Modèles multi-flux pour la reconnaissance de mots manuscrits

Yousri KESSENTINI[1, 2], Thierry PAQUET[1], AbdelMajid BENHAMADOU[2]

[1] Laboratoire LITIS EA 4051, université de Rouen France

[2] Laboratoire MIRACL, université de Sfax Tunisie

{yousri.kessentini, thierry.paquet}@univ-rouen.fr

abdelmajid.benhamadou@isimsf.rnu.tn

Résumé – Nous présentons dans cet article une approche basée sur une modélisation par des modèles de Markov cachés (MMC) multi-flux pour la reconnaissance hors-ligne de mots manuscrits. Chaque mot est représenté par deux MMCs appris sur des données extraites respectivement des contours supérieur et inférieur de l'image du mot. La combinaison de ces deux sources d'informations est étudiée à travers les modèles multi-flux. Nous présentons les résultats de reconnaissance obtenus sur une base de mots isolés extraits de courriers acquis en environnement industriel.

Abstract *– We present in this paper a new approach based on multi-stream hidden Markov models (HMM) for the recognition of Off-Line handwriting. Every word is presented by two HMM models: the first one is learned with features extracted from upper contour, the second with features extracted from lower contour. The combination of these two sources of information is studied using the multi-stream framework. We present experiment results obtained on a database composed of isolated words extracted one industrial application.*

1. Introduction

Ces dernières années ont connu une explosion du nombre de documents papier générés par des activités administratives et économiques. Pour faciliter l'archivage, le traitement et le transfert de ces documents, des systèmes de Gestion Electronique de Documents (GED) ont été développés. Les documents papier sont alors numérisés et peuvent être stockés et transférés électroniquement. Dans ces conditions, le traitement automatique de documents, qui vise à automatiser la lecture des contenus, a connu un essor rapide. De nombreuses approches de reconnaissance sont en effet proposées dans la littérature pour tenter de résoudre ces problèmes. Parmi ces études quelques approches sont maintenant mises en oeuvre dans des applications industrielles de reconnaissance de chèques, de formulaires, ou d'adresses ou encore les assistants personnels (PDA) pour lesquels le clavier a été remplacé par un stylet et un moteur de reconnaissance de caractères.

Dans les systèmes de reconnaissance de l'écriture manuscrite, le module d'extraction de caractéristiques a une grande importance puisqu'il permet de représenter une image (matrice de pixels) par un ensemble de caractéristiques permettant d'identifier plus facilement ce dernier. On peut imaginer l'utilisation de différentes sources d'informations, ceci conduit donc à un espace de dimension élevée. Compte tenu de la quantité limitée des données d'entraînement, on suppose parfois que les différents vecteurs représentatifs de ces données sont indépendants, et on combine les paramètres des différentes sources en un seul vecteur. Cela peut conduire à une augmentation de la variance lors de l'estimation des paramètres des modèles et donc à une diminution des performances du système de reconnaissance.

Une autre solution, est de diviser l'espace des caractéristiques en plusieurs groupes. Chaque groupe est alors considéré indépendant des autres et la combinaison se fait dans l'espace des probabilités : l'idée est de combiner les sorties de différents classifieurs pour créer un système avec une fiabilité plus élevée. Plusieurs travaux ont été réalisés dans le domaine de la combinaison de classifieurs, en particulier les travaux de [6, 7] qui ont montré l'importance d'avoir des solutions robustes pour les problèmes de reconnaissance de l'écriture manuscrite en particulier.

Dans la littérature et plus précisément dans le domaine de la reconnaissance de la parole, des approches n'imposant pas le synchronisme parfait entre les différentes sources d'informations ont été proposées. Ces approches basées sur une modélisation par des modèles de Markov cachés HMM, sont désignées par les approches multi-flux.

Les avantages potentiels de ces approches sont multiples [3]:

- Elles offrent un moyen pour combiner différentes sources d'information.
- Cette combinaison peut être adaptative, certaines sources d'informations pouvant être sous-pondérées, voir rejetées si elles sont identifiées comme très peu fiable.
- La topologie des modèles de Markov cachés peut être adaptée à chaque source d'observation.
- Les différents flux peuvent se désynchroniser jusqu'à certains points lexicaux prédéfinis.

