Système Neuro-Markovien pour la reconnaissance de l'écriture manuscrite arabe à vocabulaire limité

Brahim Farou[1] , Samir Hallaci[1], Hamid Seridi[1,2]

[1]Département d'informatique, Université 08 mai 45 Guelma, B. P. 401, Algérie
[2]CResTIC, EA 3804, Université de Reims, B.P 1035, 51687, Reims, Cedex, France
LAIG, Université 08 mai 45 Guelma, B. P. 401, Guelma 24000, Algérie
{fbrahim24, s.hallaci,seridi}@yahoo.fr

Résumé. Nous proposons une manière de coopérer des MMC et des réseaux neuronaux dans une architecture probabiliste en tirant avantage des deux outils : la génération d'une liste des N meilleures hypothèses de mots ainsi que leurs segmentations en caractères par un classifieur MMC et les propriétés de modélisation des réseaux neuronaux appliquées aux caractères. Le classifieur RN utilise la segmentation du classifieur MMC afin de retourner à l'image du mot et d'extraire les caractéristiques convenables à la reconnaissance de caractères isolés. Le classifieur RN réévalue chacune des N meilleures hypothèses des mots et les scores générés sont combinés avec ceux du classifieur MMC. Finalement, la liste des N meilleures hypothèses est réordonnée selon les nouveaux scores faisant ainsi ressortir la meilleure hypothèse. Nous avons utilisé deux bases de données. La première contient 14400 échantillons écrits sur du papier par 100 scripteurs différents et 3 occurrences pour chaque mot. Les mots de cette base constituent un lexique de 48 mots des wilayas algériennes. La deuxième contient 2800 caractères segmentés manuellement à partir des mots de la première base. Avec un taux de réussite de 91,77 %, le système proposé a montré une bonne performance.

Mots clés : Reconnaissance de l'écriture manuscrite, Segmentation et Classification, MMC, RN, Viterbi, Baum welch.

1 Introduction

Dans les dix dernières années, des progrès considérables ont été réalisés dans le domaine de la reconnaissance de l'écriture manuscrite. Ce progrès est du d'une part aux nombreux travaux effectués dans ce domaine et d'autre part a la disponibilité de bases de données internationales standards relatives à l'écriture manuscrite qui permettait aux chercheurs de rapporter de façon crédible les performances de leurs approches dans ce domaine, avec la possibilité de les comparer avec d'autres approches vu qu'ils utilisent les mêmes bases.

La langue arabe n'a pas eu cette chance, contrairement au latin, elle reste encore au niveau de la recherche et de l'expérimentation [1], c'est-à-dire que le problème reste encore un pari ouvert pour les chercheurs. L'écriture arabe étant par nature cursive, elle pose de nombreux problèmes aux systèmes de reconnaissance automatique. Le problème le plus difficile lors de la conception d'un système de

reconnaissance de l'écriture manuscrite est la segmentation des mots manuscrits en vue de leur reconnaissance, qui n'est pas toujours triviale et demande beaucoup de temps et de calcul. D'autre part, les informations locales sont un peu négligées dans les systèmes se basant sur une analyse globale ce qui peut diminuer considérablement leurs performances [2]. Pour remédier à ces problèmes, des approches hybrides ont été proposées pour la reconnaissance des mots arabes manuscrits dans un vocabulaire limité. Un tel système nécessite la prise en compte d'un nombre très important de variabilités.

Notre travail consiste à la conception d'un système de reconnaissance d'écriture manuscrite arabe dans un vocabulaire limité, nous proposons une approche hybride basée sur les réseaux de neurones et les modèles de Markov cachés comme outil de classification. Le schéma suivant illustre l'architecture de notre système.

