

UNE NOUVELLE STRATÉGIE POUR L'AMÉLIORATION DES JEUX DE PRIMITIVES D'UN SYSTÈME DE RECONNAISSANCE DE L'ÉCRITURE

Frédéric Grandidier^{1,2,3}, Robert Sabourin^{1,2}, Ching Y. Suen¹ et Michel Gilloux³

¹CENPARMI, Concordia University,
Suite GM-606, 1455 de Maisonneuve Blvd West,
Montréal H3G 1M8, Canada

²LIVIA, Ecole de Technologie Supérieure,
1100 rue Notre Dame Ouest,
Montréal H3C 1K3, Canada

³RMO, Service de Recherche Technique de La Poste,
10, rue de l'Île Mâbon, BP86334,
44263 Nantes Cedex 02, France

Résumé : Dans cet article, nous introduisons une nouvelle stratégie permettant l'amélioration d'un jeu de primitives d'un système de reconnaissance de l'écriture manuscrite. L'idée de base repose sur la constatation que parmi un ensemble de primitives certaines sont plus discriminantes que d'autres. La technique d'amélioration proposée consiste à conserver les primitives les plus discriminantes et à substituer celles qui ne le sont pas. Dans un premier temps le pouvoir discriminant de chaque primitive doit être évalué, la perplexité conditionnelle est utilisée à cet effet. Ensuite l'identification de classes de primitives non-discriminantes permet de choisir la stratégie de descente d'un niveau perceptif. Un nouvel ensemble de primitives spécialisé pour chaque sous-problème de classification identifié par ces classes de primitives est ensuite défini. Leur regroupement avec les primitives discriminantes initiales permet de construire un nouveau système de reconnaissance amélioré. L'application de cette technique sur un système existant a montré sa pertinence.

Mots clés : reconnaissance hors-ligne de l'écriture manuscrite, évaluation et amélioration du pouvoir discriminant des primitives

1. INTRODUCTION

Dans le domaine de la reconnaissance de formes bi-dimensionnelles, la reconnaissance de l'écriture manuscrite se distingue par la grande variabilité intra et interclasses des formes à reconnaître. De ce fait les performances d'un système de reconnaissance automatique sont fortement dépendantes des informations extraites de l'image [1] ainsi que de leur intégration.

Le domaine de recherche de la reconnaissance de l'écriture étant très actif depuis plusieurs décennies, nous pouvons maintenant trouver dans la littérature un grand nombre de travaux portant sur la définition et l'évaluation de primitives, tant spécialisées pour la reconnaissance de caractères [1] que pour des formes bi-dimensionnelles plus générales [2]. Une fois l'extraction de primitives réalisée, le

système doit accomplir sa tâche de reconnaissance principalement, voire uniquement, à l'aide de cette information symbolique. De ce fait l'espace des primitives peut être considéré comme la perception que le système a de la forme à reconnaître. L'extraction de plusieurs primitives d'une même forme revient à avoir plusieurs perceptions de cette forme.

L'intégration de sources d'information multiples peut être réalisée différemment. Une première option possible est de créer un classificateur pour chaque ensemble de primitives, puis de combiner leurs scores de manière optimale [3,4]. Une autre stratégie consiste à réaliser la combinaison non plus au niveau des classificateurs mais à celui des primitives, en utilisant le produit Cartésien des ensembles associés à chaque source d'information [5-7]. Cette technique permet de construire un ensemble de primitives global utilisé par un seul classificateur.

Dans cette communication nous souhaitons introduire une alternative aux techniques précédemment citées. Nous proposons une nouvelle stratégie pour l'intégration de plusieurs sources d'information en vue d'améliorer le pouvoir discriminant des primitives et les performances du système de reconnaissance. Dans un premier temps nous allons introduire le formalisme de cette technique. Son application à un système existant [5] sera présentée dans la section 3, nous discuterons alors des résultats obtenus. La dernière section permettra de présenter nos conclusions ainsi que nos travaux futurs.

