparation des documents à traiter, la sélection et la configuration du matériel de numérisation, tel que le scanner, et la détermination du format de stockage des images. [7]

5.2. Phase de prétraitement

Le prétraitement de l'image a pour objectif de préparer le document pour le traitement ultérieur en appliquant différentes opérations de prétraitement. Ces opérations, qui peuvent être appliquées de manière indépendante ou successive, comme le lissage ou l'alignement des caractères, l'uniformisation de l'épaisseur du trait, etc., visent à améliorer la qualité de l'image pour faciliter les traitements ultérieurs. [7]

Parmi les opérations de prétraitement on peut citer : la binarisation, élimination de bruit, le lissage, la squelettisation et la normalisation.

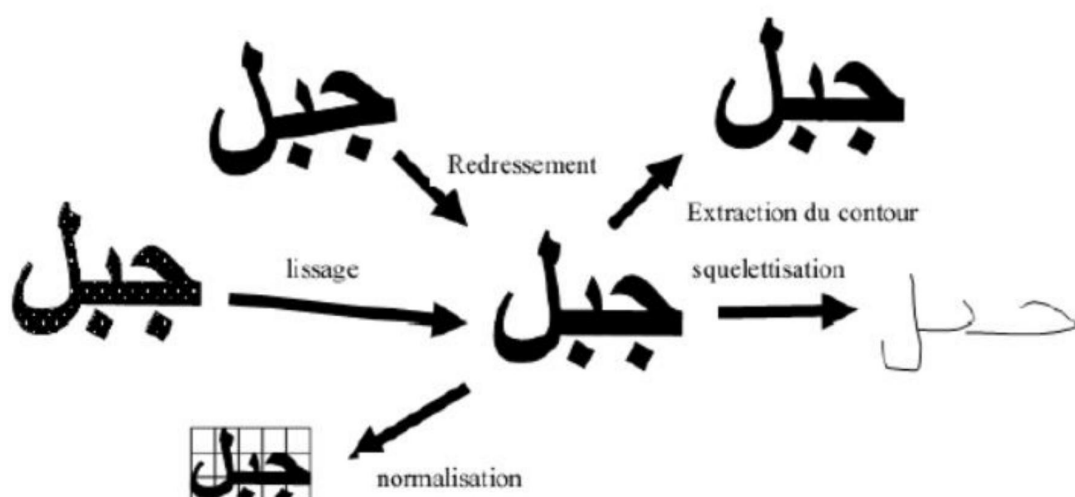


Figure 7. Effets de certaines opérations de prétraitement [24].

5.2.1. La binarisation

Les systèmes de reconnaissance nécessitent une étape de binarisation qui vise à séparer les pixels de texte des pixels de l'arrière-plan de l'image traitée. En fait, la Plupart des systèmes ne fonctionnent que sur des images binaires. La plus simple façon pour obtenir une image binaire est de choisir une valeur seuil, puis de classer tous les pixels dont les valeurs sont au-dessus de ce seuil comme étant des pixels d'arrière plan, et tous les autres pixels comme étant des pixels de texte.

Soit l'image $I(M \times N)$, supposons que $f(x,y)$ représente le niveau de gris du pixel aux coordonnées (x,y) , $0 \leq x \leq M$, $0 \leq y \leq N$ et S est le seuil choisi, les pixels de l'objet sont ceux ayant le niveau de gris inférieur à S et les autres ayant le niveau de gris supérieur à S sont des pixels du fond. Alors, l'image binarisée G est déterminée par les pixels (x,y) dont la valeur est donnée par l'équation (1) [8] :

$$G(x,y) = \begin{cases} 1 & \text{si } f(x,y) > S \\ 0 & \text{si } f(x,y) \leq S \end{cases} \quad (1)$$

La plupart des de binarisation des images de texte se basent ces deux types :

a) Seuillage global

Seuillage global Le seuillage global consiste à prendre un modifiable seuil, mais identique pour toute l'image. La méthode de seuillage globale consiste à calculer un seuil unique à partir d'une mesure globale sur toute l'image et la recherche de seuil s'effectue par l'analyse de l'histogramme des niveaux de gris et par la détermination d'un minimum local (voir figure ci-dessous). Il nous donner la possibilité de décider l'appartenance d'un pixel à l'objet ou au fond sur toute l'image. Tel que les pixels ayant un niveau de gris inférieur au seuil sont mis en noir et les autres en blanc.

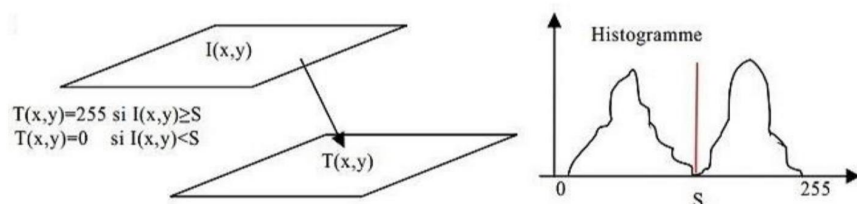


Figure 8. Principe de la binarisation par seuillage global [5].

Cette méthode est fiable pour les documents simples et de bonne qualité (visuel).

b) Seuillage adaptatif

Est plus approprié dans le cas contraire, le calcul du seuil se fait localement en fonction du voisinage du pixel traité. Et le seuillage adaptatif.

Le but de la binarisation est de faire surgir l'information utile par rapport à l'arrière-plan.

5.2.2. Élimination de bruit

L'élimination de bruit est un processus visant à réduire les variations aléatoires ou parasites d'intensité de pixels dans une image numérique. Cela peut être fait en appliquant des filtres spécifiques pour réduire les variations de luminance ou de couleur, ou en utilisant des techniques de traitement d'image telles que la restauration d'image ou la réduction de bruit par seuillage. Le but est d'améliorer la qualité de l'image en éliminant les perturbations qui peuvent masquer ou altérer les informations importantes.

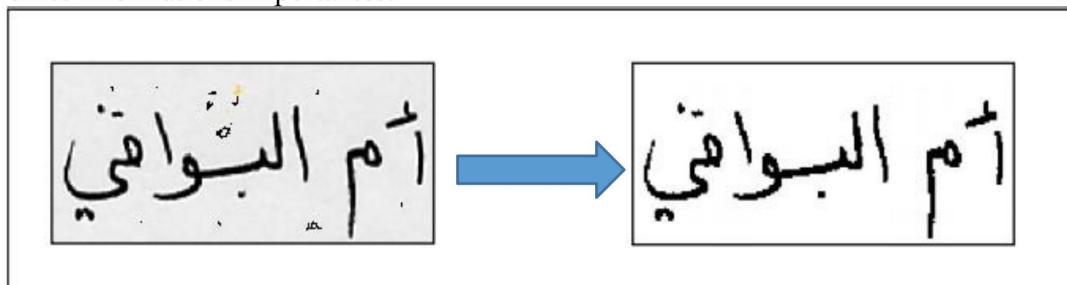


Figure 9. Exemple d'élimination de bruit d'une image

5.2.3. Le lissage

Le processus de lissage consiste à minimiser les irrégularités et les interruptions qui peuvent apparaître lors de la transformation d'une image, afin de restaurer la fluidité et la continuité des contours des caractères. En examinant les pixels environnants, si le nombre de pixels noirs dans cette zone dépasse un seuil donné, un pixel se verra attribuer la valeur 1. Les bruits dans l'image des caractères peuvent être causés par des artefacts d'acquisition ou une qualité de document médiocre, ce qui peut conduire à une sous-représentation ou une sur-représentation des points. Le lissage résout ces problèmes en effectuant des opérations locales telles que le bouchage et le nettoyage. L'opération de nettoyage supprime les petites taches et les saillies, tandis que le bouchage consiste à égaliser les contours et à remplir les trous internes en ajoutant des pixels noirs à la forme du caractère [9].

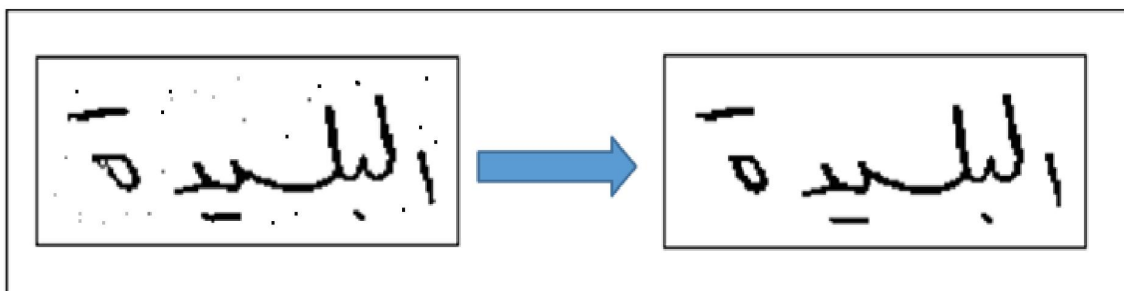


Figure 10. Exemple de lissage [10].

5.2.4. La squelettisation

La technique de squelettisation permet d'obtenir une épaisseur uniforme de 1 pour le trait d'écriture, ce qui permet de le représenter sous forme linéaire. Il est important que le squelette résultant conserve la forme, la connectivité, la topologie et les extrémités du tracé, tout en évitant d'introduire des éléments superflus qui pourraient perturber la représentation.



Figure 11. Exemples de squelettisation [11].

5.2.5. La normalisation

Une étape de normalisation est nécessaire, car le problème majeur pour la segmentation est la taille variable des images et l'inclinaison des lignes.

a) Normalisation de taille

La normalisation est une étape clé qui permet de redimensionner les images de chiffres manuscrits à des tailles standardisées. Bien que cela puisse entraîner des déformations, cette étape est essentielle pour les systèmes de reconnaissance qui sont sensibles aux variations de taille et de position, tels que les méthodes de corrélation et de comparaison de modèles. En normalisant les images, il devient plus facile de comparer et de reconnaître les caractères manuscrits avec précision [12].

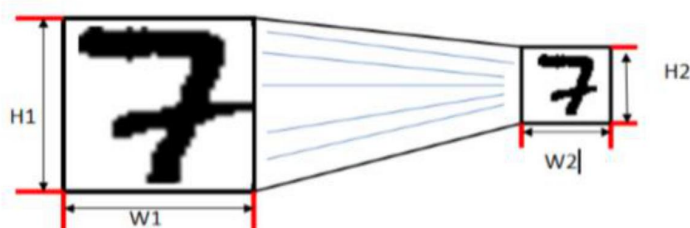


Figure 12. Exemple d'un chiffre manuscrit normalisé. [12]

b) La normalisation de l'inclinaison (Redressement)

Le redressement est une technique permettant de corriger l'inclinaison des images numérisées de documents manuscrits, que ce soit à cause d'un mauvais positionnement du papier dans le scanner ou d'une inclinaison interne des parties du document. Si le document n'est pas bien ajusté, plusieurs algorithmes de reconnaissance peuvent donner de mauvais résultats. Le redressement s'effectue en deux étapes : le redressement de la ligne de base pour rendre les chiffres horizontaux en utilisant une transformation géométrique de type rotation isométrique des points de l'image, et le redressement des écritures penchées pour faciliter la segmentation préalable des chiffres en trouvant l'angle moyen d'inclinaison et en appliquant une transformation géométrique de type cisaillement de l'image.

5.3. Segmentation

L'étape de segmentation peut diviser l'image en différentes vignettes de plus petite taille, qui peuvent être des graphèmes, des lettres ou même des sous-mots. Cependant, l'image est toujours une matrice de pixels. Il existe deux techniques pour réaliser la segmentation : la segmentation explicite et la segmentation implicite.

5.3.1. Segmentation explicite

La segmentation explicite ou discrète consiste à sélectionner les points les plus probables de segmentation, puis à diviser les composants de l'image en suivant un chemin en raison de connexions multiples ou de contacts prolongés. Cette analyse peut utiliser différents critères tels que :

- Analyse des contours de la forme du squelette.
- Analyse de l'amincissement du fond.
- Analyse bidimensionnelle du tracé.
- Combinaison d'analyse des contours et d'amincissement du fond.

Cette approche présente l'avantage de localiser explicitement l'information, car la séparation des lettres est effectuée non pas selon leur reconnaissance, mais selon des critères topologiques ou morphologiques. Cependant, il n'existe pas de méthode de segmentation parfaitement fiable à 100%, car toute erreur de segmentation peut pénaliser les performances du système. [13]

5.3.2. Segmentation implicite

La segmentation basée sur les modèles de lettres implique une découpe préalable de l'image en intervalles de taille régulière. Cette technique est similaire à celle utilisée dans la reconnaissance de la parole où le signal est également découpé en intervalles de temps réguliers. Contrairement à la segmentation explicite, cette approche ne nécessite pas de pré-segmentation du mot. La segmentation est effectuée pendant la reconnaissance et guidée par celle-ci. Le système recherche des composantes ou des groupes de graphèmes dans l'image qui correspondent à ses classes de lettres. [14].

L'avantage de cette segmentation c'est que l'information est localisée par les modèles des lettres et la validation ce fait par ces modèles. Dans les approches à segmentation implicite, la tâche de segmentation est accomplie par le système, elle est soutenue simultanément par un processus de reconnaissance, en évitant la pré-segmentation d'un mot en lettre ou entités plus fines [13].

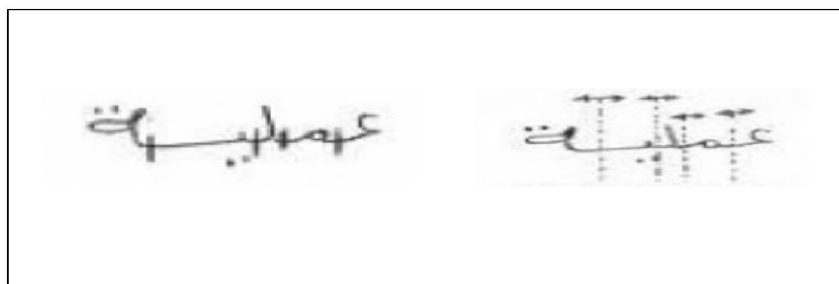


Figure 13. Les différents types de segmentation [2].

5.4. Phase d'extraction des caractéristiques

L'extraction de caractéristiques, souvent appelées primitives, consiste à représenter les données d'entrée (mots, caractères, graphèmes) avec des vecteurs primitifs de dimensions fixes. Il s'agit d'une étape cruciale dans l'identification du système. Le but de cette phase est de sélectionner des informations pertinentes, discriminantes et de taille limitée pour l'étape de classification, tout en évitant le risque de perdre des informations importantes. En effet, même si l'on utilise des classificateurs très efficaces, un mauvais choix de primitives peut avoir un impact négatif notable sur les résultats. [2].

5.5. Phase de classification

La classification est l'étape où les attributs des formes sont transformés en classe d'appartenance à l'aide d'une règle de décision. Dans un système de reconnaissance de mots hors-ligne, la classification comprend l'apprentissage et la reconnaissance et décision qui consistent à déterminer la classe d'appartenance de l'objet d'entrée. Les caractéristiques extraites dans l'étape précédente sont utilisées pour identifier un segment de texte et le comparer à un modèle de référence [12].



Figure 14. L'étape de la classification.

5.5.1. L'apprentissage

L'apprentissage consiste à fournir au système un ensemble de formes préalablement étiquetées pour qu'il puisse apprendre à reconnaître des formes inconnues. Il existe deux types d'apprentissage : supervisé et non supervisé. Dans l'apprentissage supervisé, les exemples sont étiquetés avec leur classe, tandis que dans l'apprentissage non supervisé, il n'y a pas d'étiquettes de classe. [15].

a) L'apprentissage supervisé

Ce type d'apprentissage est appelé supervisé car il implique la présence d'un "professeur" qui guide le processus en fournissant des exemples de référence pour chaque classe. Cette étape préliminaire de reconnaissance nécessite l'insertion d'un grand nombre d'échantillons de référence, dont le nombre peut varier selon l'application, allant de quelques unités à plusieurs

centaines par caractère. Le choix des caractères de référence est effectué manuellement et dépend de l'objectif de l'application.

b) L'apprentissage non supervisé

L'apprentissage non supervisé ou sans professeur consiste à doter le système d'un mécanisme automatique qui s'appuie sur des règles précises de regroupement pour trouver les classes de référence avec une assistance minimale. Dans ce cas les échantillons sont introduits en un grand nombre par l'utilisateur sans indiquer leur classe [16].

5.5.2. Reconnaissance et décision

La décision finale d'un système de reconnaissance de caractères est l'étape où le module de reconnaissance cherche parmi les modèles de référence disponibles ceux qui correspondent le plus aux paramètres de la forme de caractère traitée. Si un seul modèle répond à cette description, le système de reconnaissance obtient un succès [9]. Cependant, s'il existe plusieurs modèles de référence qui correspondent à la description, le système peut être confronté à une confusion, et dans le cas où aucun modèle de référence ne correspond à la description, le système peut décider de rejeter la forme [15]. Dans les deux premiers cas, la décision peut être accompagnée d'une mesure de vraisemblance, appelée score ou taux de reconnaissance.

