Une Méthode Rapide de Reconnaissance de l'Écriture Arabe Manuscrite

H. MILED, C. OLIVIER, M. CHERIET*, K. ROMEO-PAKKER

Laboratoire d'Informatique Industrielle et Images, Université de Rouen 76821 Mont-Saint-Aignan Cédex
Téléphone : 02 35 14 68 74, Télécopie : 02 35 14 66 18
*Ecole de Technologie Supérieure, Université du Québec, Montréal, Canada
e-mail : olivier@la3i.univ-rouen.fr

RESUME

Nous décrivons une méthode rapide de reconnaissance off-line de l'écriture arabe manuscrite. Le problème de la reconnaissance est découpé en différentes sous-tâches et distribué à plusieurs agents. La segmentation des mots arabes en graphèmes est effectuée en analysant le contour supérieur des composantes connexes qui nous sert de signal utile pour la détection des points de segmentation primaires PSP. Une analyse locale détermine les points de segmentation décisifs PSD. Les primitives se rapportant à chaque mot sont analysés dans un premier module de reconnaissance où la décision est donnée par maximum de vraisemblance. Un deuxième module effectue l'étiquétage des observations HMM par rapport aux caractères. Les résultats des deux modules sont analysés.

ABSTRACT

A fast recognition method for Arabic handwritten characters is proposed. The recognition problem is divided in several tasks and distributed to appropriate agents. In order to segment Arabic words into graphemes we analyse the upper contour of the connected parts of words. This signal is used to the detection of primary segmentation points PSP. A local analysis gives the decisive segmentation points, DSP. Each word is analysed with its characteristics and the decision is calculated with a maximum likelihood classifier. A second classification is performed with the modelisation by Hidden Markov Models (HMM) applied on the graphemes. The results of the two classifiers are discussed.

1. Introduction

Depuis dix ans la reconnaissance des caractères arabes prend un nouvel essor et fait l'objet d'articles de plus en plus nombreux. [Badr 95] énumère de nombreux travaux sur la reconnaissance des documents arabes dont certains s'intéressent à la reconnaissance de caractères imprimés ou manuscrits. Malgré ce nouvel élan, les travaux de recherche restent peu nombreux par rapport aux besoins administratifs, bancaires et postaux. Dans cet article, nous nous intéressons à la recherche d'une méthode rapide de reconnaissance off-line de l'écriture arabe manuscrite dans le but d'une lecture automatique de support comme les libellés d'enveloppes. Notre système de reconnaissance est basé sur la coopération de deux approches complémentaires par rapport à l'information qu'elles amènent au niveau du mot (approche globale) et au niveau du caractère (approche analytique) [Oliv 96]. Une organisation multi-agent efficace est utilisée pour coordonner les données calculées par différents agents pendant la phase de segmentation en graphèmes. Le problème de chevauchement des caractères est résolu grâce à l'utilisation du contour supérieur des tracés. Une analyse floue permet la décision des points de segmentation douteux. L'ensemble des informations obtenues, après l'analyse du mot, permet une représentation du mot en une séquence de codes. Ces codes sont utilisés par un classifieur sur des modèles de Markov cachés (HMM) selon le modèle de [Chen 94]. Cette application nouvelle sur les caractères arabes considère la possibilité de sursegmentation des caractères au maximum en trois graphèmes et s'adapte parfaitement aux exigences des règles grammaticales arabes et aux formes des caractères. Après un bref aperçu sur les caractéristiques de l'écriture arabe, nous exposons la segmentation des mots arabes en graphèmes et nous décrivons les primitives se rapportant à chaque mot. Les différents modules de la partie reconnaissance des mots sont définis avec les résultats sur une base de 9440 mots écrits par 40 scripteurs.

2. Caractéristiques de l'écriture arabe

L'écriture arabe s'écrit de droite à gauche. Elle est cursive, c'est-à-dire que les lettres sont liées généralement entre elles. Chaque caractère peut prendre quatre formes différentes, suivant sa position dans le mot. Un ensemble de pixels noirs adjacents les uns aux autres est appelé une composante connexe. Cette dernière, dans l'écriture arabe, ne représente pas

forcément un mot entier, elle peut être seulement une partie du mot, car certains caractères ne doivent pas être attachés à leur successeur à gauche dans le mot. Par ailleurs, il existe des lettres différentes qui ont la même forme, mais qui se distinguent par la position et le nombre de points qui leur appartiennent. Les voyelles "a", "i" et "ou" ne sont pas utilisées systématiquement dans l'écriture arabe; des signes qui correspondent à des voyelles sont employés pour éviter des erreurs de prononciation. On peut distinguer deux types de textes : les textes avec ou sans les signes de voyelles. Quelques textes arabes (Le Coran et les livres d'apprentissage de la lecture et de l'écriture pour les enfants) contiennent des signes de voyelles. Les autres, c'est-à-dire les livres, les journaux, les publications sont des textes sans ces signes.