2. LA NOUVELLE STRATÉGIE D'AMÉLIORATION D'UN ENSEMBLE DE PRIMITIVES

Notre réflexion est basée sur les constatations obtenues lors d'une analyse approfondie d'un système de reconnaissance de l'écriture manuscrite [6]. La conclusion principale de cette étude est que la longueur des exemples en termes de caractères a une influence sur les taux de reconnaissance. En effet les mots longs sont plus faciles à reconnaître du fait que la séquence de primitives extraite est plus longue. Il y a donc plus de chance qu'elle contienne les primitives les plus discriminantes, qui servent alors de point d'ancrage pour la reconnaissance. Cela signifie que les performances du système sont dépendantes du pouvoir discriminant de chacune des primitives.

L'idée de base de notre nouvelle stratégie est d'améliorer l'ensemble de primitives utilisé par le système de reconnaissance en conservant les plus discriminantes et en remplaçant les autres par un ou plusieurs nouveaux ensembles de primitives, spécialisés en fonction d'un sous-problème à résoudre.

2.1 L'évaluation du pouvoir discriminant des primitives

La première étape de notre processus est bien sûr l'évaluation du pouvoir discriminant individuel de chaque primitive. Nous avons choisi d'utiliser l'indicateur proposé par El-Yacoubi : la perplexité conditionnelle [7]. Cet indicateur est basé sur la notion d'entropie de la théorie de l'information. L'entropie a été introduite dans le domaine de la reconnaissance de la parole par Bahl [8] pour mesurer *a priori* la difficulté de la tâche de reconnaissance. Elle est définie de la manière suivante :

$$H = - \sum_m p(m) \cdot \log p(m) \quad (1)$$

où $p(m)$ est la probabilité *a priori* du mot m et la somme est calculée sur l'ensemble des mots du vocabulaire relatif à l'application considérée.

Dans [7], l'auteur définit l'entropie conditionnelle des classes C considérées lors de la modélisation étant donnée une primitive f_j par :

$$H(C|F=f_j) = -\sum_{i=1}^{N_c} p(c_i|f_j) \cdot \log p(c_i|f_j) \quad (2)$$

où c_i est une des classes considérées lors de la modélisation et N_c le nombre de ces classes. La fonction $H(C|F=f_j)$ permet de quantifier la capacité d'une primitive f_j à faire la discrimination entre les classes c_i . Elle atteint son maximum $\log N_c$ lorsque :

$$p(c_i|f_j) = \frac{1}{N_c}, \quad \forall i \quad (3)$$

Dans ce cas particulier la primitive f_j n'apporte au système de reconnaissance aucune information pour effectuer la discrimination entre les N_c classes. La fonction atteint son minimum 0 lorsqu'il existe une classe c_i telle que :

$$p(c_i|f_j) = 1 \quad \text{et} \quad p(c_k|f_j) = 0, \quad \forall k \neq i \quad (4)$$

La perplexité PP est liée à l'entropie H par la relation suivante :

$$PP = 2^H \quad (5)$$

L'avantage d'utiliser la perplexité plutôt que l'entropie est qu'elle varie entre 1 et N_c , elle peut donc être directement reliée au nombre de classes mises en jeu.

Le principal intérêt d'utiliser l'entropie conditionnelle est que cette mesure permet d'évaluer de manière objective le pouvoir discriminant des primitives sans faire appel à la phase de reconnaissance. De manière à obtenir un indicateur global du pouvoir discriminant d'un ensemble de primitives, l'entropie conditionnelle des classes étant données les primitives peut être calculée à l'aide de la probabilité *a priori* des primitives $p(f_j)$ de la manière suivante :

$$H(C|F) = \sum_{j=1}^{N_f} p(f_j) \cdot H(C|F=f_j) \quad (6)$$

où N_f est le nombre de primitives de l'ensemble évalué.

Pour un ensemble de primitives donné E_i , nous pouvons maintenant ordonner ses éléments en fonction de leur perplexité conditionnelle. Plus cette valeur est grande, moins la primitive est discriminante. Un ensemble ordonné sera noté (E_i, \succ) . En déterminant un seuil de perplexité t , l'ensemble de primitives peut être divisé en 2 sous-ensembles : D_i contenant les primitives jugées suffisamment discriminantes et \overline{D}_i contenant les non-discriminantes.

2.2 La descente d'un niveau perceptif

La deuxième étape de la technique proposée consiste à remplacer les primitives jugées non-discriminantes par d'autres. A cet effet, un nouvel espace de représentation va être utilisé afin de re-caractériser les graphèmes regroupés en classes par les primitives jugées non-discriminantes. Dans notre terminologie, cette étape est appelée : *descente d'un niveau perceptif*. Les primitives étant la perception de la forme pour le système de reconnaissance, l'extraction d'une nouvelle

information basée sur une première représentation est considérée comme un changement de niveau perceptif. Du fait de l'approche proposée les nouvelles primitives extraites à un niveau intègrent implicitement l'information apportée par celles du niveau supérieur.