6. Les approches de reconnaissance

Il y a quatre approches de classification, dont une approche hybride et trois approches fondamentales

6.1. Approche statistique

L'approche statistique permet de prendre une décision de classification d'un objet inconnu en se basant sur une description extensive des classes. Cette approche nécessite un grand nombre d'exemples pour réaliser un apprentissage correct des lois de calcul des probabilités des classes. Elle est fondée sur l'étude statistique des mesures opérées sur les formes à reconnaître. En se basant sur des théories fondées telles que la règle de Bayes, les réseaux de neurones et les chaînes de Markov, elle bénéficie des méthodes d'apprentissage automatique [17].

Les méthodes statistiques comprennent quatre étapes :

- le choix des paramètres.
- Une description statistique des classes.
- Calcul de la distance d'une forme à chaque classe.
- Choix de la classe la plus proche.

6.2. Approche structurelle

Les méthodes structurelles se basent sur la structure physique des caractères, en cherchant à identifier des éléments simples ou primitifs tels que les graphèmes ou les pixels de l'image et à décrire leurs relations. Contrairement aux méthodes statistiques, ces primitives sont de type topologique ou de forme élémentaire plutôt que des mesures. Il existe plusieurs techniques structurelles telles que : structures syntaxiques, la comparaison des chaînes et la programmation dynamique [10].

6.3. Approche stochastique

Contrairement aux méthodes précédemment décrites, l'approche stochastique utilise un modèle pour la reconnaissance, prenant en compte la grande variabilité de la forme. La distance communément utilisée dans les techniques de « comparaison » est remplacée par des probabilités calculées de manière plus fine par apprentissage. La forme est considérée comme un signal continu observable dans le temps à différents endroits constituant des états « d'observations ». Le modèle décrit ces états à l'aide de probabilités de transitions d'états et de probabilités d'observation par état.

La comparaison consiste à chercher dans ce graphe d'états, le chemin de probabilité forte correspondant à une suite d'éléments observés dans la chaîne d'entrée [11]. Les méthodes les plus répondues dans cette approche sont les méthodes utilisant les modèles de Markov cachés (H.M.M).

6.4. Approche hybride

Afin d'améliorer la précision de la reconnaissance de caractères, la pratique courante consiste à combiner plusieurs types de caractéristiques et à intégrer plusieurs classifieurs en couches au sein d'un même système. Cette approche permet de compenser les faiblesses de chaque méthode et de fournir des résultats plus précis qu'en utilisant chaque méthode séparément. Par conséquent, les approches combinées sont souvent préférées aux approches individuelles pour la reconnaissance de caractères [18].

7. Méthodes de classification statistiques

L'approche statique a été conçu pour une reconnaissance des textes manuscrites et plus particulièrement la reconnaissance des chiffres sur des bases théoriques telle que les méthodes paramétriques, elles englobent plusieurs méthodes parmi ces dernières en peut citer

7.1. Réseaux de neurones

Un réseau neuronal est un ensemble de neurones artificiels interconnectés qui permettent de réaliser des fonctions algébriques complexes en composant les fonctions réalisées par chacun des neurones. Il est capable de résoudre des problèmes tels que le traitement du langage naturel, la reconnaissance de formes et d'écriture manuscrite en s'inspirant du fonctionnement des neurones biologiques. Le réseau neuronal peut également apprendre et résoudre des problèmes sans programmation classique [19].

Le réseau neuronal convolutif est une méthode de classification d'images qui se base sur l'idée que le modèle fonctionne mieux lorsqu'il comprend les caractéristiques locales de l'image. Il utilise moins de paramètres qu'un réseau entièrement connecté en réutilisant plusieurs fois le même paramètre. Au lieu de générer des poids à partir de chaque pixel de l'image, un réseau neuronal convolutif génère suffisamment de poids pour couvrir une petite zone de l'image à la fois. Lorsqu'un modèle CNN est formé pour classer une image, il recherche les entités à leur niveau de base en examinant les courbures des limites de ces entité [16].

7.2. Les Machines à Vecteur Support (SVM)

Les réseaux de neurones sont confrontés à des problèmes de traitement de données très volumineuses, ainsi qu'à un grand nombre de paramètres d'apprentissage. Pour résoudre ce problème, il est souvent nécessaire de réduire la dimension de l'espace d'entrée en choisissant certains attributs des données, mais cela nécessite souvent des hypothèses simplificatrices qui ne sont pas toujours vérifiées en pratique. Heureusement, les Support Vector Machines (SVM) sont une méthode de formulation pour la théorie de l'apprentissage statistique qui permet de surmonter

ces difficultés. Introduites par Vapnik en 1995, les SVM exigent une liaison linéaire entre le nombre de paramètres et de données d'apprentissage, ce qui les rend solides en termes de garanties théoriques, flexibles et simples à utiliser. Les SVM sont une famille d'algorithmes d'apprentissage automatique qui peuvent être utilisés pour résoudre des problèmes de classification, de régression ou de détection d'anomalies.

7.3. Le classifieur du K plus proches voisins

L'algorithme des "k plus proches voisins" est une méthode simple d'apprentissage supervisé utilisée pour la régression et la classification. Contrairement à d'autres méthodes, elle ne nécessite pas de paramètres préétablis, le modèle stocke simplement les observations du groupe d'apprentissage pour la classification des données du groupe de test. Cet algorithme est qualifié de "paresseux" car il n'apprend pas pendant l'étape d'entraînement. Pour prédire la classe d'une nouvelle donnée d'entrée, il recherche les K voisins les plus proches (en utilisant la distance euclidienne ou autre) et sélectionne la classe majoritaire parmi ces voisins. Pour déterminer le paramètre K lors de l'implémentation de l'algorithme, il est possible de faire varier K et pour chaque valeur de K, estimer le taux d'erreur de l'ensemble de test et stocker le paramètre K qui minimise ce taux d'erreur [16] [20].

8. Post-traitements

La phase de post-traitement vise à améliorer le taux de reconnaissance des mots obtenus après la classification, en utilisant des informations contextuelles telles que la fréquence d'apparition des mots et les lexiques. Étant donné que la classification peut produire plusieurs résultats possibles, le post-traitement utilise des informations de niveaux supérieurs (syntaxiques, lexicales, sémantiques) pour sélectionner la solution la plus appropriée et pour corriger les erreurs éventuelles [19]. De plus, cette phase évalue la pertinence de la réponse obtenue en se basant sur des données non disponibles lors de la classification.

9. Conclusion

Dans ce chapitre, nous avons abordé le sujet de la reconnaissance des lettres, avec un accent particulier sur les caractéristiques de la calligraphie arabe, ses différents styles, les différents aspects de la reconnaissance des lettres et les différentes étapes du processus de reconnaissance. Le prochain chapitre de ce travail sera consacré à la revue des techniques et méthodes les plus couramment utilisées pour extraire des caractéristiques à partir d'images. Nous nous intéresserons ensuite au cheminement des classifieurs CLBP, LBP, OBIF, OBIF column et le classifieur SVM en les présentant en détail.

Chapitre 2

Extraction des caractéristiques

1. Introduction

La phase d'extraction des caractéristiques est cruciale car elle influence directement l'efficacité du système de reconnaissance. Le choix de la méthode d'extraction des caractéristiques est donc déterminant pour obtenir de bonnes performances de reconnaissance. Ce chapitre se concentre sur la présentation des différentes méthodes d'extraction des caractéristiques. Une partie est consacrée aux caractéristiques d'images et expose les méthodes utilisées pour les extraire. Ensuite, une deuxième partie présente en détail la méthode d'extraction des caractéristiques choisie. Enfin, les concepts d'apprentissage et le classifieur sélectionné pour traiter les données sont également abordés.

2. Caractéristiques d'une image numériques

Une image numérique est une représentation d'image réelle sous la forme d'un ensemble de pixels structurés (éléments d'image) et qui sont caractérisés par les paramètres suivants :

2.1. LES PIXELS

Une image numérique est composée de groupes de points appelés pixels (abréviation de PICTure Element) pour former une image. Ainsi, un pixel représente le plus petit élément constitutif d'une image numérique. Tous ces pixels sont contenus dans le tableau 2D qui compose l'image [21].

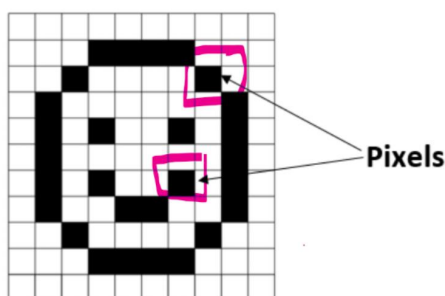


Figure 15. Ensemble de pixels qui représente une image.

2.2. Dimension

La dimension d'une image numérique représente sa taille, et est exprimée sous forme d'une matrice composée de pixels. Cette matrice est formée par un nombre de lignes et un nombre de column, et chaque élément de la matrice représente une valeur numérique correspondant à l'intensité lumineuse d'un pixel. Le nombre total de pixels dans une image est obtenu en multipliant le nombre de lignes par le nombre de column de la matrice.

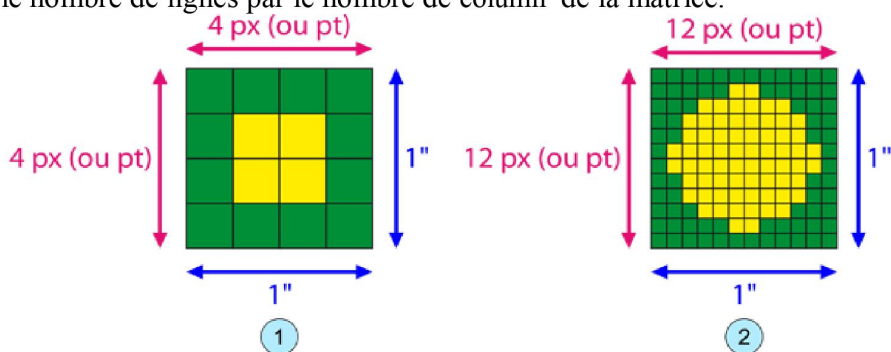


Figure 16. Exemple de dimension d'image numérique

2.3. Résolution

La résolution d'une image numérique se réfère au niveau de clarté et de détails fins affichés par un écran ou une imprimante. Elle est généralement exprimée en termes de pixels par unité de longueur, comme le nombre de pixels par pouce ou PPI (pixels per inch), également appelé DPI (dots per inch). Une résolution plus élevée correspond à un plus grand nombre de pixels et donc à une plus grande quantité d'informations et de précision dans l'image [22].

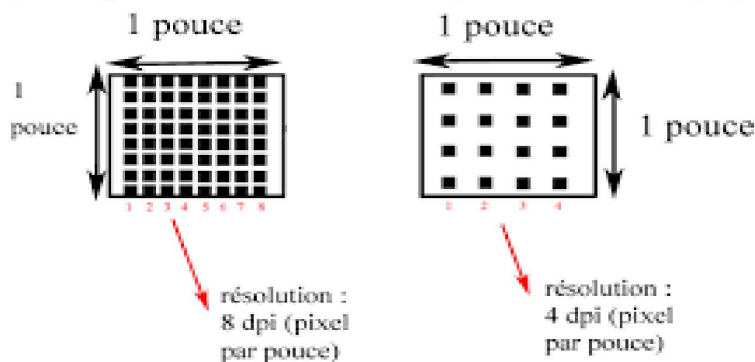


Figure 17. Exemple de résolution d'image

2.4. Bruit

Le bruit se réfère aux variations soudaines d'intensité lumineuse d'un pixel par rapport aux autres pixels d'une image numérique. Ce phénomène peut être causé par le capteur d'image, les circuits de numérisation d'un scanner ou d'un appareil photo numérique, le grain du film photographique, ou le bruit de fond inévitable d'un détecteur de lumière idéal. Le bruit est un effet indésirable de la capture d'image qui peut masquer les informations importantes et altérer la qualité de l'image.

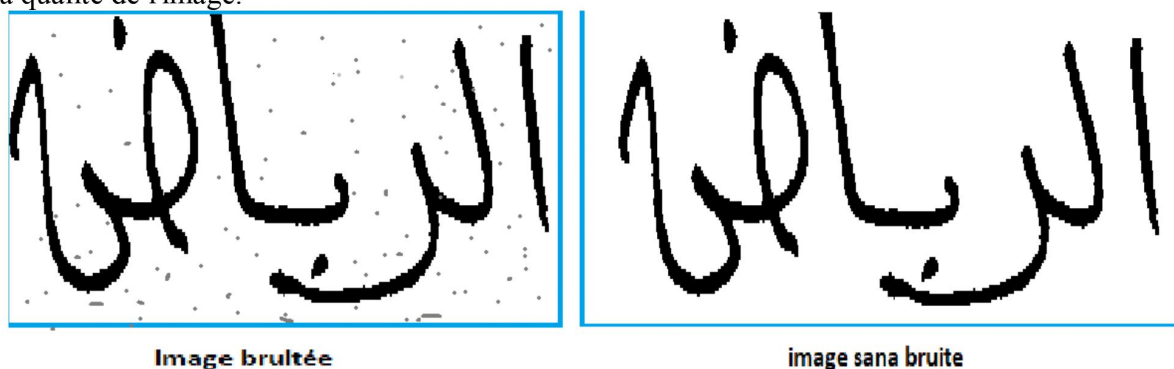


Figure 18. Exemple de bruit dans l'image

2.5. Histogramme

L'histogramme d'une image numérique est une représentation graphique de la distribution des niveaux de gris ou des couleurs de l'image. Il indique la fréquence d'apparition de chaque niveau de gris ou couleur dans l'image et peut être utilisé pour corriger des différences de luminosité, comparer des images prises dans des conditions différentes ou mesurer certaines propriétés de l'image. L'histogramme permet de visualiser rapidement la répartition des niveaux de gris ou de couleur dans l'image et d'identifier les zones où l'image est trop claire ou trop foncée [22].

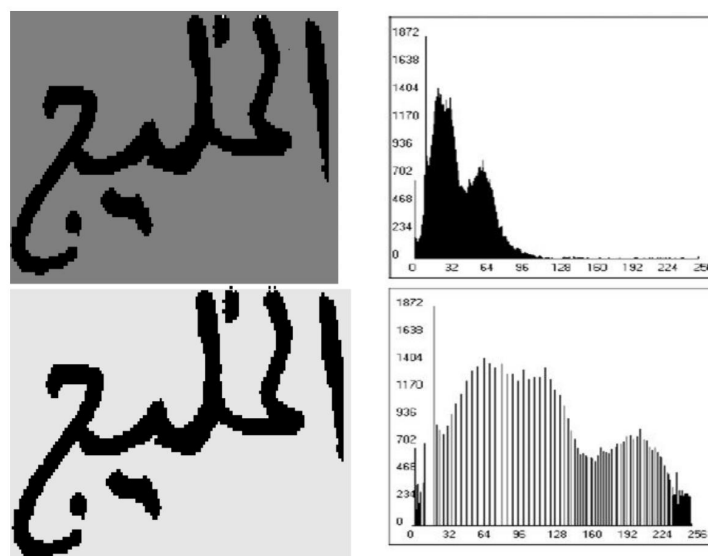


Figure 19. Exemple d'histogramme d'intensités de gris dans une image

2.6. Texture

Une texture est une région dans une image numérique qui a des caractéristiques homogènes. Ces caractéristiques sont par exemple un motif basique qui se répète. La texture est composée de Texel, l'équivalent des pixels.

3. Extraction des caractéristiques

En traitement d'image, les caractéristiques d'une image sont des parties ou des propriétés mesurables qui contiennent des informations importantes. Dans le cadre de la reconnaissance de l'écriture manuscrite, l'extraction de caractéristiques est une technique qui consiste à réduire la dimensionnalité des données d'origine en obtenant les informations les plus pertinentes et à les représenter dans un espace de dimensionnalité inférieure. Lorsque les données d'entrée sont volumineuses et redondantes, elles sont transformées en un ensemble réduit de vecteurs de caractéristiques.

L'objectif principal de l'extraction de caractéristiques est de trouver un ensemble d'éléments de caractéristiques qui permettent une reconnaissance précise avec un minimum d'éléments. Cette technique permet de générer un ensemble de caractéristiques similaires pour une variété d'instances du même symbole.