Malgré toutes ces possibilités, les modèles multi-flux n'ont pas été étudié, à notre connaissance, dans le domaine de la reconnaissance de l'écriture manuscrite. Dans cet article, nous proposons de combiner deux sources d'informations extraites respectivement sur les contours supérieurs et inférieurs des mots, à travers les modèles multi-flux.

Ce papier est organisé de la manière suivante : dans la section 2, nous décrivons l'approche multi-flux, la section 3, est consacrée à la présentation de notre système multi-flux pour la reconnaissance de mots manuscrits. Une évaluation expérimentale est donnée dans la section 4 présentant les résultats de reconnaissance obtenus sur une base de mots isolés extraits de lettre de réclamations provenant d'une application industrielle. Finalement dans la section 5, nous terminons par une conclusion et une discussion des perspectives.

2. L'approche multi-flux

L'approche multi-flux proposée dans [1][2] est une méthode adaptative permettant de combiner différentes sources d'information en utilisant des modèles de Markov coopératifs. Les modèles des différentes sources sont traitées indépendamment jusqu'à certains point d'ancrage (correspondants aux frontières des lettres ou des syllabes par exemple) où elles sont contraintes à se resynchroniser et à combiner leurs contributions partielles (voir figure 1).

Soit K le nombre de sources d'informations et soit M le modèle composé d'une séquence de J sous-modèles qui correspondent à des sous-unités lexicales M_j (j = 1, ..., J) (des lettres par exemple). Chaque sous-modèle M_j est composé de K modèles de Markov cachés (HMM) indépendants, notés M_j^k.

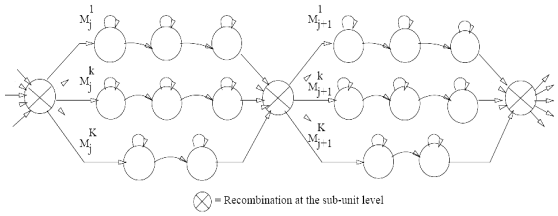


FIG.1: Structure générale d'un modèle multi-flux

La reconnaissance consiste à déterminer le modèle M dont la probabilité a posteriori est maximale, étant donnée la séquence d'observation X :

$$M^* = \text{argmax} \quad P(M \mid X)$$

La loi de Bayes nous donne alors :

$$M^* = \text{argmax} \quad \frac{P(X \mid M)\ P(M)}{P(X)}$$

P(X) étant indépendant du modèle M, le problème de la reconnaissance revient à déterminer le modèle M qui maximise le produit de la vraisemblance P(X | M) et de la probabilité du modèle P(M).

Dans notre cas, le modèle M est constitué d'une séquence de sous-unités lexicales elles mêmes composées de modèles de Markov cachés parallèles.

La vraisemblance P(X | M) est calculée de manière exacte comme la somme des vraisemblances associées à chacun des chemins C possibles dans les données.

$$P(X \mid M) = \sum_C P(X, C \mid M)$$

En décomposant pour tous les sous-modèles :

$$P(X, C \mid M) = \prod_{j=1}^{J} P(X_j, C_j \mid M_j)$$

Sur la base de l'approximation de Viterbi, on peut écrire :

$$P(X \mid M) \cong \max_{C_j} \prod_{j=1}^{J} P(X_j, C_j \mid M_j)$$

Par définition du modèle multi-flux, la vraisemblance $P(X_j, C_j \mid M_j)$ est alors calculée sur la base des vraisemblances associées à chacun des sources d'information :

$$P(X_j, C_j \mid M_j) = f(\{P(X_j^k, C_j^k \mid M_j^k), k = 1,..., K\})$$

avec X_j^k est la séquence de vecteurs d'observation associés au flux k, C_j^k est la sous-séquence d'états associés au flux k du sous-modèle M_j^k.