Deux approches distinctes sont envisagées pour réaliser la descente d'un niveau perceptif, dépendant de la nature des primitives à remplacer. Nous proposons de les regrouper en classes : une *classe de primitives* CE_i d'un ensemble donné E_i , est un sous-ensemble de primitives possédant au moins une propriété de base commune. Pour un ensemble construit sur les dépassements et les boucles, comme celui de notre système [5], une classe de primitives peut être définie par exemple, comme le sous-ensemble de celles possédant un dépassement bas. Nous pouvons également utiliser un indicateur de la performance des primitives, comme la perplexité conditionnelle, afin de les regrouper en classes. Elles seront alors définies par un intervalle de valeurs dans le domaine des perplexités.

Dans le cas où toutes les primitives non-discriminantes sont regroupées dans une seule classe, la descente d'un niveau perceptif se fait à l'aide d'un seul ensemble de primitives. Par contre si plusieurs classes de primitives ont été identifiées, un nouvel ensemble est utilisé pour chacune de ces classes, comme présenté sur la (fig. 1). Dans ce deuxième cas, chaque nouvel ensemble de primitives E_i va être spécialisé en fonction des propriétés de la classe de primitives détectées au niveau supérieur.

Durant la phase de reconnaissance, pour chaque graphème, le système va extraire une primitive à partir de l'ensemble de départ E_1 . Si cette dernière fait partie du sous-ensemble \bar{D}_1 , alors une nouvelle primitive sera extraite en fonction de l'appartenance de la première à une des classes de primitives identifiées au cours de l'apprentissage.