3.1. Sélection des caractéristiques

La sélection des caractéristiques les plus importantes est une étape cruciale dans le processus d'extraction de caractéristiques, en particulier pour les systèmes de reconnaissance de l'écriture manuscrite. Cette étape est nécessaire pour trouver tous les sous-ensembles de caractéristiques possibles à partir de l'ensemble initial, ce qui peut prendre beaucoup de temps. Les caractéristiques sélectionnées doivent être significatives pour différencier les différentes classes de caractères à reconnaître, tout en maintenant un nombre de dimensions limité pour éviter le phénomène de la dimensionnalité. De plus, les caractéristiques doivent être stables par rapport à la translation, la rotation et le changement d'échelle, et doivent être indépendantes de la distorsion et du bruit. Cependant, cette phase d'extraction de caractéristiques peut entraîner une perte d'information, il est donc important de prendre en compte la qualité et la quantité des données. En réduisant le nombre de caractéristiques, on peut améliorer la visualisation et la compréhension des informations, réduire les temps d'apprentissage et de classification des systèmes de reconnaissance, améliorer les performances en classification et minimiser la taille

des bases d'apprentissage.

3.2. Types des caractéristiques

Il est possible de distinguer cinq catégories principales de caractéristiques qui sont : [26]

- Les caractéristiques topologiques.
- Les caractéristiques statistiques.
- Les caractéristiques structurelles.
- Les caractéristiques globales ou locales.
- La superposition des modèles et la corrélation.

3.2.1. Caractéristiques topologiques ou métrique

Les caractéristiques topologiques, ou caractéristiques métriques, sont des mesures géométriques de l'objet étudié qui permettent de décrire sa forme globale. Ces mesures peuvent être des propriétés simples comme le nombre de points, de courbes ou d'angles présents dans l'objet, ou des mesures plus complexes comme la longueur des contours ou la surface de l'objet. Ces caractéristiques sont souvent utilisées pour distinguer des classes de formes différentes, par exemple pour reconnaître des lettres manuscrites [27].

Les caractéristiques topologiques ont l'avantage d'être relativement simples à calculer, car elles ne nécessitent pas une analyse détaillée de l'objet. De plus, elles permettent de décrire la forme globale de l'objet, ce qui peut être suffisant pour certaines applications de reconnaissance de formes.

3.2.2. Caractéristiques structurelles

Les caractéristiques structurelles sont dérivées du squelette ou du contour de l'image ou de la forme, ce qui permet de donner des caractéristiques globales et locales. Contrairement aux caractéristiques topologiques, qui portent sur la connectivité des pixels, les caractéristiques structurelles portent sur les formes et les structures des objets.

Parmi les caractéristiques structurelles courantes, on peut citer les lignes et les anses avec différentes directions et tailles, les points terminaux, la hauteur et la largeur de la primitive, les pentes et les arcs, les points d'intersections, les boucles, etc.

3.2.3. Caractéristiques statistiques

Les caractéristiques statistiques sont dérivées de la distribution de l'intensité des pixels dans l'image. Elles permettent d'obtenir des informations sur la texture de l'image, notamment sa rugosité et sa densité. Parmi les caractéristiques statistiques, on peut citer :

- La moyenne : c'est la somme de toutes les valeurs de pixels divisée par le nombre total de pixels. Elle donne une idée de la luminosité globale de l'image.
- L'écart-type : c'est une mesure de la dispersion des valeurs de pixels par rapport à la moyenne. Elle permet de caractériser la variation de l'intensité des pixels dans l'image.
- L'asymétrie : c'est une mesure de l'asymétrie de la distribution des valeurs de pixels par rapport à la moyenne. Elle permet de caractériser la forme de la distribution de l'intensité des pixels.

- Le coefficient de variation : c'est le rapport entre l'écart-type et la moyenne. Il permet de caractériser la variabilité relative de l'intensité des pixels dans l'image. [28]

3.2.4. Caractéristiques globales ou locales

Les caractéristiques globales et locales peuvent être utilisées ensemble pour fournir une représentation plus complète de l'image. Les caractéristiques globales sont utiles pour la reconnaissance d'objets dans leur ensemble, tandis que les caractéristiques locales permettent de capturer les détails fins qui peuvent être importants pour la discrimination entre différents objets ou motifs.

Les caractéristiques globales et locales sont largement utilisées dans les domaines de la vision par ordinateur et de l'apprentissage automatique pour la reconnaissance d'objets, la segmentation d'images, la détection de contours, la reconnaissance de caractères, entre autres applications. [29]

3.2.5. Superposition des modèles et corrélation

La superposition des modèles et la corrélation sont des techniques de reconnaissance d'objets qui consistent à comparer un modèle d'objet à une image pour déterminer si l'objet est présent dans l'image. La corrélation est souvent utilisée pour mesurer la similarité entre le modèle et l'image, mais d'autres méthodes, telles que la méthode de moments invariants et la méthode de Fourier, sont également utilisées pour améliorer la robustesse de la reconnaissance aux variations d'échelle et de rotation.

4. Méthodes d'extractions des caractéristiques

En ce qui suit on va détailler le principe et le fonctionnement des méthodes d'extraction des caractéristiques et le classifieur utilisé dans notre système qui concerne seulement la reconnaissance des mots manuscrits.

4.1. LBP (Local Binary Pattern)

Les LBP sont intrinsèquement des caractéristiques de texture au début mais elles ont été utilisées dans un très large éventail d'applications en informatique dont beaucoup dépassent la reconnaissance de texture typique. Tâches comme :

- En 2004, Ahonen et al, utilisé avec succès LBP pour la reconnaissance faciale [37].
- En 2007, Zhao et al, a étendu l'opérateur comme une version 2D plus temps voxel de LBP, appelé VLBP, et les a utilisés avec succès pour la reconnaissance des gestes faciaux [38].
- Dans 2009, Whang et al. Fonctions LBP combinées avec HOG pour résoudre le problème des occlusions partielles dans le problème de détection humaine. Plus récemment, LBP a gagné l'attention de l'analyse d'image de document [39].
- En 2010 Luft et al, utilisé LBP sur les diacritiques de l'écriture arabe pour améliorer l'identification des écrivains [40].
- Du et al. Proposer LBP extraction sur le domaine des ondelettes pour l'identification de l'auteur [41].
- En 2013, Ferrer et al, utilisé LBP orienté sur différents bandes horizontales dans

une ligne de texte pour détecter le script [42].

Dans ce travail nous nous intéressons au descripteur LBP pour l'extraction des caractéristiques dans les mots manuscrits.

4.1.1. Principe de LBP

LBP fonctionne en principe par la division de l'image en cellules des niveaux de gris. Pour chaque pixel de la cellule, on doit comparer le pixel aux pixels voisins, si les valeurs du pixel central sont supérieures à la valeur du voisin on écrit 1 sinon 0. Il donne un nombre binaire à 8 chiffres. Après avoir obtenu ce chiffre, on le convertisse en nombre décimal correspondant. Puis on Calcule l'histogramme de chaque cellule, de la fréquence de chaque nombre, dont l'histogramme est un vecteur de caractéristiques à 256 dimensions. Enfin, la normalisation d'histogramme.

L'histogramme normalisé de chaque cellule donne le vecteur caractéristique de l'image en niveaux de gris donnée. Une extension de LBP est un motif uniforme, qui réduit la taille du vecteur d'entité. Dans un motif uniforme, la longueur de la fonction vecteur pour une seule cellule condensée de 256 à 59.

LBP utilise des voisins de 8 pixels avec 1 pixel central pour fonctionner comme valeur de seuil. La valeur de chaque pixel voisin sera comparée au pixel central, ensuite tous les pixels voisins seront combinés de gauche à droite on obtiendra des nombres binaires qui peuvent être convertis en décimaux aussi appelés LBP [43]. Comme dans l'exemple suivant :

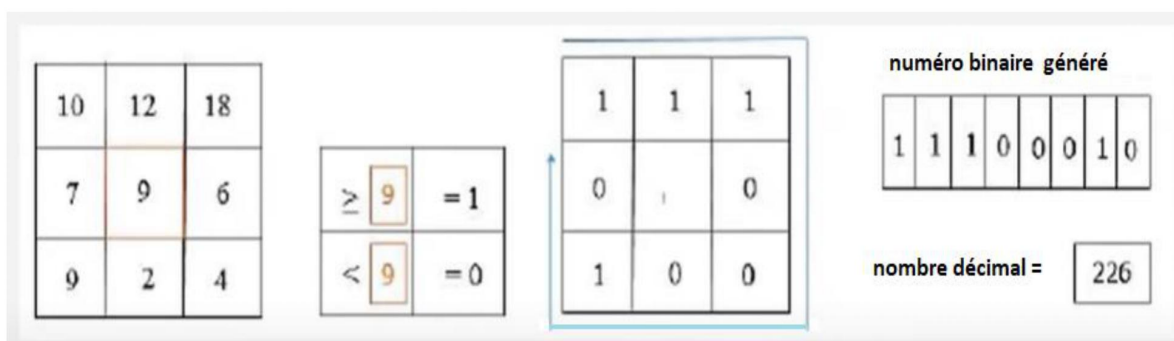


Figure 20. Exemple de Calcul de base du LBP

4.1.2. Méthodologie

LBP opère sur chaque pixel en seillant ses voisins pixels avec la valeur du pixel central lui-même. Il en résulte des valeurs binaires, 0 et 1, où 0 indique que la valeur d'échelle de gris du pixel voisin est inférieure à celle du pixel central, tandis que 1 indique le contraire. Il faut concaténer le flux binaire de tous les voisins est ensuite converti en décimal et est défini comme la valeur LBP du pixel central. Ce processus est expliqué avec les mathématiques suivantes expressions :

$$S(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

$$LBP = \sum_{p=0}^{p-1} s(g_p - g_c) 2^p$$

Où $s(x)$ est la fonction de seuil, et g_c et g_p représentent la valeur d'échelle de gris du pixel central et le p -ème voisin respectivement. Par conséquent, la valeur binaire du p -ème voisin est

multipliée par son poids correspondant 2^p . LBP est calculé en additionnant ces valeurs binaires pondérées. C'est illustré sur la figure 18 qui montre que le motif binaire 01110101 est converti en $LBP = 1 + 4 + 16 + 32 + 64 = 117$.

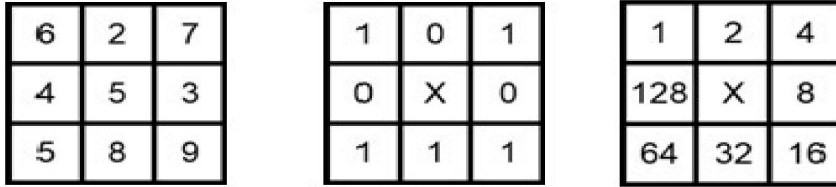


Figure 21. Exemple Illustration du calcul du modèle binaire local (LBP) (a) exemple image, (b) patron binaire par seuillage et (c) matrice de poids

Le LBP de base a une limitation qui est la taille de 3×3 pixels voisins afin qu'il n puisse pas capturer le domaine de texture propriétés à plus grande échelle donc pour surmonter la texture à grande échelle, LBP est développé en utilisant les plus grand locaux pixels voisins [43].

La première étape qui peut être effectuée en développer une taille de pixel locale consiste à définir la texture T dans l'environnement voisin de l'image en niveaux de gris $P + 1$ ($P > 0$) comme dans la formule suivante :

$$T = t(rc, r_0, \dots, r_{P-1})$$

Où r_c est la valeur centrale des pixels locaux voisins. r_p ($P=0, \dots, P-1$) est la valeur en pixels de l'échelle de gris P avec espace ou une distance égale au rayon du cercle R ($R > 0$) formera un voisin symétrique circulaire voisin.

La figure 22: illustrations de pixels locaux voisins où les valeurs P et R varient (P, R), montre la valeur du pixel voisin P avec un rayon de tir de R valeur. La valeur d'un pixel voisin incorrect tombant sur le pixel sera calculée en utilisant une interpolation bilinéaire.

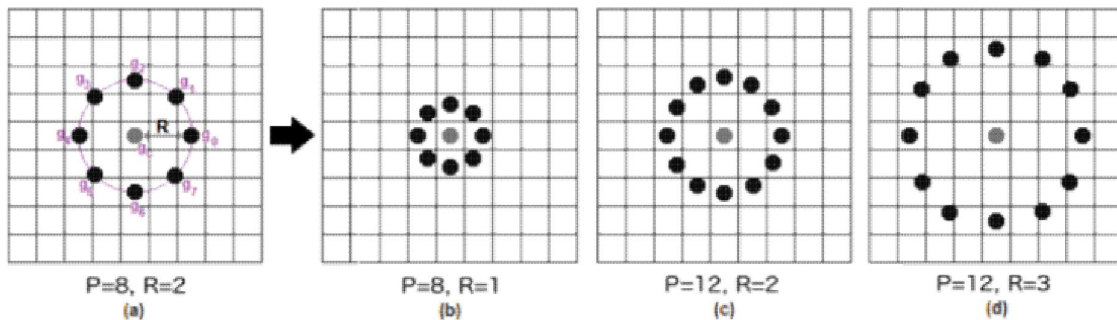


Figure 22. Illustrations de pixels locaux voisins

4.2. CLBP (Compound Local Binary Pattern)

Local binary patterns sont les seuls à rendre compte de la structure locale de l'image, ignorant la différence de taille entre le pixel central et ses pixels voisins. Selon Guo et al. [45], LBP a fréquemment tendance à produire des codes incohérents car il prend simplement en compte la différence entre deux valeurs de gris, comme le montre la figure 23. La figure affiche le code binaire produit par l'opérateur LBP pour le pixel central avec la valeur d'intensité (49). Dans ce cas, le code LPB généré représente une fausse tache sombre. La figure 5 fournit des exemples de schéma binaire incohérent dans le processus de codage LBP.

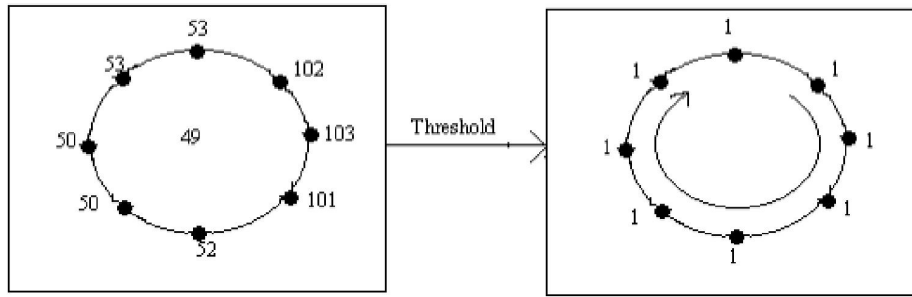


Figure 23. LBP.

Guo et al. [45] ont développé un modèle LBP complet nommé CLBP pour résoudre ce problème. Les variances locales d'amplitude et les informations concernant le signe de chaque pattern ont été intégrées aux données centrales en niveaux de gris. Pour capturer le signe et l'amplitude du changement, deux bits sont utilisés. L'équation 2 fournit un résumé du calcul

$$s_p = s(i_p - i_c), \quad m_p = |i_p - i_c| \tag{2}$$

Où i_p est le niveau d'intensité du pixel le plus proche, i_c est le niveau d'intensité du pixel central, s_p est la différence de signe entre les niveaux d'intensité central et voisin, et m_p est la différence d'amplitude. De plus, CLBP-Sign (CLBP_S) et CLBP-Magnitude (CLBP_M) sont calculés à l'aide de s_p et m_p . Les équations 3 et 4 sont utilisées, respectivement, pour exprimer mathématiquement le signe CLBP et la magnitude CLBP

$$CLBP_S_{p,R} = \sum_{p=0}^{P-1} 2^p s(i_p - i_c), \quad \begin{cases} s_p = 0, i_p < i_c \\ s_p = 1, i_p \geq i_c \end{cases} \tag{3}$$

$$CLBP_M_{p,R} = \sum_{p=0}^{P-1} 2^p t(m_p, c), \quad \begin{cases} t(m_p, c) = 1, |i_p - i_c| \geq c \\ t(m_p, c) = 0, |i_p - i_c| < c \end{cases} \tag{4}$$

Où i_p représente le niveau d'intensité du pixel voisin, i_c représente le niveau d'intensité du pixel central, R est le rayon du voisinage et P est la valeur du pixel central. Guo et al. [36] ont en outre introduit un nouvel opérateur appelé CLBP Center (CLBP_C) qui utilise le niveau de gris de chaque pattern (Equation 5).

$$CLBP_C_{p,R} = t(i_c, c_i) \tag{5}$$

Les résultats de la figure 6 du calcul CLPB. Le CLPB divise les Différences Locales (voir Figure 24(b)) pattern d'un block 3x3 (voir Figure 6(a)) la Différences locale, le Composant Signe (voir Figure 24(c)) et le Composant Magnitude (voir Figure 24(d)).

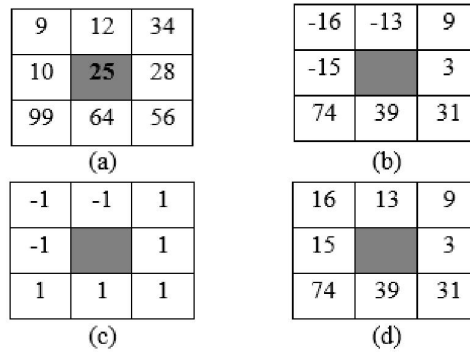


Figure 24. Calcul du CLPB (a): pattern d'un block 3x3 (b): la Différences locale (c): le Composant Signe (d): le Composant Magnitude.