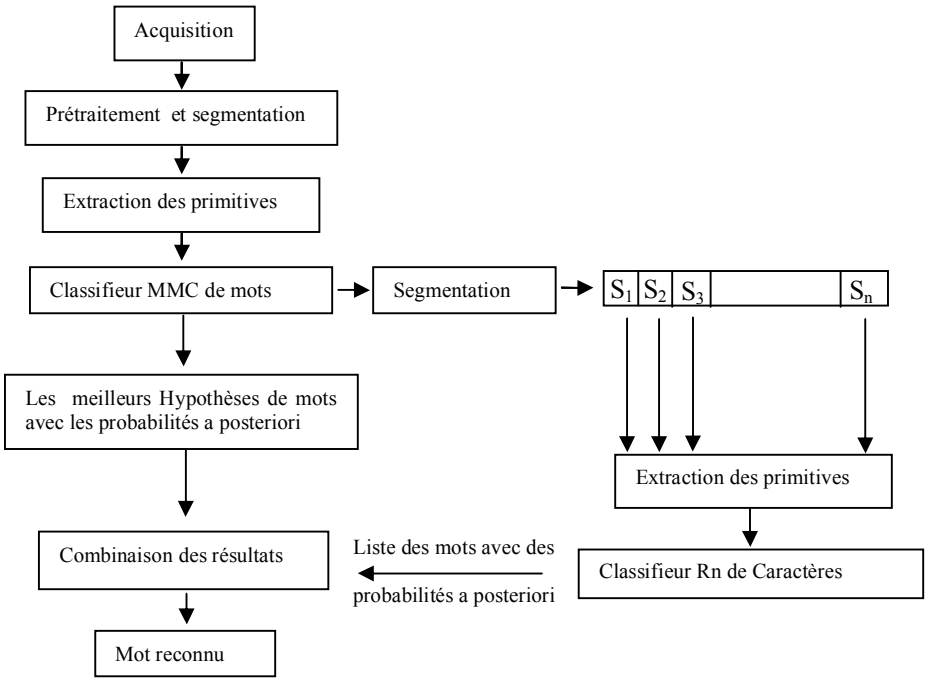

Fig.1. Architecture du système

2 Prétraitement

Les prétraitements appliqués sur l'image du mot permettent, d'une part, d'éliminer ou de réduire le bruit dans l'image, et d'autre part, de simplifier les traitements ultérieurs. Dans notre système nous avons utilisé la binarisation, le lissage, la normalisation, le cadrage, la squelettisation, la correction de la déformation des caractères et l'estimation de la ligne de base.

Le but de la binarisation est de faire surgir l'information utile par rapport à l'arrière-plan, malheureusement à cause de la mauvaise qualité de l'image reçue en entrée (Niveau du gris de l'arrière-plan très élevé) nous étions obligés de tester plusieurs seuils afin de trouver un compromis entre les différentes images utilisées.

L'opération de lissage est appliquée afin d'éliminer les bruits introduits dans l'image à cause des systèmes d'acquisition, les effets de temps ou tout simplement à cause de la qualité du papier et du stylo utilisé, en vue de la décrire par une séquence de vecteurs de caractéristiques plus au moins stables.

La normalisation est une tache nécessaire lorsque l'acquisition n'est pas réalisée avec un scanner relié au système (image existante). En effet si l'entrée du système est une image externe par rapport au système (Notre Système par exemple), il faux impérativement ramener les caractères a la même taille, car à cause de la variation des fontes ou des opérations d'agrandissement ou de réductions de la taille des images, les caractères peuvent subir une légère déformation dans la taille ce qui complique les tâches de segmentation et influence sur la stabilité des paramètres.

L'opération de cadrage consiste à chercher la première et la dernière ligne / colonne signifiante (pixel <> de l'arrière-plan), ensuite créer une nouvelle image cadrée à partir de l'image mère.

La squelettisation est l'une des techniques les plus utilisées dans la reconnaissance des formes. Elle permet de diminuer l'information utile en ne gardant que le squelette de la forme. Le principe est de ramener l'image du mot à une écriture linéaire d'une épaisseur égale à un pixel, en préservant la forme, la connexité et la topologie du tracé.

Nous désirons de cette étape la correction de la déformation des caractères. En effet après la squelettisation du corps des caractères, plusieurs déformations de type concave et convexe ont apparu au niveau des lignes continues (supposées sans concave ni convexe). La fig.2 montre quelques déformations détectées après l'étape de squelettisation.