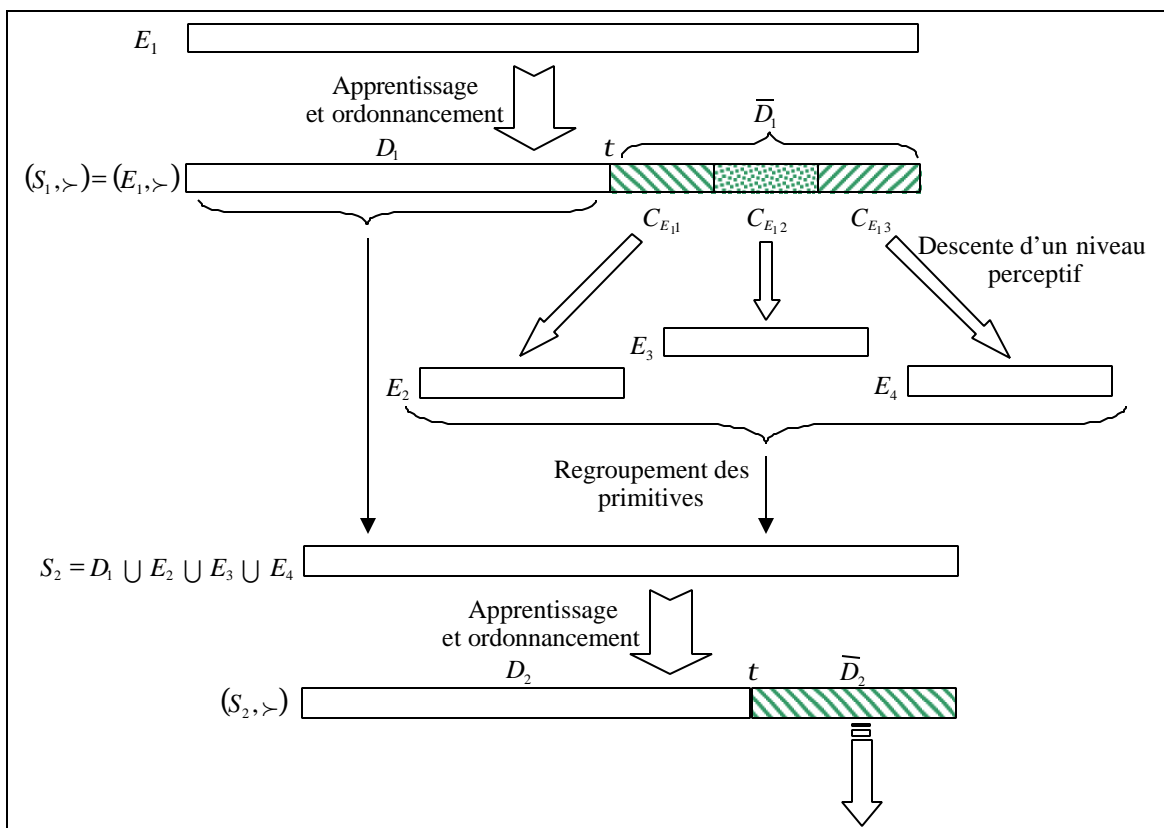


Figure 1 Synopsis de la nouvelle stratégie pour l'amélioration d'un ensemble de primitives E_1 .

L'étape suivante du processus consiste à regrouper les primitives discriminantes du niveau supérieur D_i avec les nouvelles, de manière à former l'ensemble de primitives de ce niveau perceptif S_{i+1} . A ce moment un nouvel apprentissage peut être réalisé, suivi de l'évaluation du pouvoir discriminant de ce nouvel ensemble de primitives. La comparaison des valeurs de perplexité conditionnelle donnera une indication sur l'amélioration du système, sans faire appel à la phase de reconnaissance. L'ordonnement du nouveau jeu de primitives peut également être réalisé et le processus entier ré-itéré.

Lors de la phase d'apprentissage le seuil t doit être déterminé, les classes de primitives identifiées et les nouveaux ensembles de primitives définis. De plus un critère d'arrêt du processus itératif doit être choisi.

3. L'AMÉLIORATION D'UN SYSTÈME DE RECONNAISSANCE EXISTANT

De manière à évaluer la pertinence de la stratégie proposée dans la section précédente, nous l'avons testée dans le but d'améliorer les performances du système présenté dans [5]. Il s'agit d'un système de reconnaissance de l'écriture manuscrite hors-ligne utilisant une modélisation markovienne discrète au niveau du caractère et une segmentation explicite. Après les phases de pré-traitement et de segmentation, chaque graphème est caractérisé par 2 symboles provenant de 2 ensembles de primitives distincts. Le premier E_1 (27 symboles) est basé sur les primitives globales que sont les dépassements hauts, les dépassements bas et les boucles (tab. 3). Il est dédié principalement à l'écriture cursive. Afin de compléter cette représentation un second jeu E_2 (14 symboles) est utilisé. Les primitives sont obtenues par analyse des histogrammes de transitions horizontal et vertical. L'apprentissage du système est réalisé à l'aide de 2 corpus de données : celui d'apprentissage (12022 noms de ville) et celui de validation (3475 noms de ville). Pour le test un corpus de 4674 noms de ville est utilisé. Les performances obtenues sont présentées ci-dessous (tab. 2) lorsqu'un seul jeu de primitives ou la combinaison des 2 est utilisé.

Ensemble de primitives	Nombre de primitives	Perplexité globale	Taille du lexique		
			10	100	1000
E_1	27	38,29	96,64%	88,29%	71,58%
E_2	14	35,23	97,50%	91,16%	78,01%
$E_1 \times E_2$	378	20,72	98,69%	95,42%	86,82%

Tableau 2 Performances du système standard.

Nous avons choisi d'appliquer la nouvelle stratégie d'amélioration sur l'ensemble E_1 , premièrement parce que son pouvoir discriminant global est plus faible que celui de E_2 et deuxièmement parce qu'après l'apprentissage plus de 50% des graphèmes sont caractérisés par la même primitive de cet ensemble : "-", correspondant à l'absence de dépassements et de boucles.