Pour une fonction de combinaison en somme pondérée de logarithme de vraisemblance, la vraisemblance de la sous-séquence X_j par rapport au modèle de sous-unité M_j est :

$$\log P(X_j \mid M_j) = \sum_{j=1}^{J} \sum_{k=1}^{K} w_j^k \log P(X_j^k \mid M_j^k)$$

ou w_j^k représente la fiabilité du flux k.

Le processus de décodage des modèles multi-flux revient donc à calculer la vraisemblance de la séquence de vecteurs d'observations des différentes sources d'informations. Ce calcul de vraisemblance fait appel à l'approximation de Viterbi : les vraisemblances ainsi obtenues pour les différents sous-modèles d'une même sous-unité sont ensuite combinées dans le but de fournir une estimation de la vraisemblance de la séquence de vecteurs étant donné le modèle M_j.

3. Système de reconnaissance

Le système de reconnaissance proposé est basé sur une modélisation par des modèles multi-flux (voir figure 2). L'entrée du système est une image d'un mot manuscrit. La première étape consiste à appliquer une série de prétraitements à l'image du mot. Ceci prépare le terrain à l'étape suivante d'extraction de caractéristiques. Chaque mot est alors représenté par deux ensembles de caractéristiques extraites respectivement sur les contours supérieur et inférieur. Ces deux ensembles serviront par la suite à l'apprentissage des deux modèles. L'apprentissage du modèle supérieur est indépendant de celui du modèle inférieur. Une fois les deux modèles appris, le décodage simultané des deux modèles est réalisé en s'accordant au formalisme multi-flux donné ci-dessus.

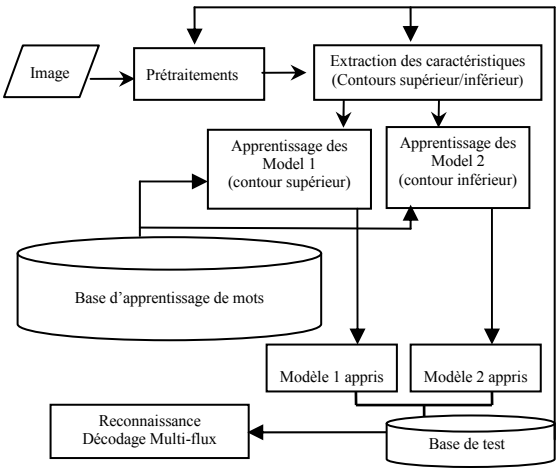
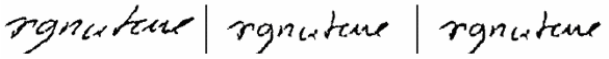


FIG. 2: Description du système de reconnaissance

3.1 Prétraitements

Avant de procéder à la reconnaissance de mot, il est nécessaire d'effectuer une série de prétraitements sur l'image de celui-ci en vue d'éliminer (ou du moins à réduire) les sources de variabilité ou de bruit, et également de simplifier la procédure d'extraction de caractéristiques. Nous appliquons une procédure de normalisation dans le but de ramener l'orientation du signal d'écriture à l'horizontale («slant») [5]. Cette normalisation s'effectue en trois étapes : une estimation de l'inclinaison est tout d'abord réalisée d'après le contour du mot. Ensuite l'image est corrigée par décalage des lignes de l'image (transformation par cisaillement, ou «sheering transform»). Comme on peut le voir sur la figure 3.b, cette transformation génère du bruit sur le contour de l'image. On applique alors une dernière étape de lissage du contour par une opération de morphologie mathématique : la fermeture. L'image finalement obtenue est présentée dans la figure 3.c.



a. image originale b. image normalisée c. image lissée

FIG. 3: Exemples de prétraitements

3.2 Extraction des caractéristiques

Les caractéristiques que nous avons choisies sont basées sur les points du contour du mot. Ces points sont séparés en un premier ensemble formant le contour supérieur et un deuxième formant le contour inférieur du mot (voir figure 3). Lors de l'extraction des caractéristiques, nous travaillons sur des fenêtres glissantes positionnées sur les contours. Une fenêtre est caractérisée par une largeur et un taux de recouvrement entre deux positions successives. Pour chaque position de la fenêtre, nous déterminons la direction de Freeman des points du contour supérieur (resp. inférieur), et un histogramme de directions de Freeman est calculé sur tous les points de la fenêtre, constituant ainsi un vecteur de 8 caractéristiques.