4.3. OBIF (Oriented Basic Image Feature)

Dans notre étude, nous avons choisi d'utiliser une combinaison de descripteurs oBIFS et de caractéristiques de fond. Les oBIFS permettent de capturer des informations synthétiques sur les mots, tandis que les caractéristiques de fond exploitent les propriétés géométriques et topologiques des mots pour une représentation discriminante. Chacune de ces caractéristiques est discutée dans les sous-sections suivantes, suivies de leurs détails de calcul pour notre problème spécifique.

4.3.1. Oriented Basic Image Features (oBIFs)

Les Oriented Basic Image Features (oBIFs) [33,34] sont un descripteur de texture puissant qui a été utilisé avec succès dans des problèmes tels que la reconnaissance de mots [35], la classification de textures [30]. Étant donné que chaque mot manuscrit peut être considéré comme une texture différente, les oBIFs sont potentiellement efficaces dans le problème de reconnaissance de mots. Les oBIFs sont une extension des Basic Image Features (BIFs) [33,34] et impliquent l'intégration de l'orientation locale avec des informations de symétrie locale. Chaque emplacement dans l'image est classé dans l'une des sept classes de symétrie locales. Celles-ci incluent des stries foncées sur fond clair, des stries claires sur fond foncé, une rotation sombre, une rotation claire, une pente, une selle ou une surface plane. La classification est basée sur la réponse d'une banque de filtres dérivés gaussiens (DoG) de taille déterminée par le paramètre d'échelle σ . Le paramètre complémentaire ε détermine si le site sera classé comme étant plat. Pour combiner les informations d'orientation locale avec les informations de symétrie locale, en fonction de la classe de symétrie du site, des directions possibles sont déterminées. Si l'emplacement est attribué à des rotations sombres, des rotations claires ou des surfaces planes, aucune direction n'est définie. Pour les classes de stries foncées sur fond clair, de stries claires sur fond foncé et de type selle, des directions possibles peuvent être attribuées, tandis que la classe de pente a un total de $2n$ directions possibles. Cela donne un vecteur d'attributs de dimension $5n + 3$. Dans notre implémentation des oBIFs sur des numéros de mots manuscrits isolés, nous fixons le coefficient de quantification de la direction $n = 4$. Cela donne un total de 23 entrées dans le dictionnaire des oBIFs. Enfin, l'histogramme (normatif) des oBIFs calculé à partir de l'image du mot est utilisé comme descripteur. La figure 25 montre un exemple de numéro encodé à l'aide des oBIFs et l'histogramme correspondant.

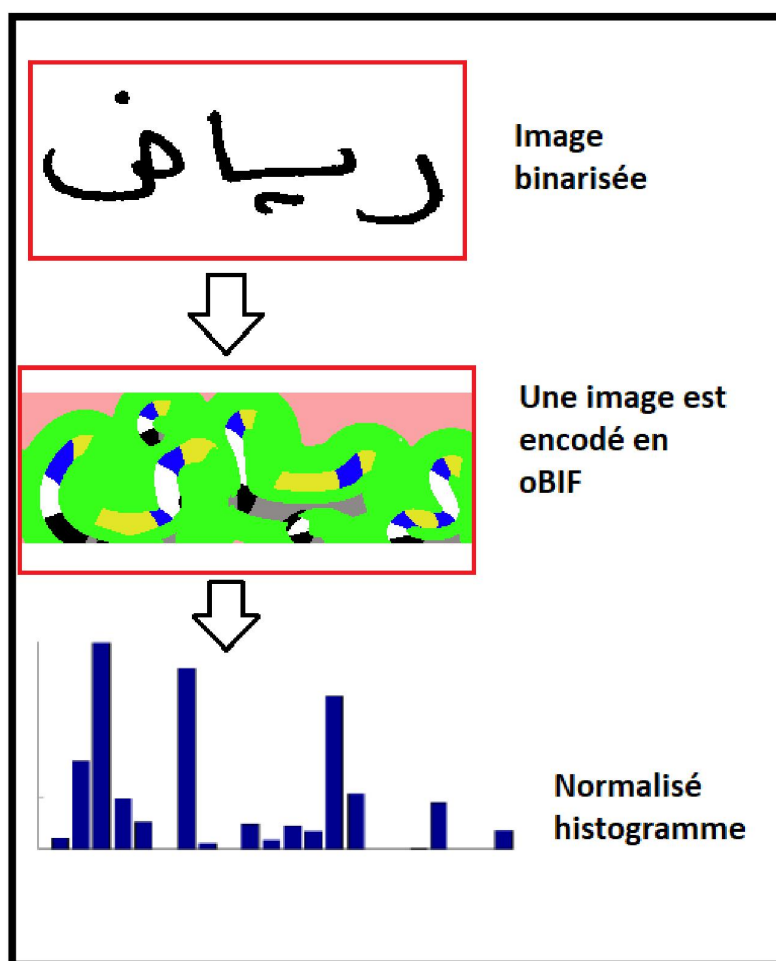


Figure 25. Exemple de calcul des OBIF d'images manuscrites $\sigma = 8$ and $\epsilon = 0.01$.

4.3.2. Caractéristiques de fond

Les caractéristiques de fond (BF) [31, 32] sont basées sur les informations de concavité et visent à capturer les propriétés topologiques d'un mot. Ces caractéristiques représentent le nombre de pixels blancs ayant une configuration de concavité spécifique. Le label de chaque configuration de concavité est choisi en fonction du code de Freeman avec quatre directions pour chaque pixel blanc. Chaque direction est explorée jusqu'à ce qu'un pixel noir ou la limite du mot soit atteint. En plus des neuf configurations de concavité standard, nous considérons également cinq configurations supplémentaires pour modéliser plus précisément la présence de boucles dans le mot. Ces configurations sont illustrées dans la figure 26.

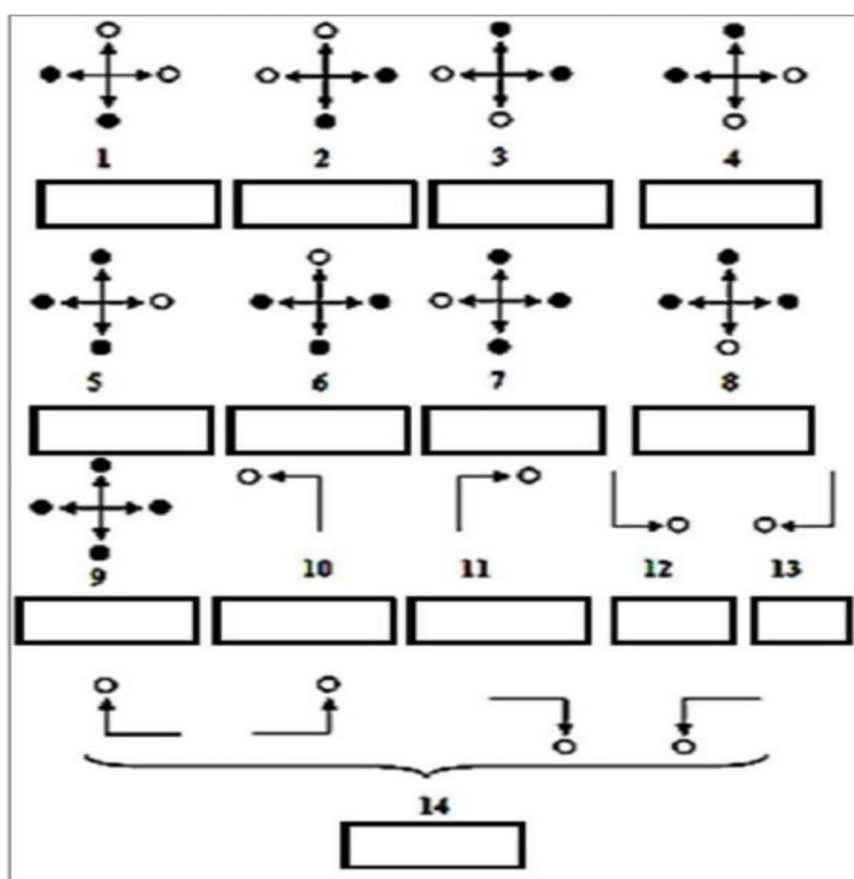


Figure 26 .Configurations de concavité

Les étiquettes de concavité des pixels d'arrière-plan pour un échantillon de mot sont illustrées dans la Figure 27, où chaque couleur représente une configuration de concavité différente. L'histogramme à 14 bin (normalisé) des étiquettes de concavité est calculé à partir d'un mot manuscrit et est utilisé comme caractéristique.



Figure 27. Étiquettes de concavité pour l'image d'un mot manuscrit.

4.4. Histogrammes oBIF column

Une autre caractéristique utilisée dans notre tâche de reconnaissance des manuscrites arabes est les fonctionnalités d'image de base orientées (oBIF) qui capturent les informations texturales dans une écriture manuscrite pour une représentation discriminatoire. Ces descripteurs ont été utiles avec succès pour des problèmes, notamment : la classification des textures [54], la classification des sexes [55], le document d'identification de l'écrivain moderne [56], la reconnaissance des chiffres [56] et le document manuscrit historique [57]. Les caractéristiques d'image de base orientées descripteurs basés sur la texture (oBIF) sont une extension des

caractéristiques d'image de base (BIF) [57], qui peuvent être inclinées, plates, en rotation sombre, en rotation claire, en selle, en ligne claire sur fond sombre ou ligne sombre sur la lumière. Chaque emplacement de l'image est classé dans l'une des sept classes de symétrie locale en fonction du type de symétrie locale, qui peut être pente, plat, rotation sombre, rotation claire, selle, ligne claire sur ligne sombre ou ligne sombre sur lumière.

Cette caractéristique calculée à l'aide d'un ensemble de six dérivées de filtres gaussiens (jusqu'au second ordre) déterminées par un paramètre d'échelle ε , est très efficace pour détecter des classes de symétrie locales. Le paramètre σ détermine si un emplacement doit être classé comme plat. La dimension des vecteurs de caractéristique oBIF est de $5n + 3$. Dans notre étude, les orientations calculées en $n = 4$ niveaux donnent 23 entrées dans le descripteur oBIFs. Nous combinons oBIF à deux échelles distinctes pour fournir les caractéristiques de la column oBIF en ignorant tout plat de type symétrie pour améliorer les performances du descripteur oBIF. En conséquence, le schéma d'histogramme de column oBIF produit 484 entrées jusqu'à $5(n + 2)^2$. Les caractéristiques de l'histogramme de column oBIF sont déterminées par deux paramètres : le paramètre ε , qui est défini sur l'une des trois petites valeurs de $\varepsilon \in \{0.1, 0.01, 0.001\}$, et le paramètre d'échelle où $\sigma \in \{1, 2, 4, 8, 16\}$. Enfin, le vecteur de caractéristiques généré est normalisé.

La figure 29 montre les différentes étapes du schéma d'histogramme de column oBIF. Le vecteur de caractéristiques final de notre système de reconnaissance des manuscrites arabes est formé en calculant les histogrammes de column oBIFs extraits de chaque image d'écriture manuscrite. Ce vecteur a été normalisé pour que la moyenne et la variance unitaire soient nulles. Différentes configurations sont formées en modifiant les deux paramètres des histogrammes de column oBIFs σ et ε .

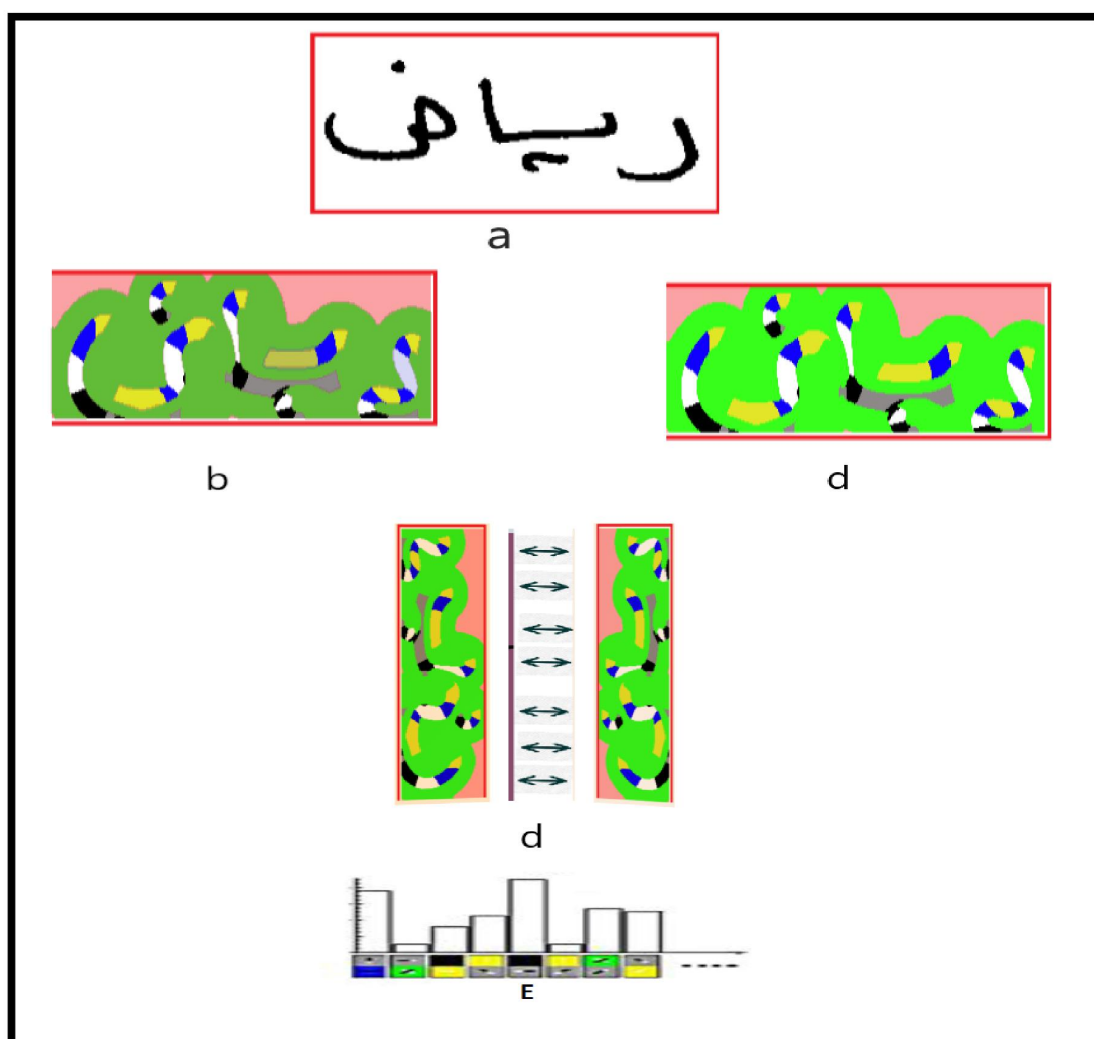


Figure 28. Les étapes de l'histogramme des column oBIF (A) Image originale de l'écriture manuscrite (B) Calcul des oBIF pour les paramètres d'échelle $\sigma = 8$ et $\sigma = 4$ (C) Les oBIF à deux échelles sont croisés pour former des column à chaque emplacement (D) L'his 'histogramme est calculé avec c non plat.

5. Classification en SVM

Une machine à vecteurs de support (SVM), Elles ont été conceptualisées dans les années 1990 à partir d'une théorie d'apprentissage statistique développée par les informaticiens russes Vladimir Vapnik et Alexey Chervonenkis : la théorie de Vapnik-Chervonenkis également connue sous le nom de "classifieur à marge maximale", est sans aucun doute l'une des formes les plus réussies et populaires de modèles d'apprentissage automatique supervisés. Elle a connu un grand succès dans le domaine de la reconnaissance d'écriture manuscrite. En apprenant à partir d'exemples fournis - les données d'apprentissage - le modèle trouve une fonction qui attribue les données d'entrée à la sortie correcte. La sortie de nouvelles données peut ensuite être prédite en appliquant la fonction extraite. Les SVM sont souvent utilisées pour des problèmes de classification, où la sortie correcte est la classe à laquelle les données appartiennent, ainsi que pour la régression, la détection des valeurs aberrantes et les problèmes de regroupement [46]. Le modèle fonctionne en créant un hyperplan qui sépare les points de données d'une classe de ceux de l'autre classe, avec une marge maximale. La marge est la largeur maximale de la bande parallèle à l'hyperplan qui ne contient aucun point de données. Les vecteurs de support, qui

donnent leur nom au modèle, sont les points de données les plus proches de l'hyperplan et déterminent ainsi la marge.