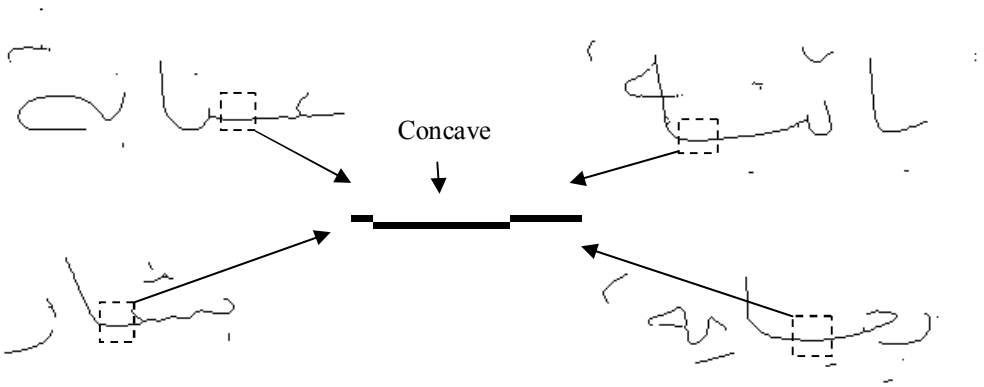

Fig.2. Déformation des mots après la squelettisation
(Création de concave et convexe non désirée)

Pour remédier à ce problème, nous avons proposé un algorithme qui permet de redresser les lignes. Cet algorithme se base sur le principe de continuité c.-à-d. si deux segments horizontaux ou verticaux qui se trouvent sur la même ligne respectivement

colonne et s'il existe un autre segment horizontal respectivement vertical et ce dernier contient des pixels voisins avec les deux segments alors c'est une déformation horizontale respectivement verticale. La correction s'effectue par le déplacement du segment (voir Fig. 3). L'algorithme peut se résumer dans les instructions suivantes :

Algorithme
Répéter
Pour chaque ligne / colonne faire
1) Détecter les bords du segment **A**
2) Détecter les bords du segment **B**
3) S'il existe un troisième segment **C** dont les pixels des bords sont voisins avec le premier et le deuxième segment alors :
   a. Créer un nouveau segment entre les segments **A** et **B**
   b. Supprimer le segment **C**

Fsi
Fpour
Jusqu'à « aucune modification n'est possible sur l'image »

L'extraction de certaines caractéristiques (c.-à-d. points diacritiques) demande l'estimation de la ligne de base d'écriture du mot. La méthode utilisée donne une bonne estimation de la ligne de base [6]. Elle est basée sur l'analyse de l'histogramme de projection horizontale.

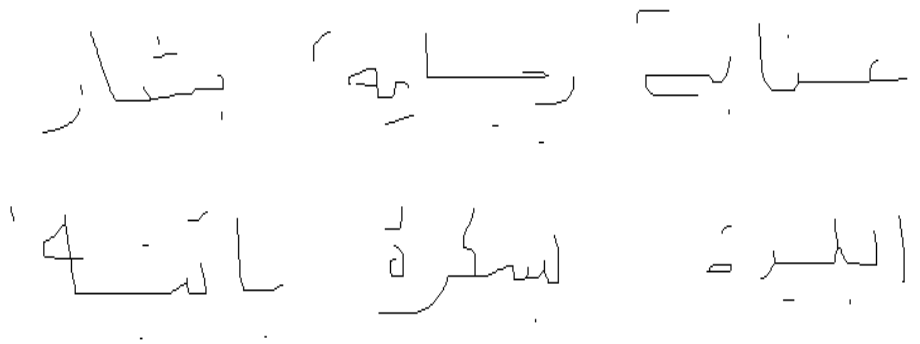

Fig.3. Résultats de l'opération de correction de la déformation sur des mots manuscrits arabe

3 Segmentation

Nous avons utilisé dans notre système une segmentation non uniforme basée sur l'analyse de l'histogramme de projection verticale. Le but de cette méthode est de diminuer la dimension de l'information contenue dans l'image. Cette technique se

base sur le principe que la liaison entre deux caractères est la partie la plus mince du tracé manuscrit.