Un deuxième ensemble de caractéristiques consiste à déterminer pour chaque point du contour supérieur (resp. inférieur) un point d'intersection d'une sonde verticale descendante de ce point (resp. montante) avec l'image du mot. Ce point d'intersection est déterminé en descendant verticalement sur l'image du mot, en partant du point du contour considéré, tant que le pixel rencontré sur l'image du mot est noir, dès qu'il y a un pixel blanc on arrête et le point recherché sera le dernier pixel noir rencontré. Quatre cas sont alors possibles :

- Ce point appartient au contour inférieur (resp. supérieur) (points rouges sur la figure 4).
- Ce point appartient à une occlusion (points bleus sur la figure 4).
- Ce point appartient au contour supérieur (resp. inférieur) (points jaunes sur la figure 4).
- Pas de point, ce qui veut dire que le point du contour supérieur (resp. inférieur) est un point extrême (points verts sur la figure 4).

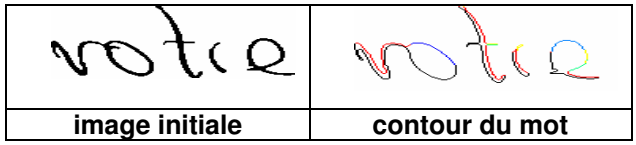


image initiale	contour du mot

FIG. 4: Différents types des points du contour supérieur

Ainsi nous distinguons 4 types de points du contour supérieur (resp. inférieur), un histogramme sera calculé pour chaque fenêtre d'où un deuxième vecteur de 4 caractéristiques. Ce deuxième ensemble de caractéristiques nous fournit des informations supplémentaires sur la correspondance entre les contours supérieur et inférieur du mot.

Nous disposons ainsi de deux séquences d'observations constituées chacune d'un vecteur de 12 caractéristiques continues. Chaque séquence sera modélisée par un modèle de caractère respectivement supérieur et inférieur.

3.3 Apprentissage

Chaque caractère est modélisé par un HMM et les modèles sont entraînés par maximisation de la vraisemblance grâce à l'algorithme de Baum-Welch. Le nombre d'états dans le modèle est le même pour chaque caractère. La topologie est de type gauche droite, n'autorisant que les transitions bouclantes et vers l'état suivant. Les probabilités d'émission sont calculées en utilisant un modèle de mélange de gaussiennes. L'apprentissage des deux modèles est effectué d'une manière indépendante. L'apprentissage est embarqué: tous les modèles de caractères sont entraînés simultanément sur l'ensemble des mots de la base d'apprentissage. Une fois appris, ces modèles pourront alors être concaténés de manière à obtenir le modèle correspondant à chaque mot du lexique. Nous disposons pour l'apprentissage d'une base de 4600 mots provenant de 500 scripteurs différents. Une base de validation de 500 mots est utilisée pour contrôler l'apprentissage. A la fin de la phase

d'apprentissage, nous disposons de deux modèles appris correspondant aux flux des contours supérieur et inférieur.

3.4 Reconnaissance

Pendant la phase de reconnaissance, les modèles de caractères sont concaténés pour former les modèles de mots. Le décodage des modèles 1D des contours supérieur et inférieur fait appel à l'algorithme de Viterbi [4]. Le décodage du modèle multi-flux est assuré par une variante de l'algorithme de Viterbi que nous avons développé ou le décodage des deux modèles se fait simultanément selon le formalisme multi-flux donné ci-dessus. Les vraisemblances des deux flux sont combinées en faisant appel à une fonction de combinaison en somme pondérée de logarithme de vraisemblance pour les états de recombinaison.

4. Expérimentations

L'évaluation du moteur de reconnaissance de mots est réalisée sur une base de 500 mots manuscrits extraits de courriers entrants provenant de 50 scripteurs différents.