3. Segmentation du mot en Graphèmes

On considère qu'un mot est une forme parmi 236 classes à reconnaître. Un mot peut être formé de plusieurs parties connexes (Fig. 1) car même si l'écriture arabe est cursive il existe des lettres qui ne s'attachent jamais à la lettre suivante. Nous appelons ces lettres, des caractères de fin de tracé et les analysons séparément [Rom1 95].

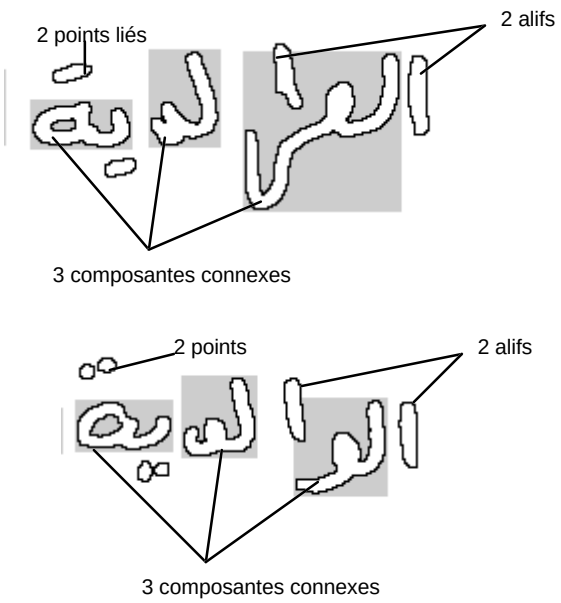

Figure 1. Deux mots appartenant à la même classe.

Le problème de la reconnaissance est découpé en différentes sous-tâches et distribué en plusieurs entités appelées Agents. Chaque agent réagit sur son environnement suivant les données qui lui sont fournies. La détection de la ligne de base se fait par un agent par projection horizontale et corrigée par une vérification de l'existence des occlusions dans les cas où les appendices de fin de tracé prennent des dimensions importantes et induisent en erreur le positionnement de la ligne d'écriture [Ame 94].

L'extraction des points et des signes diacritiques ainsi qu'une première classification indiquant le tracé le plus proche est effectuée par un deuxième agent. Les données sont stockées sur un tableau noir et récupérées par un autre agent qui analyse les composantes connexes et trie en composantes principales ou alifs. Le mot est désormais représenté sous forme d'un arbre avec ses tracés numérotés représentés comme noeuds et signes diacritiques, les occlusions, les alifs sont représentés sur les branches appartenant au noeud correspondant. Ces arbres permettent d'accéder directement au tracé recherché et ainsi diminuer le temps d'analyse du mot à l'étape de reconnaissance.

La détection des entités connexes formant le mot se fait par une méthode de suivi de contour rapide basée sur le code de Freeman. En parcourant la forme à analyser dans le sens contraire des aiguilles d'une montre si on ne retient que les codes "ouest", "nord-ouest" et "sud-ouest" nous obtenons le contour supérieur du tracé. Le contour inférieur est décrit généralement avec des données symétriques mais avec moins de relief. C'est pourquoi il n'est pas conservé, c'est le contour supérieur qui nous sert de signal utile pour la modélisation et le codage [Rom2 95], [Beg 94]. Le contour supérieur permet l'élimination du chevauchement dû aux caractères à extension haute et la détection des points de segmentation primaires (PSP).

Un automate fini qui nous sert d'agent sélectionne les points de segmentation décisifs (PSD) en élargissant la zone de recherche par analyse locale du signal utile codé reflétant les différents changements dans l'ordre chronologique (Fig. 2). Les minima locaux du contour supérieur sont jugés décisifs si on ne détecte ni occlusion ni chevauchement sur l'image d'origine; et si l'épaisseur du trait en ce PSP est inférieure à un seuil α x (l'épaisseur du trait); α étant un coefficient de proportionnalité égale à 1,5 dans notre cas.