4 Extraction des caractéristiques

L'identification directe du mot à partir de son image (matrice de pixels) est presque impossible à cause de la grande variabilité inhérente au style d'écriture utilisé et au bruit entachant l'image. D'où la nécessité d'extraire, à partir de la représentation en pixels du mot, un ensemble de caractéristiques permettant d'identifier facilement ce dernier. Ces primitives doivent être discriminatives et invariantes vis-à-vis les différentes transformations que peut subir l'image telle que la rotation, variation de la taille… etc.

Nous avons utilisé dans notre système un mélange entre les caractéristiques statistiques et structurelles qui peut d'après la littérature donner de meilleurs résultats. Le choix des caractéristiques n'est pas une tâche simple, malheureusement il n'y a pas de théorie qui permet de choisir telle ou telle caractéristique. Après plusieurs tests, nous avons choisi les caractéristiques suivantes :

- Les sept moments invariants.
- Le nombre de boucles.
- Le nombre et le type de chaque concave (vers la Gauche, vers la Droite, vers le Haut et vers le Bas).
- Les hampes et les Jambes.
- Le nombre, le type et la position des points diacritiques (Hamza, Madda et chapeau inclus).
- Nombre de points extrêmes.
- Nombre de points de branchement.
- Nombre de points de croisement.
- Le nombre de composantes connexes

Le résultat de cette étape est un vecteur de caractéristiques comportant 26 caractéristiques statistiques et structurelles.

5 Classifieur MMC

Un modèle de Markov caché discret est un automate probabiliste à nombre d'états finis constitué de N états. C'est un processus aléatoire qui se déplace d'état en état à chaque instant, et on note q_t le numéro de l'état atteint par le processus à l'instant t.

L'état réel q_t du processus n'est pas directement observable, il est caché, mais peut être observé par un autre processus aléatoire qui émet après chaque changement d'état un symbole o_t. Dans le cas d'un processus markovien d'ordre 1, la probabilité de passer de l'état i à l'état j à l'instant t et d'émettre o_t ne dépend ni du temps, ni des états aux instants précédents [3].

La modélisation du classifieur nécessite la définition des observations émises par les états du modèle et l'architecture des modèles de mots [4].

Pour définir l'architecture du modèle, nous devons tenir compte de la topologie et le nombre d'états du modèle [5]. La topologie adoptée dans notre système est de type droite-gauche conformément à l'écriture arabe avec saut inter-états et intra-état. Ce type de modèles a l'avantage de conserver la notion du temps dans la modélisation, s'approchant ainsi de la nature de l'écriture. En outre, c'est le type le moins gourmand en temps de calcul et en nombre de paramètres à estimer lors de l'apprentissage. Dans notre modélisation chaque état correspond à un caractère c.-à-d. le nombre d'états est différent d'un modèle a un autre, ce qui nécessite la création de 48 modèles représentant les 48 wilayas algériennes.

Nous avons considéré les caractéristiques statistiques et structurelles extraites à partir de l'image de mot comme des observations pour notre modèle. La quantification vectorielle garantit la transformation du vecteur de caractéristiques en une séquence discrète d'observation.

Quantification vectorielle

Les HMMs utilisés pour la modélisation des mots sont de nature discrète, leurs densités de probabilités d'observations sont discrètes, ce qui nécessite l'utilisation d'un quantifieur vectoriel pour faire correspondre chaque vecteur continu à un indice discret d'un dictionnaire de référence (CodeBook). Une fois le dictionnaire de référence obtenu, cette correspondance entre les vecteurs caractéristiques des trames et les indices du dictionnaire deviennent un simple calcul de type plus proche voisin.