Nous présentons les performances pour différentes tailles de lexique. La génération des lexiques est réalisée par tirage aléatoire des mots parmi un lexique de 1400 mots (lexique complet de la base de mots annotés). Pour chaque mot à reconnaître, un lexique de taille N est obtenu par un tirage aléatoire de N-1 mots parmi les 1400, complété par l'étiquette du mot à reconnaître.

Nous présentons les résultats de reconnaissance obtenus pour les modèles des contours supérieur et inférieur ainsi que les résultats obtenus par combinaison des décisions de ces deux modèles selon les règles de la somme et du produit. Ces résultats sont comparés à celles obtenus pour le modèle multi-flux.

Table 1: Résultats de reconnaissance

Modèle	Rang	Lexique 10	Lexique 100
Contour supérieur	1	83.8	63.4
	5	97.8	84.2
Contour Inférieur	1	76.6	51.4
	5	96.2	90.8
Multi-flux	1	89.4	70.8
	5	98.2	85
Somme	1	88.2	69
	5	97.6	84.2
Produit	1	89	69.8
	5	97.8	84.6

Les résultats de reconnaissance sont meilleurs pour le modèle du contour supérieur que celles du modèle du contour inférieur, cela s'explique par le fait que le contour supérieur est généralement plus informatif sur les lettres minuscules.

Les résultats de reconnaissance du modèle multi-flux sont les meilleurs. Ce pendant, le taux de reconnaissance n'est pas si élevé vu la mauvaise qualité d'écriture de certains mots de la base (voir figure 5.a) ou les fautes d'orthographe de certains autres (voir figure 5.b). Ces résultats peuvent être améliorés en utilisant des poids adaptés pour chaque modèle de caractère et en essayant d'autres règles de combinaison. Nous proposons aussi de combiner d'autres caractéristiques et de tester les modèles multi-flux asynchrones.

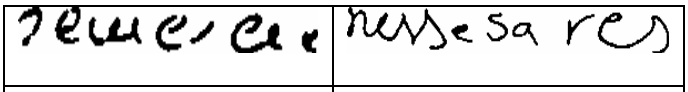


Image du mot « remercie »	Image du mot « nécessaires »

FIG. 5 : Exemples de la base de test

5. Conclusion

Nous avons présenté dans ce travail une première approche de reconnaissance basée sur une modélisation par des modèles de Markov cachés multi-flux.

Nous avons choisis des caractéristiques basées sur les points du contour. Ces points seront représentés par leur direction de Freeman : chaincode. Nous avons amélioré ces caractéristiques par de nouvelles caractéristiques originales présentant la correspondance entre le contour supérieur et inférieur. Les modèles multi-flux présentés, nous ont permis de combiner les caractéristiques extraites sur le contour supérieur et celles extraites sur le contour inférieur. Ce travail est prometteur et pourrait être poursuivi par les perspectives déjà énoncées.

Références

[1] C.J.Wellekens, J.Kangasharju, C.Milesi, "The use of meta-HMM in multistream HMM training for automatic speech recognition", *Proc. of Intl. Conference on Spoken Language Processing (Sydney)*, December 1998, pp. 2991-2994.

[2] H.Bourlard, S.Dupont. "Sub-band-based Speech Recognition". *In IEEE Int. Conf. on Acoust., Speech, and Signal Processing*, 1997, pages 1251-1254.

[3] H.Bourlard, S. Dupont, and C. Riss. "Multi-stream speech recognition". *Technical Report IDIAP-RR 96-07,* IDIAP, 1996.

[4] G.Forney, "The Viterbi algorithm*", Proc. IEEE 61 (3)*, 1973, pp. 268–278.

[5] F. Kimura, S. Tsuruoka, Y. Miyake & M. Shridhar, "A Lexicon Directed Algorithm for Recognition of Unconstrained Handwritten Words". *IEICE Trans*. Inf. &Syst., vol. E77-D, no. 7, 1994.

[6] S.Srihari, Reliability analysis of majority vote systems. *Information Sciences, 26: 243-256,* 1982.

[7] J.Hull, S.Srihari, and R.Choudhuri, An integrated algorithm for text recognition: comparison with a cascaded algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 5(4):384-395, 1983.*