3.1 L'évaluation du pouvoir discriminant des primitives

La première étape est l'évaluation du pouvoir discriminant individuel de l'ensemble de primitives E_1 . Pour cela nous devons disposer des probabilités $p(c_i|f_j)$ et $p(f_j)$ (équations 2 et 6). Le nombre de classes N_C pour la modélisation markovienne dépend de l'architecture utilisée. Dans notre système, les observations étant émises par les transitions, le nombre de classes est équivalent au nombre de transitions pouvant émettre un symbole caractérisant le graphème. Chaque modèle

caractère en contient 4, correspondant soit à une lettre entière, soit à son premier, second ou troisième segment. Cependant le concept d'états liés est utilisé afin de partager entre la majorité des modèles (tous sauf ceux associés aux lettres m, w, M, W) la transition relative au troisième segment. En plus des lettres et des chiffres notre modélisation prend en compte certains symboles propres à l'application visée, ce qui conduit à un total de 68 modèles. Le nombre de classes à considérer est alors de $3 \times 68 + 5 = 209$. A l'aide de l'algorithme de Viterbi et de l'étiquetage correct des exemples, il est possible de récupérer l'alignement entre les couples graphème/primitive et les classes de notre modélisation [7]. Les probabilités sont alors obtenues à partir des fréquences d'occurrence. Nous présentons partiellement ci-dessous (tab. 3) les primitives de l'ensemble E_1 triées par ordre croissant de perplexité conditionnelle. Les primitives sont représentées par un symbole arbitraire et définies à l'aide d'une matrice de propriétés de base. Il est à noter que l'évaluation de la taille d'une boucle n'est réalisée que si elle se situe dans la zone médiane. De plus dans ce cas, sa position relative par rapport aux dépassements est prise en compte. L'évaluation des perplexités conditionnelles est réalisée sur le corpus de validation, de manière à utiliser les données du corpus d'apprentissage uniquement pour l'ajustement des paramètres du système.

Primitives	<i>l</i>	<i>g</i>	<i>G</i>	...	<i>S</i>	<i>D</i>	<i>d</i>	<i>b</i>	<i>T</i>	<i>y</i>	<i>f</i>	<i>B</i>	<i>t</i>	<i>H</i>	<i>h</i>	-
Dépassement haut	p			...	G	G	p	p	G		G		p	G	p	
Dépassement bas		p	G	...						p	G	G	G			
Boucle supérieure	x			...												
Boucle médiane		G	G	...	Gd	pg	pg	pd	pd							
Boucle inférieure		x	x	...												
Occurrence (%)	0,01	0,01	0,1	...	0,9	0,8	1,3	1,1	0,5	0,3	0,4	1,3	0,4	11,8	5,8	51,9
Perplexité	1,9	3,5	4,7	...	18,5	20,2	20,9	23	23,6	28,5	30,3	32,7	41	44,3	49,6	61,1

Tableau 3 Ordonnement des primitives de l'ensemble E_1 en fonction de leurs valeurs de perplexité conditionnelle (p : petit, G : grand, x : présence, g : gauche, d : droite).

Les valeurs de perplexité conditionnelle des primitives les plus discriminantes confirme l'intérêt de conserver ces dernières. Une analyse approfondie nous a permis de remarquer que les 7 primitives les moins discriminantes (valeur > 24) sont les seules ne possédant pas de boucles. De ce fait elles appartiennent à une classe de primitives de même nature définie par : "*primitives sans boucle*".

3.2 L'espace de représentation des concavités

Pour cette première étude, nous avons choisi d'utiliser les concavités comme espace de représentation au second niveau perceptif. Ce choix nous a été dicté par une analyse visuelle des graphèmes caractérisés par la primitive la plus fréquente (51,9%) de l'ensemble E_1 : "-". En effet les graphèmes regroupés par cette primitive montrent une grande diversité de forme. En plus de ceux provenant des caractères sans boucle ni dépassement, nous avons remarqué la présence de graphèmes appartenant à des caractères dont les boucles sont mal fermées. Dans le cas des exemples bâtons, les zones d'écriture supérieure et inférieure sont souvent inexistantes, ce qui conduit la plupart des graphèmes associés à être caractérisés par la primitive "-". Suite à ces constatations, l'analyse de concavités a été choisie car elle semble adaptée à ce sous-problème de reconnaissance de formes.

La technique d'étiquetage des pixels blancs de l'image [9] a été utilisée. L'analyse est réalisée sur les rectangles englobant des graphèmes. Uniquement 39 configurations distinctes sont rencontrées sur la base d'apprentissage. Si une

occlusion est rencontrée, une analyse approfondie permet de chercher une sortie en empruntant 2 directions consécutives ; quatre configurations ont été rajoutées à cet effet. Le pourcentage de pixels noirs de la zone d'analyse est également pris en compte. Finalement le vecteur de caractéristiques contient 44 composantes qui sont des valeurs réelles variant entre 0 et 1.