Apprentissage et reconnaissance

L'apprentissage des paramètres des modèles HMMs (A, B, Π) correspondant aux classes de mots est réalisé par l'algorithme de Baum-Welch [5] en appliquant les formules de Russel [7] et Levinson [8] pour estimer les paramètres des différentes distributions de probabilité d'état. Cet algorithme permet d'aligner les observations sur les états, et d'une façon générale, converge vers un minimum local du fait de manque de données [9]. La reconnaissance est effectuée par la recherche du modèle discriminant, elle peut se faire simplement par le calcul des probabilités d'émission de la forme par les modèles que l'on suppose a priori équiprobables. La forme à reconnaître est affectée à la classe dont le modèle fournit la probabilité la plus importante.

$$\lambda^* = \text{argmax}_{\lambda \in A} \; P(O/\lambda) \qquad \textbf{(1)}$$

Où A désigne l'ensemble des modèles . L'évaluation de la probabilité de chaque modèle est réalisée grâce à une méthode à base de programmation dynamique qui est l'algorithme de Viterbi [10].

6 Classifieur RN

Le classifieur que nous avons utilisé est un PMC à rétro-propagation du gradient d'erreur à une couche cachée [11]- [12]. Les 26 caractéristiques statistiques et structurelles extraites à partir des segments de caractère isolés générés par le module

de segmentation sont les entrées du réseau. La couche cachée est composée de 25 neurones. Les classes à discriminer sont les 28 caractères de l'alphabet arabe, d'où le choix de 28 neurones pour la couche de sortie. La fonction d'activation des neurones est la fonction sigmoïde unipolaire. Ce choix de classifieur est basé sur les critères suivant : sa rapidité, sa capacité à traiter des données hétérogènes et le plus important, s'il est bien entraîné, un PMC estime des probabilités bayésiennes a posteriori [14] [15]. Ce dernier point est très important dans notre modélisation vu que nous avons utilisé les modèles de Markov cachés et que ce dernier génère aussi des probabilités a posteriori. Donc la mise en œuvre de la combinaison des résultats va être simplifiée.

7 Combinaison des résultats

Nous voulons de cette étape la correction des inconvenances générées par les modèles de Markov cachés. En effet, malgré le progrès réalisé au niveau de la reconnaissance de la parole et de l'écriture manuscrite par les MMCs, nous leur reprochons de négliger un peu les informations locales. De plus, la condition d'indépendance imposée par le modèle de Markov (chaque observation doit être indépendante des observations voisines) rend les MMC incapable de tirer avantage de la corrélation qui existe réellement parmi les observations d'un même caractère, mais le faible pouvoir discriminatif reste l'inconvénient majeur des MMCs.

Pour effectuer la combinaison des résultats, nous avons calculé un nouveau score des mots à partir des probabilités a posteriori de chaque classifieur. En effet étant donné que les deux classificateurs estiment des probabilités a posteriori en sortie, nous pouvons calculer un score composé P^* par combinaison des sorties des classificateurs.

$$P^* = \log(P_{MMC}) + \log(P_{PMC}) \quad \textbf{(2)}$$

La sortie du système (complet) est une liste des N meilleures hypothèses. Le choix du mot candidat est effectué par rapport à la probabilité la plus élevée.

8 Tests et résultats

Pour construire notre système et évaluer ses performances, nous avons utilisé deux bases de données. La première contient 14400 échantillons écrits sur du papier par 100 scripteurs différents et 3 occurrences pour chaque mot. Les mots de cette base constituent un lexique de 48 mots des wilayas algériennes. Les échantillons de la base de données utilisée ont été collectés au sein du laboratoire de recherche en informatique d'Annaba. La deuxième contient 2800 caractères segmentés manuellement à partir des mots de la première base. Le choix des mots à segmenter est pris au hasard parmi les 14400 mots contenus dans la première base. Nous avons pris de chaque caractère 100 exemplaires qui n'appartiennent pas forcement au même scripteur. Nous avons construit la deuxième base pour éviter l'apprentissage collectif.

La base des caractères a permis simplement de réaliser l'apprentissage du Perceptron multicouche, tandis que la base des mots a servi pour l'apprentissage des modèles de Markov cachés et pour le test des performances du système. Nous avons divisé la base de mots en deux sous-bases, la première contient 75 % (10800 échantillons) des mots pour l'opération d'apprentissage et la deuxième contient 25 % (3600 échantillons) des mots pour les tests [16].

Nous allons dans ce qui suit montrer seulement les résultats finaux de la reconnaissance des mots manuscrits arabes de notre système c.-à-d. avec toutes les corrections effectuées au niveau des modules du système (version finale du système). La meilleure configuration de notre système a donné les résultats illustrés dans le Tableau 1.