Les SVM sont généralement basées sur la théorie de l'apprentissage statistique et suivent le principe de la minimisation du risque. Elles reposent sur les concepts de fonction de noyau et de marge maximale. La fonction de noyau permet de transformer les données d'entrée dans un espace de dimensions supérieures, où une séparation linéaire peut être possible. En maximisant la marge, les SVM visent à trouver la frontière de décision optimale qui minimise l'erreur de généralisation [47].

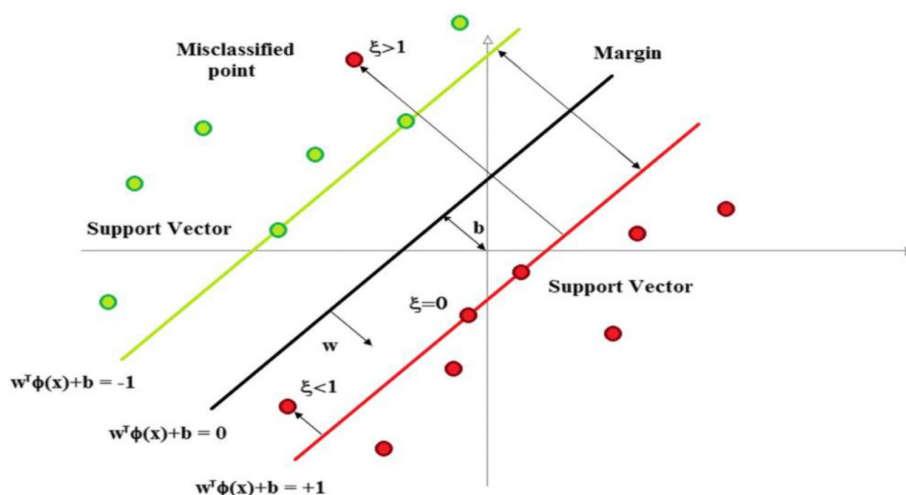


Figure 29. Exemple de classification en SVM [48].

5.1. Apprentissage statistique et SVM

L'apprentissage statistique joue un rôle fondamental dans les machines à vecteurs de support (SVM) en permettant d'obtenir des résultats à partir d'un ensemble de couples entrée/sortie. L'apprentissage peut être divisé en deux types : supervisé et non supervisé. Dans le cas des SVM, nous nous concentrons sur l'apprentissage supervisé.

L'objectif principal de l'apprentissage supervisé est de développer des méthodes capables de traiter et d'analyser les données d'apprentissage afin de déduire des règles qui permettent de prédire le comportement ou d'identifier la classe à laquelle une donnée appartient. Ces méthodes prennent souvent la forme d'une fonction qui correspond aux exemples d'apprentissage et qui peut être utilisée pour prédire les sorties pour de nouvelles entrées non traitées. Les entrées représentent généralement des descriptions d'objets (données), et la sortie est la classe à laquelle ces objets appartiennent (étiquettes) [49].

dans le contexte des SVM, l'apprentissage statistique est utilisé pour entraîner le modèle en fournissant des exemples étiquetés, où les entrées sont les données d'objets et les sorties sont les classes correspondantes. À partir de ces exemples d'apprentissage, le modèle SVM apprend à établir des règles et des relations afin de prédire les classes pour de nouvelles données d'entrée.

5.2. Objectif de l'apprentissage statistique

Pour mieux comprendre le rôle de la théorie d'apprentissage statistique dans SVM coté mathématique, on prend comme exemple de la classification Bi-classe (de deux classes) qui se base sur l'estimation d'une fonction d'apprentissage $f: x \rightarrow \{+1, -1\}$

L'ensemble d'apprentissage peut prendre les couples (x_i) , selon une distribution de probabilité $P(x, y)$ inconnue, tel que :

$$(x_i, y_i) \in X \times Y \text{ où } i=1, \dots, N \text{ et } y_i \in \{+1, -1\}$$

De sorte à ce que f classe correctement des exemples inconnus (x_i) . Prenant le cas où on peut affecter x_k à la classe (+1) si $(x_k) \geq 0$, sinon à la classe (-1).

L'ensemble des exemples inconnus suivent la même distribution $P(x, y)$ comme dans l'ensemble d'apprentissage. Pour obtenir la fonction optimale f en doit minimiser le risque par cette fonction:

$$R[f] = \int L[f(x), y] dP(x, y).$$

Où L est une fonction de coût, par exemple :

$$L[f(x), y] = (f(x) - y)^2.$$

5.3. Le sur-apprentissage et le sous-apprentissage

Le sur-apprentissage (overfitting) et le sous-apprentissage (underfitting) sont deux problèmes courants dans l'apprentissage statistique, y compris lors de l'utilisation des machines à vecteurs de support (SVM). Voici une explication de ces deux concepts

5.3.1. Sur-apprentissage

Le sur-apprentissage se produit lorsque le modèle appris est trop complexe et s'adapte de manière excessive aux données d'apprentissage spécifiques, au point de capturer également le bruit ou les variations aléatoires présentes dans les données d'apprentissage. En conséquence, le modèle ne généralise pas bien et ne parvient pas à produire des prédictions précises sur de nouvelles données. Le sur-apprentissage se manifeste par une très bonne performance sur les données d'apprentissage, mais une performance médiocre sur les données de test ou de validation.

Les causes courantes de sur-apprentissage sont un modèle trop complexe par rapport à la quantité de données d'apprentissage disponibles, une sélection excessive de caractéristiques ou une mauvaise régularisation des paramètres du modèle. Pour éviter le sur-apprentissage, il est important d'utiliser des techniques de régularisation telles que la pénalisation des paramètres, le contrôle de la complexité du modèle et l'utilisation de données de validation pour ajuster les hyperparamètres [44]

5.3.2. Sous-apprentissage

Le sous-apprentissage se produit lorsque le modèle n'est pas suffisamment complexe pour capturer les structures et les relations présentes dans les données d'apprentissage. En conséquence, le modèle ne parvient pas à saisir les modèles sous-jacents et produit des prédictions simplistes qui ne sont pas précises. Le sous-apprentissage se manifeste par une mauvaise performance à la fois sur les données d'apprentissage et sur les données de test ou de validation.

Les causes courantes de sous-apprentissage sont un modèle trop simple ou une quantité insuffisante de données d'apprentissage pour estimer correctement les paramètres du modèle. Pour éviter le sous-apprentissage, il est important d'utiliser un modèle plus complexe ou d'explorer des techniques d'enrichissement des données telles que l'ajout de nouvelles caractéristiques ou l'utilisation de techniques de prétraitement appropriées [44].

5.4. Principe de SVM

Le principe des machines à vecteurs de support (SVM) repose sur la création d'un hyperplan dans un espace de dimensions supérieures, qui sépare les données en différentes classes. L'objectif est de trouver l'hyperplan qui maximise la marge entre les exemples de différentes classes, c'est-à-dire la distance entre l'hyperplan et les exemples les plus proches de chaque classe, appelés vecteurs supports [50].

Voici les étapes principales du principe de SVM :

5.4.1. Préparation des données :

Les données d'apprentissage doivent être prétraitées et représentées sous forme de vecteurs d'entrée. Chaque vecteur d'entrée est associé à une classe ou une étiquette correspondante.

5.4.2. Sélection du noyau :

Un noyau est une fonction qui calcule le produit scalaire entre deux vecteurs dans un espace de dimensions supérieures. Le choix du noyau détermine la manière dont les données sont transformées pour être séparées par l'hyperplan. Les noyaux couramment utilisés sont le linéaire, le polynomial et le gaussien.

5.4.3. Détermination de l'hyperplan optimal :

L'objectif est de trouver l'hyperplan qui maximise la marge entre les classes. Cela revient à résoudre un problème d'optimisation qui consiste à minimiser une fonction coût tout en respectant des contraintes liées aux vecteurs supports [50].

5.4.4. Classification des nouvelles données

Une fois que l'hyperplan optimal est trouvé, il peut être utilisé pour classer de nouvelles données en les projetant dans l'espace de dimensions supérieures et en déterminant de quel côté de l'hyperplan elles se trouvent.

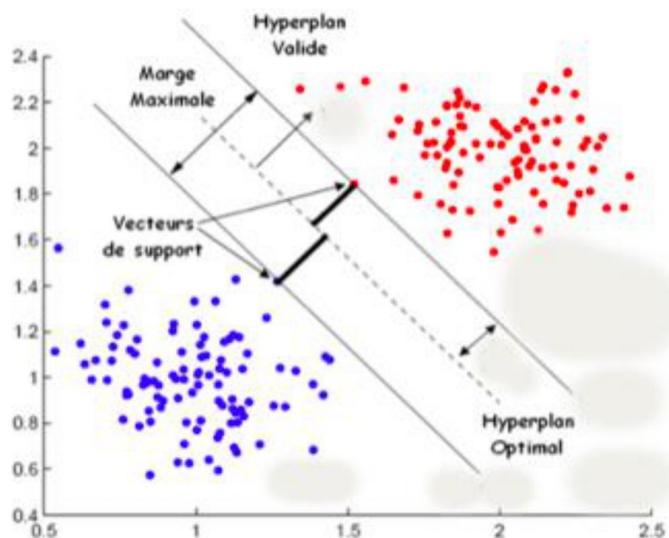


Figure 30. Les différents composants de base d'une SVM [50].

5.5. Types de SVM

Les machines à vecteurs de support (SVM) peuvent être largement classées en deux types :

5.5.1. SVM linéaire

Dans ce type de SVM, les données sont linéairement séparables et la frontière de décision est un hyperplan dans l'espace d'entrée. L'objectif est de trouver l'hyperplan optimal qui maximise la marge entre les classes. Les SVM linéaires sont efficaces et fonctionnent bien lorsque les classes peuvent être séparées par une ligne droite ou un hyperplan.

5.5.2. SVM non linéaire

Dans ce type de SVM, les données ne sont pas linéairement séparables dans l'espace d'entrée. Pour traiter des données non linéaires, une fonction de transformation est appliquée pour les projeter dans un espace de dimension supérieure où elles deviennent linéairement séparables. Cela est réalisé en utilisant une fonction de noyau qui mesure la similarité entre les paires de points de données dans l'espace d'entrée. Les fonctions de noyau les plus couramment utilisées sont le noyau polynomial, le noyau gaussien (RBF) et le noyau sigmoïde. Les SVM non linéaires sont plus flexibles et peuvent capturer des frontières de décision complexes [51].

5.6. Fonctions noyaux (Kernel)

Les fonctions noyaux (kernels) sont des fonctions mathématiques utilisées dans les machines à vecteurs de support (SVM) pour effectuer des transformations non linéaires des données d'entrée. Elles permettent aux SVM de travailler efficacement dans des espaces de caractéristiques de dimension supérieure sans effectuer explicitement les calculs dans cet espace [52].

Voici quelques exemples courants de fonctions noyaux utilisées dans les SVM :

Noyau linéaire : C'est la forme la plus simple de noyau qui correspond à une classification linéaire dans l'espace d'entrée d'origine.

Noyau polynomial : Cette fonction noyau permet de créer des frontières de décision polynomiales dans l'espace d'entrée.

Noyau gaussien (RBF) : Également connu sous le nom de noyau à base radiale, il utilise une fonction de similarité gaussienne pour capturer des relations non linéaires complexes entre les données.

Noyau sigmoïde : Ce noyau est utilisé pour modéliser des relations non linéaires en utilisant une fonction sigmoïde.

5.7. Avantages et inconvénients des SVM

Quelques avantages et inconvénients associés à l'utilisation des SVM :

- **Avantages des SVM**
 - Efficacité pour les données de grande dimension : Les SVM fonctionnent bien même lorsque le nombre de caractéristiques (dimensions) des données est élevé, ce qui les rend adaptés à l'analyse de données complexes.
 - Capacité à gérer les données non linéaires : Les SVM peuvent utiliser des fonctions noyaux pour transformer les données d'entrée dans des espaces de caractéristiques de dimension supérieure, leur permettant de trouver des frontières de décision non linéaires.
 - Gestion des données peu structurées : Les SVM peuvent traiter des données non structurées et bruitées, grâce à leur capacité à trouver un hyperplan de séparation robuste.

- Gestion des problèmes de sur-apprentissage : Les SVM ont une capacité naturelle à gérer le sur-apprentissage (overfitting), en utilisant la marge maximale comme critère de séparation, ce qui permet de généraliser les résultats sur de nouvelles données.
 - Robustesse aux valeurs aberrantes : Les SVM sont relativement robustes aux valeurs aberrantes (outliers) dans les données, car ils se concentrent sur les vecteurs supports les plus proches de la frontière de décision.
- **Inconvénients des SVM**
 - Sensibilité aux paramètres : Les SVM ont plusieurs paramètres qui doivent être réglés avec soin, tels que le choix de la fonction noyau et des paramètres de régularisation. Un mauvais réglage de ces paramètres peut entraîner des performances médiocres.
 - Complexité de l'interprétation : Les SVM ne fournissent pas directement des explications ou des informations sur les caractéristiques importantes des données. Il peut être difficile d'interpréter les résultats de manière intuitive.
 - Temps de calcul élevé : L'entraînement d'un SVM peut être coûteux en termes de temps de calcul, en particulier pour les grands ensembles de données et les problèmes non linéaires.
 - Sensibilité aux données déséquilibrées : Les SVM peuvent être sensibles aux déséquilibres de classes dans les données, ce qui peut entraîner des biais vers la classe majoritaire

6. Conclusion

Dans ce chapitre, nous nous concentrons sur la conception et la mise en œuvre de notre système. Nous commençons par discuter des caractéristiques de l'image numérique, qui sont essentielles pour notre approche. Ensuite, nous abordons les méthodes d'extraction des caractéristiques que nous utilisons, à savoir LBP (Local Binary Patterns), CLBP (Compound Local Binary Patterns), OBIF (Oriented Basic Image Feature) et OBIF column. Ces méthodes nous permettent de capturer différentes informations sur la texture et la structure de l'image.

En outre, nous examinons la classification des données en utilisant une méthode appelée SVM (Support Vector Machine). Le SVM est un algorithme d'apprentissage supervisé largement utilisé pour la classification. Nous expliquons comment nous utilisons le SVM pour classer les images en fonction des caractéristiques extraites.

Ce chapitre nous permet de présenter en détail les différentes étapes de notre système, de la préparation des données à l'extraction des caractéristiques et à la classification.

Chapitre 3

Etat de l'art

1. Introduction

La reconnaissance de l'écriture arabe manuscrite est un domaine de recherche actif et en constante évolution. Au cours des dernières années, de nombreux travaux ont été réalisés dans ce domaine, visant à améliorer les techniques de reconnaissance et à obtenir des résultats plus précis.

Ce chapitre a pour objectif de présenter un état de l'art des travaux existants dans la reconnaissance des mots arabes manuscrits. Il vise à recenser et à comparer les différentes approches et méthodes utilisées, en se basant sur les résultats obtenus par chaque étude.

La comparaison entre les travaux se fera en fonction de divers critères tels que le taux de reconnaissance, la taille de la base de données utilisée, les caractéristiques extraites, les algorithmes de classification employés, etc.

2. Description de différents travaux ultérieurs

Dans ce titre, on explique 11 des dernières études sur la reconnaissance des mots arabes écrits à la main, qui sont réalisées sur la base de données IFN/ENIT.

2.1. La méthode proposée par M. Pechwitz et V. Maergner [58].

Ce travail présente un système de reconnaissance basé sur les modèles de Markov cachés (HMM) en une dimension semi-continue. Pour chaque image de mot binaire, des paramètres de normalisation ont été estimés. Tout d'abord, la hauteur, la largeur et l'inclinaison de la ligne de base ont été normalisées. Ensuite, les caractéristiques ont été regroupées en utilisant une fenêtre glissante. Les expériences ont été menées sur quatre ensembles distincts (a, b, c, d) de la base de données IFN / ENIT, qui contient 26 459 mots arabes manuscrits. Les ensembles (a, b, c) ont été utilisés pour l'apprentissage et l'ensemble d a été utilisé pour les tests. Le système a obtenu un taux de reconnaissance d'environ 89 %.

2.2. La méthode proposée par P.Dreuw et al [59] .

Cette étude présente un système de reconnaissance hors ligne de l'écriture arabe basé sur les modèles de Markov cachés (HMM). L'originalité de ce système réside dans sa modélisation des espaces blancs entre les caractères et les morceaux de mots arabes. Les modèles d'espaces blancs proposés ont permis d'améliorer considérablement les performances du système, surpassant ainsi les taux d'erreur précédemment rapportés.

Les expérimentations ont été menées sur la base de données IFN/ENIT, qui est composée de quatre ensembles d'apprentissage et d'un ensemble de test. Cette base de données contient plus de 32 000 mots arabes écrits à la main par plus de 1000 écrivains, avec un vocabulaire de 937 noms de villes tunisiennes. Les résultats obtenus ont démontré un taux de reconnaissance de 92,86 %, témoignant de l'efficacité du système développé.