Notre système étant basé sur une représentation discrète de l'information, une quantification vectorielle doit être utilisée de manière à discrétiser l'espace des concavités. A cet effet l'algorithme LBG [10] a été choisi pour sa simplicité et sa rapidité d'exécution.

3.3 La descente d'un niveau perceptif avec une seule classe de primitives

Afin d'appliquer la nouvelle stratégie, l'ensemble de primitives E_1 doit être divisé en 2 sous-ensembles D_1 et \overline{D}_1 . La valeur du seuil t a été fixée à 24 pour 2 raisons. Premièrement ce seuil correspond à un saut dans les valeurs de perplexité et deuxièmement il permet de regrouper toutes les primitives de \overline{D}_1 au sein d'une classe unique définie par : "*primitives sans boucle*". Nous pouvons remarquer que la fréquence d'occurrence totale de ce sous-ensemble de primitives dépasse les 70%.

Le vecteur de concavités a été extrait pour l'ensemble des graphèmes caractérisés au premier niveau par une des primitives non-discriminantes de \overline{D}_1 (141907 graphèmes). L'algorithme de quantification vectorielle a été appliqué de manière à obtenir 64 puis 128 centres, conduisant ainsi à 2 nouveaux ensembles de primitives distincts. Chacun d'eux a permis de réaliser une expérience où il est utilisé afin de substituer les primitives de \overline{D}_1 . Le regroupement de ces primitives de concavité avec celles de D_1 permet d'obtenir l'ensemble de primitives du second niveau perceptif et ainsi de réaliser l'apprentissage et le test d'un nouveau système. Les performances obtenues au cours de ces 2 expériences sont présentées ci-dessous (tab. 4).

Ensemble de primitives	Nombre de primitives	Perplexité globale	Taille du lexique		
			10	100	1000
E_1	27	38,29	96,64%	88,29%	71,58%
$S_2^1 = D_1 + 64$	84	21,12	98,48%	94,61%	85,28%
$S_2^2 = D_1 + 128$	148	18,08	98,72%	94,93%	86,05%
$E_1 \times E_2$	378	20,72	98,69%	95,42%	86,82%

Tableau 4 Performances obtenues lors de la descente d'un niveau perceptif avec une seule classe de primitives.

Nous pouvons constater une augmentation significative du pouvoir discriminant global des nouveaux jeux de primitives, ainsi qu'un gain de 14,5% au niveau du taux de reconnaissance, lors de l'utilisation d'un lexique de taille 1000. De plus ces performances ont été obtenues avec moins de primitives que pour la combinaison des ensembles E_1 et E_2 , ce qui signifie moins de paramètres à estimer pour notre système. Nous pouvons conclure de la pertinence de notre nouvelle stratégie d'amélioration d'un ensemble de primitives.

3.4 La descente d'un niveau perceptif avec plusieurs classes de primitives

Afin d'évaluer les 2 alternatives de descente d'un niveau perceptif, nous avons décomposé le sous-ensemble \overline{D}_1 en plusieurs classes de primitives de même nature :

- " - " : primitives ne possédant pas de dépassement ni de boucle,

- “**hH**” : primitives possédant un petit ou un grand dépassement haut seulement,
- “**tf**” : primitives possédant un grand dépassement bas et un petit ou un grand dépassement haut,
- “**By**” : primitives possédant un petit ou un grand dépassement bas seulement.

Pour chacune de ces classes, plusieurs jeux de primitives de concavité ont été construits en augmentant le nombre de centres au cours de la quantification vectorielle. Chacun d’eux a permis d’effectuer la descente d’un niveau perceptif avec une seule classe de primitives comme décrit dans la sous-section précédente. Après avoir réalisé l’apprentissage de chaque nouveau système, le pouvoir discriminant global de l’ensemble de primitives du second niveau perceptif a été calculé. Cet indicateur est utilisé pour évaluer l’amélioration du système car le temps de calcul nécessaire est beaucoup plus court que pour l’évaluation des taux de reconnaissance et parce qu’une relation existe entre ces 2 fonctions [8].

Classe de primitives	Nombre d’échantillons	Nombre de primitives de concavité ajoutées				
		4	8	16	32	64
“-”	77 898	34,07	30,28	27,42	24,41	22,30
“ hH ”	25 660	37,24	36,83	36,62	35,81	
“ tf ”	1 644	38,14	38,07	38,05	37,99	
“ By ”	2 144	38,12	38,06	37,99	37,86	

Tableau 5 Evaluation du pouvoir discriminant de différents jeux de primitives améliorés.