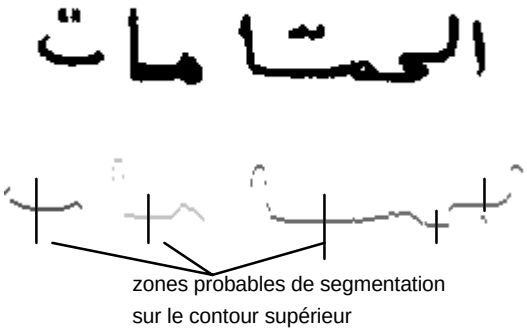

Fig. 2 Les Points de Segmentation Primaires (PSP) et la segmentation du mot.

Nous avons testé la segmentation sur une base de test de 6000 mots écrits par 20 scripteurs d'origines différentes. Nous obtenons une segmentation en 32328 graphèmes avec 98.82 % de bonne segmentation et un taux de 0.83 % de sous-segmentation avec 0.35 % de sur-segmentation.

Une étude statistique sur 3000 mots de notre base démontre que les caractères de fin de tracé représentent 40% des graphèmes, et que 28 % de ces caractères appartiennent à quatre classes de formes [Mil 96]. Ce qui justifie une analyse approfondie du signal utile sur la forme globale, la variation de la fréquence du signal et la position de quelques paliers particuliers par rapport à la ligne de base. Une analyse floue est effectuée par un agent pour la description floue de la forme et pour obtenir la position du point de segmentation par rapport à la ligne de base avec l'analyse du graphème précédent. Trois décisions peuvent découler de ces analyses: Si le module de l'analyse floue est douteux on décide le rejet; et s'il n'y a pas d'objection on fusionne les deux graphèmes. Sur 13097 graphèmes de fin de tracé traités nous obtenons 98.12 % d'extraction réussie avec 0.88 % de taux d'erreur et 1 % de rejet.

4. Extraction de primitives

Pendant la phase de segmentation nous recençons les données suivantes que nous vectorisons pour la phase de reconnaissance. Un mot est représenté par un vecteur de 9 primitives normalisées:

a) Les alifs sont comptabilisés en dehors des tracés,
b) Les points et les signes diacritiques situés au-dessus du tracé,
c) Les points et les signes diacritiques situés au-dessous du tracé,
d) Les PSD et leurs positions,
e) Le nombre de graphèmes dans chaque tracé,
f) La longueur du mot en graphèmes,
g) Le nombre de tracés sans compter les alifs,
h) Le nombre de hampes dans chaque tracé,
i) Le nombre de jambages dans chaque tracé.

Nous supposons que ces neuf primitives peuvent être correlées, et nous cherchons à réduire la matrice de variance-covariance sur la base d'apprentissage. Nous obtenons une nouvelle base dans laquelle on a des primitives décorrelées. La décision est prise avec un classifieur Bayésien en supposant que les mots sont équiprobables sachant qu'il existe 20 mots dans chacune des 236 classes. Nous calculons le produit de probabilité $(P(X_1 - X_9)/M_i$, M_i étant l'ensemble des mots. Deux des primitives donnent des coefficients très faibles et nous permettent d'avoir l'indépendance linéaire pour 7 primitives. Le nombre de différents vecteurs (avec les primitives) pour chaque classe est non connu. La distribution s'avère être globalement gaussienne, ce qui justifie l'indépendance sur tout l'ensemble des classes. Nous allons donc utiliser seulement 7 primitives, ce qui accélère le temps de traitement pour la phase de reconnaissance.

5. La Reconnaissance

La reconnaissance aboutit à une liste de propositions découlant de la coopération de deux modules : Le premier module analyse le mot dans son ensemble, c'est l'approche globale; le deuxième module est une approche analytique au niveau du graphème [Oli 95]. Le premier module reprend des primitives décrites dans le paragraphe 4, la décision est donnée par maximum de vraisemblance.

Après un apprentissage sur une base de 236 mots écrits par 30 scripteurs nous testons notre algorithme sur une base écrite par 10 autres scripteurs et nous obtenons le bon candidat en première position pour 60 % des cas et dans les cinq premiers candidats pour 85 % des cas.

Le deuxième module s'intéresse à l'étiquetage des observations par rapport aux caractères, ce qui permet l'utilisation de la syntaxe lettre au lieu de graphèmes. Nous supposons qu'un caractère α peut être segmenté au plus en trois graphèmes: α_p, α_q, α_r et que la limite de sous-segmentation est de deux caractères. Pour le nombre M de symboles observés nous obtenons 117 classes de graphèmes. Une deuxième réduction est réalisée avec la sélection des classes par un algorithme à seuil et une répartition des graphèmes par un algorithme de nuée dynamique modifié, d'où 87 classes de graphèmes et une observation indiquant le passage inter-tracé, soit M égal à 88. Les états de base sont les formes principales des caractères, nous avons N égal à 21 états. Nous déterminons les sous-codes générés par un caractère α suivi d'un caractère β. La base d'apprentissage contient 236 mots écrits par 15 scripteurs, nous faisons le test avec 5 autres scripteurs en calculant le taux de reconnaissance sur une moyenne de 4 tests. Nous obtenons le bon candidat en

première position dans 51 % des cas et dans les cinq premiers candidats pour 75.7 % des cas.