2.3. La méthode proposée par A. Benouareth et al [60].

Dans cette étude, un système de reconnaissance hors-ligne des mots arabes sans segmentation est présenté. Le système utilise des modèles de Markov cachés semi-continus (SCHMMs) avec une durée d'état explicite pour améliorer la reconnaissance des mots et des lettres. Les auteurs proposent une nouvelle version de l'algorithme Viterbi qui tient compte de cette durée d'état explicite.

L'étude compare l'utilisation de trois distributions différentes (Gamma, Gauss et Poisson) pour la modélisation de la durée d'état explicite. De plus, le système utilise une approche de fenêtre glissante basée sur l'histogramme de projection verticale du mot pour extraire de nouvelles caractéristiques statistiques et structurelles, ce qui contribue à améliorer le taux de

reconnaissance des mots.

Les résultats obtenus sont très prometteurs, avec un taux de reconnaissance de 90,20 % (dans le top 1) sur l'ensemble des 26 459 mots arabes de la base de données IFN/ENIT. Ces résultats démontrent l'efficacité du système proposé dans la reconnaissance des mots arabes hors-ligne.

2.4. La méthode proposée par R.A. Mohamad et al [61].

Cette approche propose de combiner trois classificateurs homogènes basés sur des modèles de Markov cachés (HMM). Ces classificateurs utilisent la même topologie que le système de référence, mais se différencient uniquement par l'orientation de la fenêtre glissante. Dans ce travail, trois modèles de combinaison de ces classificateurs sont comparés au niveau de la décision.

Les performances de cette approche sont évaluées sur la base de données IFN/ENIT, qui contient 26 459 mots arabes correspondant à des noms de villes tunisiennes répartis en 946 classes. Les résultats obtenus révèlent un taux de reconnaissance de 90 %. Les résultats démontrent que la combinaison de ces classificateurs offre de meilleures performances par rapport à l'utilisation d'un seul classificateur.

2.5. La méthode proposée par J.H. AlKhateeb et al [62].

Ce travail présente un système de reconnaissance hors ligne des mots arabes manuscrits qui se compose de trois étapes principales : prétraitement, extraction des caractéristiques et classification. Dans la première étape, les mots sont segmentés à partir des scripts d'entrée et normalisés en termes de taille. Ensuite, chaque mot segmenté est divisé en blocs chevauchants. Dans la deuxième étape, des caractéristiques sont extraites à partir de ces blocs. Plus précisément, les valeurs absolues des moyennes calculées pour chaque bloc de mots segmentés sont utilisées pour former un vecteur de caractéristiques représentant le mot.

Enfin, dans la troisième étape, les vecteurs de caractéristiques sont utilisés pour classer les mots en utilisant le classifieur k plus proches voisins (KPPV). Le système proposé a été testé avec succès sur la base de données IFN/ENIT, qui contient 32 492 mots arabes manuscrits représentant 937 classes. Les résultats expérimentaux ont montré un taux de reconnaissance de 76.04 %.

2.6. La méthode proposée par J.H. Alkhateeb et al [63].

Ce travail propose un système de reconnaissance hors-ligne des mots arabes en utilisant des modèles de Markov cachés. Les mots sont segmentés, normalisés, et des caractéristiques sont extraites. Un classifieur HMM est utilisé pour la classification, et les caractéristiques structurelles sont utilisées pour reclasser les résultats. Les résultats expérimentaux sur la base de données IFN/ENIT montrent un taux de reconnaissance de 89.24 % (dans le top 1).

2.7. La méthode proposée par H. Nemmour et Y. Chibani [64].

Travail propose une méthode de reconnaissance des mots arabes manuscrits en combinant les transformées de ridgelets et les SVMs. Les ridgelets sont utilisés pour extraire des caractéristiques pertinentes à partir des mots manuscrits, tandis que la classification est réalisée à l'aide d'un classifieur SVMs multi-classes basé sur l'approche un-contre-tous.

L'expérimentation a été menée sur un vocabulaire de vingt-quatre classes extraites de la base de données IFN/ENIT. La performance des ridgelets a été évaluée en comparant les résultats obtenus avec les caractéristiques extraites à partir de la transformée de Radon et de la grille uniforme.

Les résultats expérimentaux ont montré un taux de reconnaissance de 84.0% en utilisant la méthode des ridgelets. Cela indique que les ridgelets sont efficaces pour extraire des caractéristiques discriminantes des mots arabes manuscrits, et le classifieur SVMs a été capable de classer les mots avec une précision satisfaisante.

2.8. La méthode proposée par M.T. Parvez et S.A. Mahmoud [65].

Ce travail présente un système de reconnaissance hors ligne de l'écriture du texte arabe en utilisant des techniques structurales. Le processus de reconnaissance se déroule en plusieurs étapes. Tout d'abord, une ligne de texte arabe est segmentée en mots ou en sous-mots, puis des points sont extraits sous forme de parties des mots arabes (PAW). Ensuite, l'inclinaison des PAW est corrigée à l'aide d'une nouvelle technique d'estimation et de correction basée sur l'approximation des polygonales.

La reconnaissance des PAW est réalisée à l'aide d'un nouvel algorithme de mise en correspondance de polygones flous. Ce processus permet de trouver la correspondance la plus probable entre les PAW et les modèles de caractères. La programmation dynamique est utilisée pour sélectionner la meilleure hypothèse d'une séquence de caractères reconnus pour chaque PAW.

Les résultats expérimentaux sont présentés sur la base de données IFN/ENIT, qui contient 32 492 noms de villes tunisiennes. Les résultats obtenus démontrent la robustesse et l'efficacité du système proposé, avec un taux de reconnaissance de 79,58%. Cela indique que le système est capable de reconnaître avec précision une grande partie des mots arabes présents dans la base de données.

2.9. La méthode proposée par H. Nemmour et Y. Chibani [66].

Ce travail propose un système de reconnaissance des mots arabes manuscrits utilisant un système de reconnaissance immunitaire artificiel et le transformé de Ridgelets. Les expérimentations réalisées sur la base de données IFN/ENIT montrent un taux de reconnaissance de 79.8%, démontrant ainsi l'efficacité de cette approche pour la reconnaissance des scripts arabes.

2.10. La méthode proposée par Amrouch et al [67].

Dans cet article, un modèle HMM basé sur CNN pour la reconnaissance de mots d'écriture manuscrite en arabe est présenté. HMM s'est avéré puissant dans la modélisation de la dynamique de l'écriture manuscrite. Pendant ce temps, CNN s'est admirablement acquitté de plusieurs tâches de vision par ordinateur, notamment la reconnaissance de caractères manuscrits. Dans ce modèle, le classificateur entraînable CNN est remplacé par le classificateur HMM. CNN agit comme un extracteur de caractéristiques génériques et HMM agit comme un identifiant. Le système proposé est supérieur au HMM de base basée sur les caractéristiques artisanales. Des expériences ont été réalisées sur la base de données IFN/ENIT.

2.11. La méthode proposée par Ghanim et al [68].

Introduire une approche en plusieurs étapes visant à la data mine Grand ensemble de données de pré-classification. L'étape du classement est Niveau intermédiaire qui réussit un groupe des classes les plus proches pour chaque test Image d'un classement final moins complexe. Classifier, L'effet de SVM et CNN a été rapporté indépendamment. Svm La classification est basée sur le vecteur de trait, qui est extrait de l'image d'entrée, tandis que la classification CNN est basée sur Les valeurs brutes des pixels de l'image d'entrée obtiennent un score de 95,6 %.

3. Comparaison entre les travaux

Dans ce qui suit, nous allons effectuer une comparaison entre les travaux d'autres systèmes qui ont également été testés sur la base IFN/ENIT.

Voici un tableau récapitulatif des principaux classifieurs utilisés par les chercheurs pour reconnaître les mots arabes manuscrits, ainsi que les taux de précision obtenus par chaque méthode de classification :

	Base de données utilisée	Classifieur utilisé	Année	Taille de la BDD	Taux Obtenu
M. Pechwitz et V.Maergner [58]	IfN / ENIT	HMM	2003	BDD version (1) App : (a, b, c) Test : (d)	89 %
P.Dreuw et al [59]	IfN / ENIT	HMM	2008	BDD Version (2) App: (a, b, c) Test : (d)	92.86 %
A.Benouareh et al [60]	IfN / ENIT	SCHMMs	2008	BDD version (1) App: (a, b, c) Test: d	90.20 %
R.A. Mohamad et al [61]	IfN / ENIT	Trois classificateurs basés sur HMM	2009	BDD version (1) App: (a, b, c) Test: d	90 %
J.H. Alkhateeb et al [62]	IfN / ENIT	KNN	2009	BDD Version (2) App : a, b, c et d Test: 20% (set e)	76.04 %
J.H. Alkhateeb et al [63]	IfN / ENIT	HMM + Reclassement	2011	26.459 (a, b, c, d)	89.24 %
H. Nemmour et Y.Chibani [64]	IfN / ENIT	SVM	2011	BDD version (2) 24 classes	84.0 %
M.T. Parvez et S.A. Mahmoud [65]	IfN / ENIT	Techniques structurelles Hmm	2012	BDD version (2) App : (a-d) Test : e	79,58%
H. Nemmour et Y. Chibani [66]	IfN / ENIT	Système de reconnaissance immunitaire artificiel (AIRS) SVM classifieur	2013	24 mots correspondant à 24 classes	79.8 %
Amrouch et al. [67]	IFN/ENIT	HMM	2018	BDD version (2) App : (a-b-c) Test : d	87.93%
		CNN—HMM			88.95%
		CNN—HMM		IFN/ENIT App: (a-b-c-d) Test: e	89.23%
Ghanim et al. [68]	IFN/ENIT Arabe base de données (mots)	CNN(AlexNet) +svm	2020	BDD version (2) 24 classes Correspondance, classement	95.6%

Tableau 2. Classifieurs utilisés dans les systèmes de reconnaissance de mots arabes manuscrits.

4. Conclusion

Dans ce chapitre, une étude comparative a été réalisée pour évaluer les travaux antérieurs dans le domaine de la reconnaissance de mots arabes manuscrits. Cette étude a été menée en utilisant plusieurs bases de données et en comparant les différentes méthodes et techniques utilisées dans chaque travail.

L'objectif principal de cette comparaison était de comprendre les forces et les faiblesses des approches existantes et d'identifier les lacunes ou les opportunités pour proposer une nouvelle approche. En analysant les résultats obtenus par chaque méthode.

Cette comparaison a permis de mettre en évidence les avantages et les inconvénients des différentes techniques, telles que les algorithmes de classification utilisés, les caractéristiques extraites, les méthodes de prétraitement, etc. Ces informations ont servi de base pour proposer une nouvelle approche qui pourrait combler les lacunes identifiées dans les travaux précédents.

Chapitre 4

Résultats expérimentaux

1. Introduction

Ce chapitre est divisé en deux parties. Dans la première partie, nous présenterons l'environnement dans lequel le travail de reconnaissance de mots manuscrits a été réalisé. Nous fournirons des détails sur les données utilisées, notamment la base de données de mots arabes manuscrits, ainsi que sur les outils et techniques utilisés pour mettre en place le système de reconnaissance.

La deuxième partie se concentrera sur les résultats obtenus lors de l'évaluation du système de reconnaissance de mots manuscrits. Nous mettrons en évidence les performances du système, en présentant des mesures telles que le taux de reconnaissance. Ces résultats permettront aux lecteurs d'avoir une idée précise de l'efficacité du système de reconnaissance.

2. Environnement de développement

Dans cette partie nous présentons l'environnement dans lequel nous avons réalisé ce travail.

2.1. Ressources matérielles

Un pc portable DELL latitude 5530 avec les caractéristiques suivantes :

- Processeur 12th Gen Intel(R) Core (TM) i5-1245U 1.60 GHz
- Une mémoire DDR4 d'une capacité de 16,00 Go
- Une carte graphique Intel® Iris Xe Graphics
- Windows 11 Professionnel
- Disque dur ssd 500 Go

2.2. Ressource logicielle

- Langage MATLAB -

MATLAB est un logiciel de calcul numérique et de programmation développé par la société The Math Works [52].

Nous avons utilisé dans ma projet MATLAB 2013a, fait référence à une version spécifique du logiciel sortie en 2013.

MATLAB est utilisé dans de nombreux domaines, notamment le traitement d'images et de signaux, les communications, les systèmes de contrôle industriels, la conception de réseaux intelligents et la robotique [53].

2.3. Base de données IFN/ENIT {v 2.0}

Dans notre projet, nous avons utilisé la base de données Tunisie-IFN/ENIT, qui est une ressource de mots arabes manuscrits développée dans le cadre d'une collaboration entre le laboratoire de Systèmes et de Traitement de Signal de l'ENIT (École Nationale d'Ingénieurs de Tunis) et le laboratoire de recherche sur la reconnaissance de l'écriture arabe de l'IFN (Institute Fuer Nachrichtentechnik). Cette base de données a été créée dans le but de favoriser la recherche et le développement de systèmes de reconnaissance de mots arabes manuscrits. Elle a déjà été utilisée par plus de 30 groupes de recherche et est disponible gratuitement sur le site www.ifnenit.com à des fins de recherche académique (non commerciales).

La version 2.0 de la base de données contient 26 459 images de noms composés de 1 à 4 mots, représentant un total de 115 585 sous-mots et 212 211 caractères. Ces données ont été rédigées par 411 scripteurs différents et représentent les 937 noms de villages tunisiens.

Cette base de données constitue une ressource essentielle pour la recherche dans le domaine de la reconnaissance de l'écriture arabe.

3. Schéma générale

Le schéma général résume les étapes qui ont été utilisées dans le système de reconnaissance de mots manuscrits arabes

Notre première étape est une image contenant le mot arabe manuscrit est fournie en entrée. Cette image provenir d'une base de données IFN/ENIT.

La prochaine étape importante est l'extraction des caractéristiques à partir de l'image. Dans cette étape, différentes techniques d'extraction de caractéristiques sont appliquées pour extraire les informations discriminantes des mots arabes. Parmi ces techniques, nous avons utilisé LBP (Local Binary Patterns), CLBP (Completed Local Binary Patterns) et OBIF (Orientation-Based Image Features).

Le LBP est une technique populaire qui consiste à extraire des motifs locaux à partir de l'image, en comparant les niveaux de gris des pixels voisins par rapport à un seuil. Cela permet de capturer les variations locales de l'intensité des pixels.

Le CLBP est une extension du LBP qui prend en compte les pixels manquants ou non valides dans l'image. Il complète les motifs locaux en utilisant des interpolations pour obtenir une représentation plus robuste des caractéristiques.

Le OBIF est une méthode basée sur l'orientation des images qui permet de capturer les informations relatives à la structure et aux motifs des caractères arabes. Dans notre système, nous avons utilisé les descripteurs de column OBIF, qui se concentrent sur les colonnes verticales de l'image pour extraire les caractéristiques.

Dans notre système, la fonction d'extraction OBIF utilise plusieurs paramètres pour capturer les informations contenues dans les images.

L'epsilon est une valeur utilisée dans l'algorithme OBIF pour déterminer la précision de l'estimation de l'orientation des images. Il contrôle la taille des régions voisines considérées lors du calcul de l'orientation locale. Plus la valeur d'epsilon est petite, plus la région considérée est petite, ce qui permet une estimation plus précise de l'orientation.

Dans notre système, nous avons utilisé différentes valeurs d'epsilon pour évaluer leur impact sur les performances de reconnaissance des mots arabes manuscrits. Les valeurs d'epsilon que nous avons utilisées sont 0.1, 0.01 et 0.001.

Un autre paramètre important est le sigma, nous avons utilisé une plage de valeurs pour le paramètre sigma, allant de [1] à [16], afin de définir la taille des fenêtres utilisées pour l'extraction des caractéristiques OBIF. Cette approche nous a permis de capturer des informations à différentes échelles spatiales dans les images des mots arabes manuscrits. En utilisant des fenêtres de différentes tailles, nous avons pu tenir compte des variations de taille des mots et extraire des informations plus complètes sur leur structure.

La fonction d'extraction LBP et CLBP utilise plusieurs paramètres pour capturer les informations contenues dans les images.

La fonction d'extraction LBP et CLBP utilise plusieurs paramètres pour capturer les informations contenues dans les images des mots arabes manuscrits. Ces paramètres comprennent :

P : Les paramètres importants pour la fonction d'extraction LBP et CLBP dans la capture

d'informations des images des mots arabes manuscrits sont le rayon, le nombre de points et les règles de codage.

R : En ajustant ces paramètres, nous pouvons obtenir différentes représentations des caractéristiques LBP et CLBP, ce qui peut affecter les performances de la classification des mots arabes manuscrits.

- $P = (2, 4, 6, 8, 16)$
- $R = (1, 2 \dots, 16)$

Classification en SVM : Une fois les caractéristiques extraites, elles sont utilisées comme entrée pour un classificateur SVM (Support Vector Machine). Le SVM est un algorithme d'apprentissage supervisé qui peut être utilisé pour la classification de données. Il est entraîné sur un ensemble d'exemples de mots arabes manuscrits préalablement étiquetés pour apprendre à distinguer différentes classes de mots.