Tableau 1 Résultats de la reconnaissance.

Wilayas	**Taux de reconnaissance (en %)**	**Taux de rejet (en %)**
أدرار	94,54	05,46
الشلف	88,37	11,63
الأغواط	89,71	10,29
أم البواقى	88,06	11,94
باتنة	92,32	07,68
بجاية	91,71	08,29
بسكرة	90,24	09,76
بشار	88,43	11,57
البليدة	94,32	05,68
البويرة	91,18	08,82
تبسة	93,42	06,58
تمنراست	88,37	11,63
تلمسان	90,44	09,56
تيارت	93,75	06,25
تيزي وزو	94,25	05,75
الجزائر	93,51	06,49
الجلفة	89,18	10,82
جيجل	92,46	07,54
سطيف	93,72	06,28
سعيدة	94,38	05,62
سكيكدة	92,31	07,69
سيدي بلعباس	90,54	09,46
عنابة	94,10	05,90
قالمة	88,81	11,19
قسنطينة	90,87	09,13
المدية	93,84	06,16
مستغانم	93,03	06,97
مسيلة	92,17	07,83
معسكر	89,36	10,64
ورقلة	89,58	10,42
وهران	91,86	08,14

البيض	94,62	05,38
اليزي	93,34	06,66
برج بوعريريج	90,64	09,36
الطارف	93,61	06,39
بومردا س	93,62	06,38
تند وف	90,37	09,63
تسمسيلت	91,03	08,97
الوادي	88,29	11,71
خنشلة	91,08	08,92
سوق أهراس	92,74	07,26
تيبازة	92,66	07,34
ميلة	89,12	10,88
عين الدفلة	94,23	05,77
النعامة	92,10	07,90
غليزان	94,32	05,68
غرداية	92,13	07,87
عين تيموشنت	92,50	07,50
Taux Moyen en %	91,77	08,23

D'après les résultats du Tableau 1 nous remarquons que le meilleur taux de reconnaissance est de **94,62 %** et le mauvais taux de reconnaissance et de **88,06 %** avec un taux de réussite globale de **91.77 %** et un taux de rejet de **08,23 %.** Nous voulons de cette présentation montrer que malgré la simplicité du script de quelques mots (الشلف, بشار, قالمة, ميلة) le taux de reconnaissance était nettement faible par rapport aux autres mots. En effet, nous avons effectué une recherche minutieuse au niveau de chaque étape du processus de reconnaissance et nous avons remarqué que les images de ces mots ont perdu de l'information au niveau de la binarisation. Donc c'est la qualité médiocre des images qui a perturbé le processus de reconnaissance.

Concernant le taux d'échec dans chaque classe, il est justifié par les erreurs commises au niveau des autres modules de notre système.

Malheureusement à cause de l'architecture de notre système nous n'avons pas pu tester chaque sous-système à part pour montrer l'influence de chaque sous-système sur l'autre. Pour cela nous avons essayé de comparer notre système avec celui de Benzenache dans [17] basé sur les MMC et celui de Zaghdoudi dans [18] basé sur le PMC. Nous estimons que la comparaison été crédible vu que nous avons utilisé la même base de mots.

Tableau 2 Comparaison des résultats de notre système avec celui de Benzenache [17] et Zaghdoudi [18]

	Notre Système (MMC/RN)	**Système de Benzenache (MMC continu)**	**Système de Zaghdoudi (RN)**
Taux de réussite	91,77 %	89,05 %	91,23 %
Taux de rejet	08,23 %	10,95 %	08,77 %

Le tableau 2 montre que notre système a donné de meilleurs résultats par rapport aux deux systèmes de référence. Donc, l'apport de l'hybridation est très clair pour la phase de reconnaissance.

9 Conclusion

Nous avons présenté dans ce chapitre notre modélisation pour un système de reconnaissance de l'écriture manuscrite arabe à vocabulaire limité. Notre approche est basée sur l'hybridation des deux classifieurs les plus utilisés dans le domaine de la reconnaissance des formes et en particulier la reconnaissance de l'écriture et de la parole.