Nous pouvons remarquer dans (tab. 5) que la perplexité conditionnelle globale décroît toujours avec l’augmentation du nombre de primitives injectées. Cette remarque est particulièrement vraie pour la classe “-”. Les raisons sont premièrement sa fréquence d’occurrence élevée et deuxièmement la grande variation de formes des graphèmes associés. Pour ces raisons nous avons décidé de porter une attention particulière à cette classe lors de la descente d’un niveau perceptif avec plusieurs classes de primitives. Une série d’expérience a été réalisée, où les classes de primitives identifiées dans \overline{D}_1 ont été remplacées par le nombre de primitives de concavités marqué en gras dans (tab. 5) (*i.e.* 64 primitives en tout) et la classe “-” par un nombre variant de 8 à 128. Plusieurs systèmes ont ainsi été construits et testés, les résultats sont présentés ci-dessous (tab. 6).

Ensemble de primitives	Nombre de primitives	Perplexité globale	Taille du lexique		
			10	100	1000
E_1	27	38,29	96,64%	88,29%	71,58%
$S_2^3 = D_1 + 64 + 8$	92	24,57	98,40%	93,67%	83,20%
$S_2^4 = D_1 + 64 + 16$	100	22,26	98,63%	94,76%	85,17%
$S_2^5 = D_1 + 64 + 32$	116	19,82	98,74%	94,76%	85,67%
$S_2^6 = D_1 + 64 + 64$	148	18,08	98,78%	95,61%	86,56%
$S_2^7 = D_1 + 64 + 128$	212	16,11	98,65%	95,38%	87,51%
$E_1 \times E_2$	378	20,72	98,69%	95,42%	86,82%

Tableau 6 Performances obtenues lors de la descente d’un niveau perceptif avec plusieurs classes de primitives.

L'augmentation du nombre de primitives servant à remplacer "-" conduit à une augmentation significative des taux de reconnaissance et du pouvoir discriminant de l'ensemble de primitives au second niveau perceptif. Dans le cas de l'utilisation d'un lexique de taille 1000 nous pouvons constater une amélioration d'environ 16% pour l'ensemble S_2^7 . Cette performance dépasse celle obtenue par le système utilisant la combinaison $E_1 \times E_2$. De plus le nombre de primitives est de 44% moindre. La comparaison des performances des tableaux (tab. 4) et (tab. 6) permet de conclure que la descente d'un niveau perceptif avec plusieurs classes de primitives est plus intéressante. Dans ce cas, chaque nouvel ensemble de primitives utilisé pour substituer les primitives non-discriminantes est spécialisé pour un sous-problème de classification identifié par la classe de primitives de départ et ses propriétés.

3.5 La combinaison des ensembles améliorés avec E_2

Afin d'évaluer l'amélioration globale de notre système, nous avons décidé de combiner les meilleurs jeux améliorés (S_2^2 et S_2^7) avec l'ensemble de primitive E_2 , à l'aide de la technique utilisé par le système standard [5]. Deux nouveaux systèmes ont été construits et testés.

Ensemble de primitives	Nombre de primitives	Perplexité globale	Taille du lexique		
			10	100	1000
$E_1 \times E_2$	378	20,72	98,69%	95,42%	86,82%
$S_2^2 \times E_2$	2 072	10,44	98,91%	95,31%	87,44%
$S_2^7 \times E_2$	2 968	9,22	98,80%	95,81%	88,68%

Tableau 7 Performances obtenues en combinant les jeux de primitives améliorés avec E_2 .

Les résultats présentés dans le tableau ci-dessus (tab. 7) montre que par rapport aux jeux améliorés (S_2^2 et S_2^7), la combinaison conduit à une légère augmentation des performances mais à un accroissement important du nombre de primitives et de ce fait du nombre de paramètres du système. Cette technique conduit à distribuer uniformément l'ensemble de primitives E_2 sur l'ensemble S_2^i sans spécialisation pour un sous-problème de classification. Nous pouvons conclure que la combinaison des ensembles de primitives conduit à une saturation des performances du système, combinée à une augmentation considérable du nombre de paramètres à estimer. Le danger de cette technique est que le système risque de tomber dans une phase de sur-apprentissage. Pour cette raison la nouvelle stratégie introduite dans cet article est plus intéressante que la combinaison des ensembles de primitives utilisée ici.