Décision finale : Après la classification, le SVM attribue une étiquette (classe) à chaque mot arabe manuscrit. La décision finale consiste à prendre en compte les résultats de classification pour déterminer la classe prédite pour le mot donné. Selon le schéma général présenté, le mot est classé en utilisant le SVM, et cette classification peut être considérée comme la décision finale.

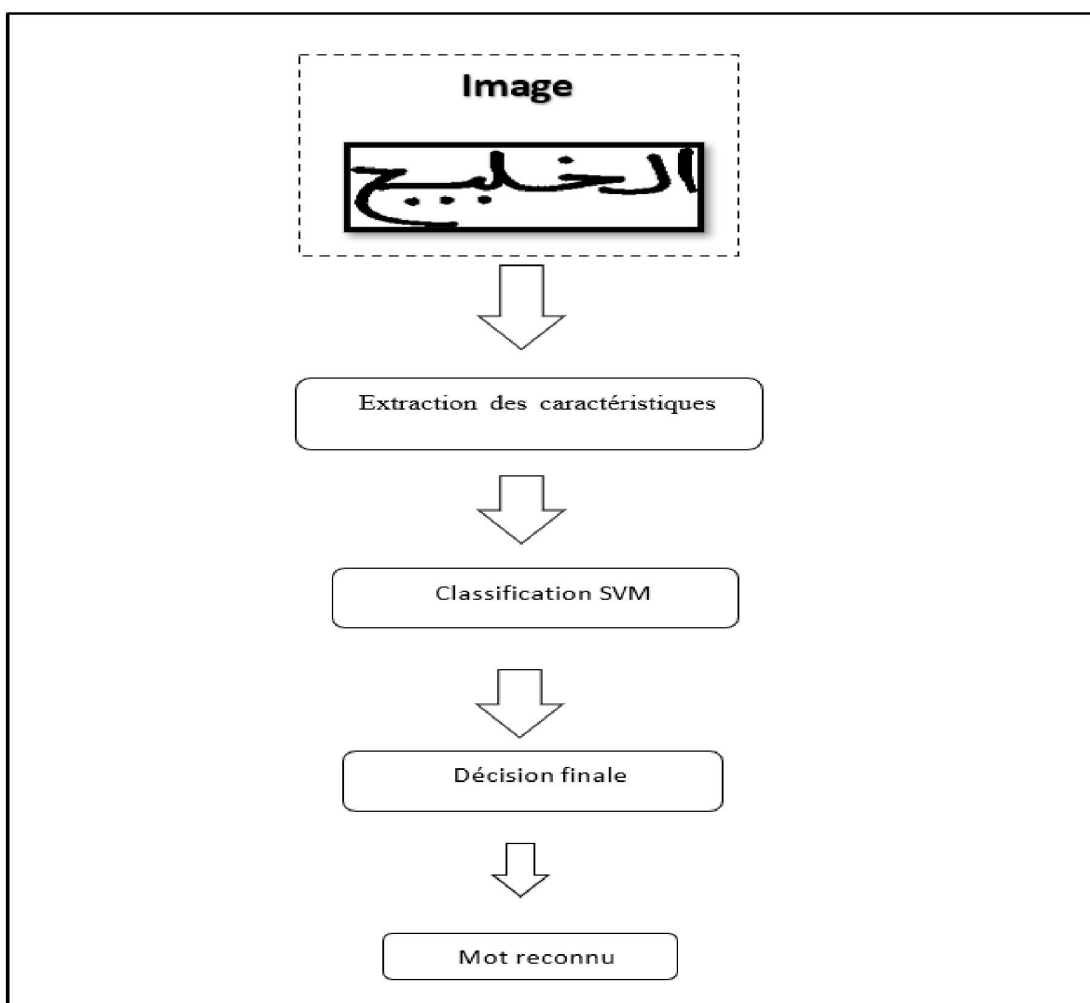


Figure 31. Schéma général du système de reconnaissance de mot manuscrit.

4. Résultats et discussions

Nous avons utilisé la Support Vector Machine (SVM) comme classificateur pour la tâche de classification. Voici les caractéristiques qui ont été extraites : Les histogrammes de column des oBIF sont utilisés pour former notre (SVM) et initier l'apprentissage du classificateur de classes défini. Le noyau Radial Basis Function (RBF) a été utilisé, avec un paramètre de noyau défini sur [0, 100] et le paramètre de marge souple C est fixé à 10.

4.1. Expérimentation avec OBIF

Dans cette expérimentation avec OBIF Les résultats de classification ont été enregistrés, différentes configurations ont été testées pour évaluer les performances de la méthode d'extraction des caractéristiques OBIF dans la reconnaissance de mots arabes manuscrits.

4.1.1. Obif simple

Nous avons utilisé la fonction d'extraction OBIF, où la base de données a été divisée en quatre classes (50,100,150,200) et les valeurs suivantes ont été utilisées :

- Epsilon (0.1, 0.01, 0.001)
- Sigma (1, 2, 4, 8, 12, 16)

epsilon	sigma	Taille de vecteur	SVM CLASSIFIER							
			50 class		100 class		150 class		200 class	
			δ	Taux	δ	TAUX	δ	TAUX	δ	TAUX
0.1	1	23	3	23.75	4	18.72	4	15.54	3	14.22
	2	23	4	26.44	4	22.93	4	20.33	3	18.54
	4	23	3	34.52	3	28.69	5	24.54	5	18.56
	8	23	3	34.52	3	30.16	3	29.45	6	20.87
	16	23	6	43.23	3	37.03	6	30.86	3	29.69
0.01	1	23	4	23.75	3	19.13	5	18.54	4	15.89
	2	23	4	26.36	3	17.73	3	16.45	4	12.55
	4	23	3	33.80	5	23.55	3	20.85	5	18.54
	8	23	3	37.84	3	32.64	5	27.25	3	23.97
	12	23	6	47.82	4	41.77	4	36.53	5	31.82
	16	23	5	47.58	4	40.06	3	36.31	4	30.35
0.001	1	23	3	24.38	6	22.54	5	18.58	3	15.48
	2	23	3	26.44	4	25.63	4	23.01	3	20.12
	4	23	4	32.22	3	29.88	4	27.44	3	24.57

	8	23	3	39.74	3	38.59	3	36.89	3	28.11
	12	23	5	47.11	3	40.58	3	36.01	3	30.16
	16	23	5	46.41	3	39.98	3	34.81	3	29.12

Tableau 3. Résultats de classification pour obif pour chaque classe (50,100,150,200).

On peut voir que l'histogramme de column des oBIF à $\sigma = \{12\}$ alors qu' $\epsilon = 0,01$

Et $\delta = \{6\}$ surpasse les autres caractéristiques, nous obtenons le taux 47.82%.

4.1.2. Concaténations des vecteur obif

On utilise le paramètre Epsilon = $\{0.01\}$ de configuration qui donne le meilleur taux avec defferent Sigma.

50 class			
Sigma	Taille de vecteur	δ	TAUX
[1,2]	46	6	30.40
[1,4]	46	7	40.70
[1,8]	46	8	47.66
[1,12]	46	6	58.51
[1,16]	46	6	57.09
[2,4]	46	6	38.72
[2,8]	46	6	49.80
[2,12]	46	6	59.46
[2,16]	46	9	57.80
[4,8]	46	8	50.67
[4,12]	46	6	60.10
[4,16]	46	7	59.38
[8,12]	46	5	56.22
[8,16]	46	7	57.96
[12,16]	46	5	57.32
[1,2,4]	69	7	40.86
[1,8,12]	69	5	61.76
[1,12,16]	69	9	64.84
[2,4,8]	69	10	54.00
[2,12,16]	69	7	65.40
[4,8,12]	69	6	63.42
[4,12,16]	69	10	65.08
[8,12,16]	69	5	61.44
[1,2,4,8]	92	9	55.11

[1,8,12,16]	92	9	67.30
[2,4,8,12]	92	7	64.53
[2,8,12,16]	92	8	66.67
[4,8,12,16]	92	7	66.35
[1,2,4,8,12]	115	11	64.21
[1,4,8,12,16]	115	10	68.96
[2,4,8,12,16]	115	8	68.25
[1,2,4,8,12,16]	138	10	69.04

Tableau 4. Résultats de classification pour obif.

On peut constater que l'histogramme à column des oBIFs, lorsque $\text{Sigma} = \{1,2,4,8,12,16\}$ et $\delta = \{10\}$, donne un meilleur résultat de 69,04%.

4.1.3. Avec normalisation

Le vecteur résultant est normalisé pour avoir une moyenne nulle et une variance unitaire.

Sigma	Taille de vecteur	δ	TAUX
[2,8,12,16]	92	10	67.50
[4,8,12,16]	92	15	68.84
[1,2,4,8,12]	115	8	66.29
[1,4,8,12,16]	115	10	69.15
[2,4,8,12,16]	115	13	69.01
[1,2,4,8,12,16]	138	15	69.83

Tableau 5. Résultats de classification pour obif Avec normalisation.

Après avoir ajouté la fonction normalisation, on remarque une amélioration du résultat.

4.2. oBIF column

Nous avons utilisé la fonction d'extraction OBIF column, où les valeurs suivantes ont été utilisées :

Epsilon (0.01)

Sigma ([1,2], [1,4] ... [2,16] + [8,16])

Sigma	Taille de vecteur	δ	TAUX
[1,2]	529	18	35.39
[1,4]	529	21	58.27
[1,8]	529	21	74.11
[1,12]	529	25	77.12
[1,16]	529	19	77.75
[2,4]	529	22	59.07
[2,8]	529	23	78.54
[2,12]	529	23	79.49
[2,16]	529	24	81.71
[4,8]	529	23	78.31
[4,12]	529	23	85.59
[4,16]	529	28	84.96
[8,12]	529	20	77.83
[8,16]	529	24	83.29
[12,16]	529	23	73.24
[8,16] + [4,12]	1058	34	89.79
[4,12] + [4,16]	1058	32	87.73
[2,16] + [8,16]	1058	32	87.57

Tableau 6. Résultats de classification pour obif column.

4.2.1. Avec normalisation

Sigma	Taille de vecteur	δ	TAUX
[8,16] + [4,12]	1058	35	90.23
[4,12] + [4,16]	1058	37	87.88
[2,16] + [8,16]	1058	30	87.60

Tableau 7. Résultats de classification pour obif column Avec normalisation.

Lorsque nous avons utilisé la fonction d'extraction obif column Avec normalisation, nous avons obtenu le meilleur résultat dans le système de reconnaissance de mots arabe manuscrite (Taux = 90,23%), où les valeurs suivantes ont été utilisées :

Sigma = ([2,16] + [8,16]) //////////////// $\delta = 35$

4.3. LBP

Nous avons utilisé la fonctionnalité d'extraction LBP, les valeurs suivantes ont été utilisées :

- $P = (2, 4, 6, 8, 16)$
- $R = (1, 2 \dots, 16)$
- $\delta = 1 \dots 100$
- Classifieur SVM

P	R	Taux	δ	Taille de vect
2	1	16.23	2	5
	2	20.19	3	
4	1	27.87	5	15
	2	35.47	4	
	4	42.68	6	
8	1	29.69	4	59
	2	46.40	10	
	4	63.34	9	
	8	69.83	7	
16	1	29.69	5	243
	2	45.84	11	
	4	63.42	11	
	8	72.45	18	
	16	61.20	17	

Tableau 8. Résultats de classification pour LBP (P, R)

On peut constater que classification pour LBP, lorsque :

$P = (16), R = (8), \delta = \{18\}$ donne un meilleur résultat de 72.45%.

4.3.1. Concaténations des vecteur LBP

Pour chaque LBP (P, R) on calcule les deux vecteurs :

V1 vecteur de caractéristiques pour (P1, R1)

V2 vecteur de caractéristiques pour (P2, R2)

Concaténation : $V = V1.V2$

(R, P)	Taux	δ	Taille de vect
(2 ,1) (2,2)	16.63	7	10
(4 ,1) (4,2)	35.79	7	30
(4 ,1) (4,4)	45.28	7	
(4 ,2) (4,4)	44.89	5	
(8 ,1) (8,2)	46.87	11	118
(8 ,1) (8,4)	62.70	9	
(8 ,1) (8,8)	65.34	9	
(8 ,2) (8,4)	70.70	8	
(8 ,2) (8,8)	72.45	12	
(8 ,4) (8,8)	75.61	9	
(16 ,1) (16,2)	47.19	15	486
(16 ,1) (16,4)	64.05	17	
(16 ,1) (16,8)	73.00	12	
(16 ,1) (16,16)	64.85	16	
(16 ,2) (16,4)	62.39	12	
(16 ,2) (16,8)	74.82	15	
(16 ,2) (16,16)	72.37	17	
(16 ,4) (16,8)	76.25	22	
(16 ,4) (16,16)	77.51	21	
(16 ,8) (16,16)	77.75	17	

Tableau 9. Résultats de classification pour LBP avec concaténation.

Le meilleur résultat obtenu pour la classification avec la méthode de concaténation LBP dans l'expérimentation était atteint avec les paramètres suivants :

$$V1 = (16, 8) + V2 = (16, 16)$$

Valeur de δ : 17

Ce paramétrage a permis d'obtenir un taux de reconnaissance de 77.75%.

4.3.2. Avec normalisation

(R, P)	Taux	Δ	Taille de vect
(16 ,8)	73.31	16	243
(16 ,8) (16,16)	78.78	16	486
(16 ,4) (16,16)	78.31	22	
(8 ,4) (8,8)	76.41	15	118

Tableau 10. Résultats de classification pour LBP Avec normalisation.

Le meilleur résultat obtenu lors de l'ajout de fonction normalisation est (78.78 %), avec valeurs suivantes ont été utilisées :

$$V1 = (16 ,8) + V2 = (16,16)$$

$$\delta = 16$$

4.4. CLBP

Les mêmes valeurs ont été utilisées que dans la fonctionnalité d'extraction LBP.

P	R	Taux	δ	Taille de vect
2	1	17.50	4	10
	2	22.41	3	
4	1	28.58	5	30
	2	39.67	5	
	4	51.94	7	
8	1	28.98	6	118
	2	47.03	14	
	4	66.98	9	
	8	75.69	13	
16	1	27.95	7	486
	2	48.22	21	
	4	68.25	23	
	8	77.98	26	
	16	71.73	21	

Tableau 11. Résultats de classification pour CLBP.

On peut constater que classification pour CLBP, lorsque :

$$P = (16), R = (8), \delta = \{26\} \text{ donne un meilleur résultat de } 77.98\%.$$

4.4.1. Concaténations des vecteur LBP

(R, P)	Taux	δ	Taille de vect
(2 ,1) (2,2)	17.54	7	20
(4 ,1) (4,2)	35.42	7	60
(4 ,1) (4,4)	50.36	7	
(4 ,2) (4,4)	51.39	6	
(8 ,1) (8,2)	46.56	16	118
(8 ,1) (8,4)	65.95	15	
(8 ,1) (8,8)	75.85	20	
(8 ,2) (8,4)	64.37	17	
(8 ,2) (8,8)	75.77	17	
(8 ,4) (8,8)	80.52	15	
(16 ,1) (16,2)	46.95	29	486
(16 ,1) (16,4)	67.14	21	
(16 ,1) (16,8)	78.38	25	
(16 ,1) (16,16)	73.40	27	
(16 ,2) (16,4)	67.38	25	
(16 ,2) (16,8)	79.89	21	
(16 ,2) (16,16)	78.94	23	
(16 ,4) (16,8)	81.00	26	
(16 ,4) (16,16)	82.74	35	
(16 ,8) (16,16)	84.71	38	

Tableau 12. Résultats de classification pour CLBP avec concaténation.

Le meilleur résultat de classification avec la méthode de concaténation CLBP dans l'expérimentation a été obtenu en utilisant les paramètres suivants: $V1 = (16, 8)$ et $V2 = (16, 16)$, avec une valeur de δ égale à **38** . Cette configuration a conduit à un taux de reconnaissance de 84.71%.

4.4.2. Avec normalisation

(R, P)	Taux	δ	Taille de vect
(16,8)	78.46	27	243
(16,8) (16,16)	85.35	37	486
(16,4) (16,16)	83.05	29	
(8,4) (8,8)	80.20	18	118

Tableau 13. Résultats de classification pour CLBP Avec normalisation.

Le meilleur résultat obtenu lors de l'ajout de fonction normalisation est (85.35%), avec valeurs suivantes ont été utilisées :

$$V1 = (16,8) + V2 = (16,16)$$

$$\delta = 37$$

5. Résultats : Présentation Et Comparaison

5.1. Les résultats obtenus

Les performances des différentes fonctions d'extraction utilisées dans cette thèse ont été comparées, et les meilleurs résultats ont été identifiés.

Le descripteur	Taux
Obif column Avec normalisation	90.23
Obif column	89.79
CLBP Avec normalisation	85.35
CLBP avec concaténation	84.71
LBP Avec normalisation	78.78
CLBP	77.98
LBP avec concaténation	77.75
LBP	72.45
Obif Avec normalisation	69.83
Obif	69.04

Tableau 14. Comparez les résultats obtenus.