Malgré un taux de réussite de 91,77 % que nous estimons encourageant, notre système est loin d'atteindre la perfection. Les erreurs de classification sont dues d'une part à l'utilisation des MMCs discrets. En effet l'utilisation de la quantification vectorielle permet de représenter chaque classe par un vecteur unique (centroïde) ce qui influence sur la qualité de l'information véhiculée dans chaque vecteur. D'autre part, la qualité des images de la base est très médiocre ; la numérisation des images est une étape extrêmement importante dont il faut prendre le soin de le faire.

Références

1. Essoukhri Ben Amara, N. : Problématique et orientations en reconnaissance de l'écriture arabe, CIFED'2002, Colloque International Francophone sur l'Ecrit et le Document, pp.1-10, Hammamet, Tunisie, Octobre 2002.
2. Essoukhri Ben Amara, N., Belaïd, A., Ellouze, N.: Utilisation des modèles markoviens en reconnaissance de l'écriture arabe: Etat de l'art, CIFED'2000, Colloque International Francophone sur l'Ecrit et le Document, pp.181-191, Lyon, France, 2000.
3. Bourlard, H., Morgan, N.: Continuous speech recognition by connectionist statistical methods. IEEE Trans. on Neural Networks, 1993.
4. Benzenache, A.: Reconnaissance hors-ligne des mots arabes manuscrits par les modèles de Markov cachés, Mémoire de magister, Département Génie Electrique, Université 08 mai 45, Guelma, Algérie, 2007.
5. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE Vol. 77, No. 4, pp. 336-349, August 1989.
6. Pechwitz, M., Maergner, V.: HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT- Database, ICDAR (Proceedings of the Seventh International Conference on Document Analysis and Recognition), pp: 890-894, 2003.
7. Russel, M. J., Moore, R. K.: Explict Modeling of State Occupancy in Hidden Markov Models for Speech Recognition, Proceeding of ICASSP (International Conference on Acoustic, Speech and Signal Processing), pp: 5-8, 1985.
8. Levinson, S.E: Continuously Variable Duration Hidden Markov Models for Automatic Speech Recognition. Computer, Speech & Language, Vol 1, N° 1, pp: 29-45, 1986.
9. Saon, G. : Modèles Markoviens uni-bidimensionnels pour la reconnaissance de l'écriture manuscrite hors-ligne, Thèse de doctorat, Université Henri Poincaré Nancy 1, 1998.
10. Forney, G.D. : The Viterbi Algorithm, Proc IEEE, Vol. 61, No. 3, pp. 268- 278, March 1973.

11. Koerich, A.: Large vocabulary off-line handwritten word rocognition, Thèse de Docteur Ecole de Technologie Supérieur, 2002.
12. Cheriet, M., Suen, C.Y.: Un système neuro-flou pour la reconnaissance de montants numériques de cheques arabes, Pattern Recognition letters 14(1993), pp, 1009-1017.
13. Davalo, E., Naim, P. : Des réseaux de neurones, Eyrolles.1993
14. Richard, M.D., lippmann, R.P.: Neural Network Classifiers Estimate Bayesian a Posteriori Probabilities, Neural Computation, vol. 3, 1991, p. 461–483.
15. Farah, N., Souici, L., Sellami, M.: Arabic Word Recognition by Classifiers and Context, JCST, Journal of Computer Science and Technology, Vol. 20, No. 3, pp. 402-410, May 2005.
16. Pechwitz, M., Maergner, V.: HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT- Database, ICDAR (Proceedings of the Seventh International Conference on Document Analysis and Recognition), pp : 890-894, 2003.
17. Benzenache, A. : Reconnaissance hors-ligne des mots arabes manuscrits par les modèles de Markov cachés, Mémoire de magister, Département Génie Electrique, Université 08 mai 45, Guelma, Algérie, 2007.
18. Zaghdoudi, R. : Reconnaissance hors-ligne des mots arabes manuscrits par les réseaux de neurones, Mémoire de magister, Département Génie Electrique, Université 08 mai 45, Guelma, Algérie, 2008.