4. CONCLUSIONS ET TRAVAUX FUTURS

Nous avons introduit une nouvelle stratégie permettant l'amélioration d'un jeu de primitives pour un système de reconnaissance de l'écriture basé sur les MMC. L'idée de base est de conserver les primitives les plus discriminantes et de remplacer les autres. Les résultats expérimentaux nous permettent de conclure que cette nouvelle technique permet d'augmenter de manière significative les performances du système tout en conservant un nombre de paramètres pas trop élevé. En fait le nombre de paramètres augmente de manière additive contrairement à la combinaison des ensembles de primitives [5] où elle est multiplicative. C'est un des points forts de cette nouvelle stratégie. Parmi les 2 techniques de descente d'un niveau perceptif proposée, celle utilisant plusieurs classes de primitives semble plus pertinente. Dans ce cas, à chaque niveau perceptif, les jeux de primitives intégrés

peuvent être spécialisés en fonction d'un sous-problème de classification identifié par une classe de primitives non-discriminantes.

De manière à évaluer le plein potentiel de cette stratégie, nous devons développer d'autres espaces de représentation. Le choix des facteurs de formes sera guidé par les sous-problèmes de classification identifiés par les classes de primitives au premier niveau perceptif. Nous désirons également définir des critères objectifs pour choisir la valeur du seuil t et pour stopper le processus itératif de changement de niveau perceptif. L'algorithme de quantification vectorielle peut également être amélioré de manière à choisir un nombre de primitives optimal à chaque niveau perceptif et ainsi éviter une augmentation trop importante du nombre de paramètres.

Remerciements : Ce travail a été supporté par le Service de Recherche Technique de La Poste à Nantes, France, par l'Ecole de Technologie Supérieure et le Center for Pattern Recognition and Machine Intelligence à Montréal, Canada.

RÉFÉRENCE

- [1] Trier, O.D., Jain, A.K. et Taxt, T., "Feature Extraction Methods for Character Recognition - A Survey", *Pattern Recognition*, vol 29, 1996, pp. 641-662.
- [2] Loncaric, S., "A Survey of Shape Analysis Techniques", *Pattern Recognition*, vol 31, 1998, pp. 983-1001.
- [3] Gader, P.D., Mohamed, M.A. et Keller, J.M., "Fusion of Handwritten Word Classifiers", *Pattern Recognition Letters*, vol 17, 1996, pp. 577-584.
- [4] Kittler, J., Hatef, M., Duin, R.P.W. et Matas, J., "On Combining Classifiers", *IEEE Trans. on PAMI*, vol 20, 1998, pp. 226-239.
- [5] El-Yacoubi, A., Gilloux, M., Sabourin, R. et Suen, C.Y., "An HMM-Based Approach for Off-Line Unconstrained Handwritten Word Modeling and Recognition", *IEEE Trans. on PAMI*, vol 21, 1999, pp. 752-760.
- [6] Grandidier, F., Sabourin, R., El-Yacoubi, A., Gilloux, M. et Suen, C.Y., "Influence of Word Length on Handwriting Recognition", *Proc. 5th Int. Conf. on Document Analysis and Recognition*, Bangalore, India, Sep. 1999, pp. 777-780.
- [7] El-Yacoubi, A., *Modélisation Markovienne de l'Écriture Manuscrite. Application à la reconnaissance des Adresses Postales*, Thèse de doctorat, Université de Rennes I, 1996, 307 pages.
- [8] Bahl, L.R., Jelinek, F. et Mercer, R.L., "A Maximum Likelihood Approach to Continuous Speech Recognition", *IEEE Trans. on PAMI*, vol 5, 1983, pp. 179-190.
- [9] Heutte, L., *Reconnaissance de Caractères Manuscrits : Application à la Lecture Automatique des Chèques et des Enveloppes Postales*, Thèse de doctorat, Université de Rouen, 1994, 220 pages.
- [10] Linde, Y., Buzo, A. et Gray, R.M., "An Algorithm for Vector Quantizer Design", *IEEE Trans. on Communication*, vol 28, 1980, pp. 84-95.