Les résultats obtenus dans l'expérimentation indiquent que la fonction d'extraction OBIF column avec normalisation a été la plus performante dans la reconnaissance des mots arabes manuscrits. Cette méthode a montré une précision élevée, avec un taux de reconnaissance atteignant 90,23% lorsqu'elle a été utilisée avec les paramètres spécifiques. L'OBIF column est une technique qui permet de capturer efficacement les caractéristiques importantes des mots arabes manuscrits, ce qui facilite leur classification et leur reconnaissance. De plus, l'ajout de la normalisation des vecteurs a contribué à améliorer les résultats, en garantissant une moyenne nulle et une variance unitaire pour les vecteurs d'OBIF column. Ces résultats soulignent l'efficacité de cette méthode dans le domaine de la reconnaissance des mots arabes manuscrits, offrant ainsi des perspectives prometteuses pour des applications pratiques telles que la numérisation et l'indexation de documents.

5.2. Comparaison

Dans ce tableau on va comparer notre système avec les travaux de littérature basée sur les techniques classique de machine Learning.

La personne	Classification	Taux
P.Dreuw et al [60].	HMM	92.86
Notre système	SVM	90.23
A.Benouareh et al [61].	SCHMMs	90.20
R.A.Mohamad et al [62].	HMM	90
M. Pechwitz et V.Maergner [58].	HMM	89
H. Nemmouret Y. Chibani [64].	SVM	84

Tableau 15. Comparer les résultats obtenus avec d'autres travaux.

Le tableau compare notre système de reconnaissance de mots arabes manuscrits basé sur la méthode SVM avec d'autres travaux de littérature utilisant des techniques classiques de machine learning. Les taux de reconnaissance obtenus sont les suivants :

Nous constatons que notre système de reconnaissance de mots arabes manuscrits atteint un taux de reconnaissance de 90.23%, ce qui le positionne parmi les meilleurs résultats de la littérature. Cela démontre l'efficacité de notre approche dans la reconnaissance de caractères arabes manuscrits.

En comparaison, les travaux de P.Dreuw et al [60] utilisant la méthode HMM ont obtenu un taux de reconnaissance de 92.86%. Bien que leur méthode ait donné de bons résultats, notre système s'en approche étroitement, ce qui souligne la performance de notre méthode basée sur SVM.

D'autres travaux utilisant des méthodes telles que SCHMMs [61], HMM [62], et SVM [64] ont également obtenu des taux de reconnaissance élevés, mais notre système les dépasse en termes de précision. Cela montre que la méthode OBIF column avec normalisation utilisée dans notre

système est compétitive et offre de bons résultats par rapport à d'autres approches classiques de machine learning.

Il est important de noter que la reconnaissance de mots arabes manuscrits est un défi complexe en raison des variations dans l'écriture, la connectivité des lettres et la présence de diacritiques. Notre système s'est montré efficace en surmontant ces difficultés et en fournissant des résultats prometteurs.

Cependant, il convient également de souligner que les performances peuvent varier en fonction de la taille de l'ensemble de données utilisé, des méthodes de prétraitement des images et des paramètres spécifiques utilisés dans chaque approche. Il est donc essentiel de mener des comparaisons approfondies et de considérer ces aspects lors de l'évaluation des systèmes de reconnaissance de mots arabes manuscrits.

Notre système de reconnaissance de mots arabes manuscrits basé sur la méthode OBIF column avec normalisation a démontré de bons résultats et s'est classé parmi les meilleurs systèmes de la littérature. Cela souligne son potentiel dans diverses applications telles que l'OCR, la vérification de signatures et la reconnaissance de documents.

6. Conclusion

Ce chapitre a présenté la base de données utilisée, l'environnement de développement et les étapes de mise en place du système de reconnaissance des mots arabes manuscrits, nous avons présenté les résultats obtenus par notre système, avec la meilleure performance atteignant 90,23% en utilisant le descripteur OBIF column Avec normalisation et le classifieur SVM. Ces résultats sont encourageants par rapport aux travaux précédents.

Conclusion **Générale**

Conclusion générale

La reconnaissance automatique de l'écriture arabe est un domaine de recherche qui a été étudié pendant de nombreuses décennies et qui présente de nombreuses applications industrielles. Cependant, jusqu'à présent, aucune solution optimale n'a été trouvée pour résoudre le défi de reconnaître avec précision et efficacité l'écriture arabe, en raison de la complexité inhérente à sa forme. Malgré cela, des progrès significatifs ont été réalisés dans ce domaine. Par conséquent, la recherche sur la reconnaissance automatique de l'écriture arabe manuscrite reste active et ouverte, dans le but de développer de nouvelles avancées et d'explorer de nouvelles approches pour relever ce défi.

Dans notre étude, nous nous sommes concentrés sur la conception d'un système de reconnaissance hors-ligne des mots arabes manuscrits en utilisant la nouvelle base de données IFN/ENIT, qui comprend des mots représentant les noms de villes tunisiennes.

L'objectif principal de notre recherche était d'obtenir des résultats de reconnaissance satisfaisants et de développer un système d'identification efficace. Pour cela, nous avons utilisé plusieurs descripteurs tels que LBP, CLBP, OBIF et OBIF column, ainsi que le classifieur SVM.

Nous avons mené des expérimentations sur la base de données IFN/ENIT, et les résultats ont montré un taux de reconnaissance de 90,23% en utilisant le classifieur SVM et le descripteur OBIF column.

En conclusion, nous suggérons quelques méthodes qui pourraient contribuer à améliorer les performances du système :

- La combinaison de la méthode d'extraction des caractéristiques LBP avec d'autres méthodes.
- L'utilisation d'autres classifieurs tels que Kmeans, HMM et CNN.

En conclusion, ce mémoire contribue à la recherche sur la reconnaissance des mots arabes manuscrits en proposant un système efficace basé sur des descripteurs et un classifieur spécifiques. Des pistes d'amélioration sont également suggérées pour continuer à progresser dans ce domaine.

Bibliographie

- [1] S. Bouzariata, « Segmentation des textes manuscrits,» Memoire master, Université de Tébessa, 2014 .
- [2] Zerdani Amina « Reconnaissance de caractères arabes manuscrits par réseau de Hopfield » mastère en informatique université Oum el bouaghi Juin 2014
- [3] <https://zanoubya123.typepad.com/blog/2008/10/th%C3%A9matique-lediff%C3%A9rents-styles-de-la-calligraphie-arabe.html>
- [4] R. Bouslimi, « Système de reconnaissance hors-lignes des mots manuscrits arabe pour multi-scripteurs,» Université de Jendouba, Jendouba, 2006.
- [5] R.Azizi, «Une approche Hybride pour la reconnaissance d'écriture arabe manuscrite» Université de Constantine, Constantine, 2007.
- [6] https://www.memoireonline.com/09/09/2694/m_Systeme-de-reconnaissance-hors-ligne-des-mots-manuscrits-arabe-pour-multi-scripteurs2.html
- [7] <https://www.cairn.info/revue-etudes-et-documents-berberes-2015-1-page-189.htm>
- [8] B. Al-Badr and R.M. Haralick, "Symbol recognition without prior segmentation", Conference SPIE-EI, 1994.
- [9] F. Grandidier, « Un nouvel algorithme de sélection de caractéristiques -Application à la lecture automatique de l'écriture manuscrite,» Thèse de Doctorat, Ecole de Technologie Supérieure, Université du Québec, Canada, 2003.
- [10] S. Nemouchi, « Reconnaissance de l'écriture arabe par systèmes flous,» Memoire magister, Université Badji Mokhtar, Annaba, 2010.
- [11] N.Zermi, «Reconnaissance de mots manuscrits arabes par les modèles de Markov cachés et les réseaux de neurones,» Thèse de doctorat, Université de Annaba, 2007.
- [12] N. Benahmed – Optimisation de réseaux de neurones pour la reconnaissance de chiffres manuscrits isolés : Sélection et pondération des primitives par algorithmes génétiques, École de technologie supérieure, 2002.
- [13] F.Biadsy, SEI-Sana, N.Habash, "Online Arabic handwriting recognition using Hidden Markov Models", Proceeding of IWFHR'06, 10th International Workshop on Frontiers in Handwriting Recognition, pp. 85-90, La Baule, France 2006.
- [14] L.Souici-Meslati, «Reconnaissance des mots arabes manuscrits par intégration neuro-symbolique,» Thèse de Doctorat d'Etat, Labo. LRI, Département d'informatique, Université d'Annaba, Algérie, Annaba, 2006.

- [15] S. Touj, N. B. Amara and H. Amiri, "Generalized Hough transform for Arabic optical character recognition", ICDAR'03, pp.1242-1246, 2003.
- [16] N. Benamara : « Utilisation des modèles de Markov cachés planaires en reconnaissance de l'écriture arabe imprimée ». Thèse de doctorat, spécialité Génie Electrique, Université des sciences, des Techniques et de médecine de Tunis II, 1999.
- [17] P.M Lallican, C. Viarp-Gaudin and S. Knerr, « From off-line to on-line handwriting recognition ». Proc. 7th workshop on frontiers in handwriting recognition, pp. 303-312, Amsterdam, 2000.
- [18] G.Abdeldjalil, «Segmentation automatique pour la reconnaissance numérique des chèques bancaires Algériens » ; Thèse de MAGISTER, centre Universitaire de Khanchela, 2011.
- [19]30/04/2023.[Enligne].Available:https://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones_artificiels.
- [20] J.Anigbogu, «"Reconnaissance de textes imprimés mutifontes à l'aide de modèles stochastiques et métriques,» Thèse de doctorat, Université de Nancy I, 1992.
- [21] T. de Campos, B. R. Babu, and M. Varma. Character recognition in natural images. International Conference on Computer Vision Theory and Applications, 2009.
- [22]https://www.memoireonline.com/12/09/3040/m_La-liaison-automatique-des-plusieurs-images-percues-sur-un-scanner4.html#fn8
- [23] <https://www.leaders.com.tn/article/30460-numerisation-des-manuscrits-historiques-arabes-pour-une-meilleure-valorisation>
- [24] <https://www.memoireonline.com/09/09/2694/Systeme-de-reconnaissance-hors-ligne-des-mots-manuscrits-arabe-pour-multi-scripteurs.html>
- [25] <https://www.institut-anwar.fr/la-chahada-profession-de-foi-tout-savoir/>
- [26] D'argenton,«Contribution à la segmentation et la reconnaissance de l'écriture manuscrite» Thèse de Doctorat. Institut National des Sciences Appliquées de Lyon,174 pages. 1994
- [27] Liwicki, M., & Bunke, H. (2005). IAM off-line handwriting recognition contest: a review. International Journal on Document Analysis and Recognition, 7(2-3), 107-118.
- [28] Gonzalez, R. C., & Woods, R. E. (2008). Digital image processing (3rd ed.). Pearson Education.
- [29] S. Chevalier et al : « Étude de primitives spectrales pour la reconnaissance de caractères manuscrits dans le cadre d'une approche markovienne 2D », DGA/Centre d'Expertise Parisien, France, Novembre 2005.

- [30] A.J. Newell, L.D. Griffin, R.M. Morgan, P.A. Bull, "Texture-based estimation of physical characteristics of sand grains," in Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA), 2010, pp. 504-509.
- [31] A. Gattal, Y. Chibani, C. Djeddi and I. Siddiqi, "Improving Isolated Digit Recognition using a Combination of Multiple Features," in Proc.of 14th International Conference on Frontiers in Handwriting Recognition (ICFHR-2014), Crete Island, Greece ,2014, pp. 446–451.
- [32] A. Britto, R. Sabourin, F. Bortolozzi, C.Y.Suen. « Complementary features combined in an HMM-based system to recognize handwritten digits ». In 12th International Conference on Image Analysis and Processing(ICIAP), Mantova, Italy, 2003, pp. 670-675.
- [33] L. D. Griffin, M. Lillholm, M. Crosier, and J. Sande, "Basic Image Features (BIFs) Arising from Approximate Symmetry Type," in Scale Space and Variational Methods in Computer Vision, vol. 5567, X.-C. Tai, K. Mørken, M. Lysaker, and K.-A. Lie, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 343-355.
- [34] L. D. Griffin and M. Lillholm, "Symmetry Sensitivities of Derivativeof- Gaussian Filters," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 6, pp. 1072-1083, 2010.
- [35] A.J. Newell, L. Griffin, Natural image character recognition using oriented basic image features, in: Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA), 2011, pp. 191-196.
- [36] Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. European Conference on Computer Vision, 430-443.
- [37] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in Computer Vision-ECCV 2004. Springer, 2004, pp.469–481.
- [38] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 6, pp. 915–928, 2007.
- [39] X. Wang, T. X. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009, pp.32–39.
- [40] M. Lutf, X. You, and H. Li, "Offline arabic handwriting identification using language diacritics," in Pattern Recognition (ICPR), 2010 20th International Conference on. IEEE, 2010, pp. 1912–1915.
- [41] L. Du, X. You, H. Xu, Z. Gao, and Y. Tang, "Wavelet domain local binary pattern features for writer identification," in Pattern Recognition (ICPR), 2010 20th International Conference on. IEEE, 2010, pp. 3691–3694.

- [42] M. Ferrer, A., A. Morales, and U. Pal, "Towards explainable writer verification and identification using vantage writers," in Document Analysis and Recognition, 2013. ICDAR 2013. Twelvth International Conference on, vol. 1. IEEE, 2013, pp. 369–373.
- [43] M. Pietikäinen, A. Hadid, M. Pietikäinen and A. Hadid, Computer Vision Using Local Binary Patterns, New York: Matti Pietikäinen Abdenour Hadid, 2011.
- [44] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
- [45] Z. Guo, L. Zhang and D. Zhang, "A Completed Modeling of Local Binary Pattern Operator for Texture Classification," in IEEE Transactions on Image Processing, vol. 19, no. 6, pp. 1657-1663, June 2010, doi: 10.1109/TIP.2010.2044957
- [46] <https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501879-machine-a-vecteurs-de-support-svm-definition-et-cas-d-usage/>
- [47] Bernhard Scholkopf, Alexander J. Smola "Learning with Kernels, Support Vector Machines, Regularization, Optimization, and Beyond", the MIT Press 2002.
- [48] <https://www.datasciencecentral.com/implementing-a-soft-margin-kernelized-support-vector-machine/>.
- [49] Schölkopf, B., & Smola, A. J. (2002). Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT Press.
- [50] <https://www.ajcact.org/en/2020/12/17/les-techniques-algorithmiques-de-lia-svm/>
- [51] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media.
- [52] Li, S. Z. (2012). Encyclopedia of biometrics. Springer Science & Business Media.
- [53] D. Jean-Luc, Introduction à MATLAB, Université de Picardie.
- [54] Z. Guo, L. Zhang and D. Zhang, "A Completed Modeling of Local Binary Pattern Operator for Texture Classification," in IEEE Transactions on Image Processing, vol. 19, no. 6, pp. 1657-1663, June 2010, doi: 10.1109/TIP.2010.2044957.
- [55] Newell, A.J., et al.: Texture-based estimation of physical characteristics of sand grains. In: 2010 International Conference on Digital Image Computing: Techniques and Applications, pp. 504–509 (2010).
- [56] Gattal, A., et al.: Isolated handwritten digit recognition using obifs and background features. In: 2016 12th IAPR Workshop on Document Analysis Systems (DAS), pp. 305–310 (2016).
- [57] Newell, A.J., Griffin, L.D.: Writer identification using oriented basic image features and the delta encoding. Pattern Recogn. 47(6), 2255–2265 (2014).
- [58] M. Pechwitz, V. Maergner, p. 5, 2003.

- [59] P.Dreuw, S.Jonas, H.Ney, p. 4, 2008.
- [60] A. Benouareth, A. Ennaji, M. Sellami, p. 11, 2008.
- [61] R.A.Mohamad,L.Likforman-Sulem et C. Mokbel, p. 13, 2009.
- [62] J.H. AlKhateeb, F. Khelifil, J. Jiani, S.S. Ipsonl, p. 4, 2009.
- [63] J.H. Alkhateeb, J. Ren, J. Jiang, H. Al-muhtaseb, p. 8, 2011.
- [64] H.Nemmour, Y.Chibani, p. 5, 2011.
- [65] M.T. Parvez, S.A.Mahmoud, p. 14, 2012.
- [66] H.Nemmour, Y.Chibani, p. 4, 2013.
- [67] M. Amrouch, M. Rabi, and Y. Es-Saady, “Convolutional feature learning and CNN based HMM for Arabic handwriting recognition,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Jul. 2018, vol.
- [68] Ghanim, T.M.; Khalil, M.I.; Abbas, H.M.: Comparative Study on Deep Convolution Neural Networks DCNN-Based Offline Arabic Handwriting Recognition. IEEE Access 8, 95465–95482 (2020).