Les tableaux 1 et 2 montrent les résultats obtenus en variant le nombre de classes à reconnaître pour les alphabets de tailles différentes et nous les avons comparé aux résultats pris avec un superviseur (tableau 3) qui décidait la classe de graphème à reconnaître.

Nb classes	Top 1	Top 2	Top 3	Top 4	Top 5
60	57.66%	72.83%	75.67%	78.66%	80.37%
120	52%	67%	72%	76%	77%
180	49%	63%	70%	74%	76%
236	48%	57%	66%	70%	72%

Tableau 1. Résultats de classification pour un alphabet de 117 graphèmes.

Nb classes	Top 1	Top 2	Top 3	Top 4	Top 5
60	71.6%	81.3%	83.6%	84.6%	84.8%
120	63%	74.4%	78.6%	80.2%	81.6%
180	56.6%	69.6%	74.5%	77.6%	79%
236	51.1%	64.3%	71.1%	73.9%	75.7%

Tableau 2. Résultats de classification pour un alphabet de 87 graphèmes.

Nb classe	Top 1	Top 2	Top 3	Top 4	Top 5
60	84.5%	85.8%	86%	86.1%	86.1%
120	82%	83.7%	84.1%	84.4%	84.4%
180	81.5%	83.2%	83.6%	84.1%	84.2%
236	79%	81.6%	81.9%	82.1%	82.6%

Tableau 3. Résultats avec un superviseur.

6. Conclusion

Nos premiers résultats sont encourageants car la qualité de la base de données est très réaliste avec des erreurs et des différences d'écriture d'un pays à un autre. En perspective des travaux futurs nous pensons attribuer pour chaque observation un coefficient de confiance avec un coefficient de pondération qui sera l'entropie mutuelle d'observation dans la base d'apprentissage ou une mesure de discrimination de l'observation entre les classes. Ainsi chaque classe de mot candidat va accumuler un score et la décision se fera par maximum de vraisemblance.

REFERENCES BIBLIOGRAPHIQUES

[Ame 94] A. Ameur, K. Romeo, H. Miled, M. Cheriet Approche globale pour la reconnaissance des Mots Manuscrits Arabes. CNED'95, Rouen, France, pp 151-157, Juillet 1994.

[Badr 95] B. El-Badr and S.A. Mahmoud. A Survey and Bibliographie of Arabic optical text recognition. Signal Processing, Vol 41, pp 49-76, 1995.

[Beg 94] M. Mohammad Beglou, M.J.J. Holt, S.Datta. Unconstrained Letter-segmentation of Off-line Cursive Script Using Contour Information. Signal Processing VII: Theories and applications, Vol.3, pp 1437-1440, 1994.

[Chen 94] M.Y. Chen, A. Kundu, J. Zhou. Off-line handwritten word recognition using Hidden Markov Model type stochastic network, IEEE PAMI, Vol 16, No 5, pp 481-496, 1994

[Mil 96] H.Miled, C.Olivier, K.Romeo, Y.Lecourtier, M.Cheriet. Segmentation et Codage de Mots Manuscrits Arabes par une Approche Blackboard. CNED'96, Nantes, France, pp 185-192, Juillet 1996.

[Oli 95] C. Olivier, T. Paquet, M. Avila, Y.Lecourtier. Recognition of Handwritten Words using Stochastic Models. ICDAR'95, Montréal, Canada, Vol.1, pp 19-23, Aug. 1995.

[Oli 96] C. Olivier, H. Miled, K. Romeo, Y. Lecourtier. Segmentation and coding of Arabic Handwritten words, ICPR'96, Vol 3, pp 264-268, Vienna, Austria, Aug 1996

[Rom1 95] K. Romeo, A. Ameur, C. Olivier, Y.Lecourtier. Structural analysis of Arabic Handwriting: segmentation and recognition. Machine Vision and Application, Vol.8, pp 232-240, 1995.

[Rom2 95] K. Romeo, H. Miled, Y. Lecourtier. A New approach for Latin/Arabic Character Segmentation. ICDAR'95, Montréal, Canada, Vol.2, pp 874-877, Aug. 1995.