



HAL
open science

Modèles graphiques pour la combinaison de descripteurs : application à la reconnaissance de symboles

Sabine Barrat, Salvatore Tabbone

► **To cite this version:**

Sabine Barrat, Salvatore Tabbone. Modèles graphiques pour la combinaison de descripteurs : application à la reconnaissance de symboles. 16e congrès francophone AFRIF-AFIA Reconnaissance de Formes et Intelligence Artificielle - RFIA08, AFRIF / AFIA, Jan 2008, Amiens, France. inria-00232594

HAL Id: inria-00232594

<https://inria.hal.science/inria-00232594>

Submitted on 1 Feb 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Modèles graphiques pour la combinaison de descripteurs : application à la reconnaissance de symboles

Graphical models for combining descriptors : application to symbol recognition

S. Barrat

S. Tabbone

LORIA-UMR 7503
Université de Nancy 2
BP 239, 54506 Vandoeuvre-les-Nancy

email : {barrat,tabbone@loria.fr}

Résumé

Nous présentons dans cet article une adaptation originale des réseaux Bayésiens au problème de la reconnaissance de formes. Plus précisément, nous proposons une méthode de combinaison de descripteurs qui permet d'améliorer significativement le taux de reconnaissance, par rapport aux taux obtenus par chaque descripteur pris individuellement. Pour ce faire, un classificateur est défini en utilisant un modèle de mixtures de Gaussiennes, représenté par un modèle graphique. Les modèles graphiques sont un mode de représentation simple et intuitif des modèles de mixtures de Gaussiennes. Afin de pallier le problème de dimensionnalité, nous utilisons une méthode de sélection de variables. Les résultats expérimentaux, obtenus dans un contexte d'apprentissage supervisé et testés sur différentes bases, sont très prometteurs.

Mots Clefs

Reconnaissance de symboles, modèles graphiques probabilistes, réseaux Bayésiens, sélection de variables.

Abstract

We present in this paper an original adaptation of Bayesian networks to shape recognition problem. More precisely, we propose a descriptor combination method which enables to improve significantly the recognition rate compared to the recognition rates obtained by each descriptor. In this perspective, we define a classifier which uses a Gaussian mixture model, represented by a probabilistic graphical model. Probabilistic graphical models are a simple and intuitive way of Gaussian mixture model representation. In order to solve the dimensionality problem, we use a variable selection me-

thod. Experimental results, obtained in a supervised learning context and tested on different databases, are promising.

Keywords

Symbol recognition, probabilistic graphical models, Bayesian networks, variable selection.

1 Introduction

La classification est une tâche courante en analyse de données et en reconnaissance de formes. Cette tâche requiert un classificateur, i.e une fonction qui assigne une classe à des observations décrites par un ensemble de caractéristiques. L'induction de classificateurs à partir d'un ensemble de données étiquetées (on parle d'apprentissage supervisé) est un problème central en apprentissage. En effet, dans de nombreuses applications, l'objectif est d'assigner un vecteur de caractéristiques $f = \{f_1, f_2, \dots, f_n\}$ à une classe c_i parmi k classes, désignées par le vecteur $c = \{c_1, c_2, \dots, c_k\}$. De nombreuses approches de ce problème sont basées sur des représentations fonctionnelles variées, telles que les arbres de décision, les réseaux de neurones [1], des règles de décision, ... Les approches probabilistes jouent elles aussi un rôle très important en classification. Un moyen de réaliser cet objectif est de calculer la distribution de probabilité conditionnelle $P(c_i|f), i \in \{1, 2, \dots, k\}$ et d'affecter l'observation f à la classe i pour laquelle ladite probabilité est maximale. On peut bien sûr formuler et résoudre des modèles probabilistes complexes uniquement par manipulations algébriques. Cependant, on gagne à représenter les distributions de probabilités sous forme de diagrammes, appelés modèles graphiques proba-

bilistes. En effet, ce mode de représentation est un moyen simple de visualiser la structure et les propriétés (incluant les propriétés de dépendance conditionnelle) d'un modèle probabiliste, mais surtout il permet d'effectuer des calculs complexes, comme l'apprentissage de distributions de probabilités et l'inférence à l'aide de manipulations graphiques. La représentation graphique est d'autant plus utile que le modèle probabiliste est complexe.

Dans ce papier nous proposons donc une méthode originale de combinaison de descripteurs appliquée à la classification de formes. Nous avons adapté la théorie des modèles graphiques à la reconnaissance de formes. De plus, notre approche se distingue par l'utilisation de la méthode de sélection de variables LASSO [16] pour pallier les problèmes de dimensionnalité liés à la taille des vecteurs caractéristiques et de complexité du réseau sous-jacent.

Un bref rappel théorique sur les classificateurs Bayésiens est fait en section 2. Ensuite, section 3, nous présentons notre méthode, basée sur un modèle de mélanges de Gaussiennes représenté par un modèle graphique. Les adaptations faites pour appliquer notre méthode au problème de la reconnaissance de symboles sont données section 4. Nous y expliquons notre choix de caractéristiques et la méthode que nous avons utilisée pour résoudre le problème de la dimensionnalité. Enfin nous présentons les résultats expérimentaux et donnons nos perspectives de travail (sections 5 et 6).

2 Les réseaux Bayésiens comme classificateurs

Un réseau Bayésien est un modèle graphique probabiliste particulier : il s'agit d'un graphe orienté sans cycle. Chaque noeud du graphe représente une variable aléatoire, continue ou discrète, et chaque arc orienté correspond à une dépendance conditionnelle entre deux variables. Notons $\{v_1, v_2, \dots, v_n\}$, les n variables aléatoires d'un réseau Bayésien, et $Pa(v_i)$, l'ensemble des parents du noeud associé à la variable $v_i, i \in \{1, 2, \dots, n\}$ dans le graphe. Dans un réseau Bayésien, chaque variable est indépendante de ses non-descendants, étant donné ses parents immédiats dans le réseau. Cette hypothèse s'exprime par la probabilité conditionnelle $P(v_i|Pa(v_i))$.

Le réseau Bayésien, dans son état initial, contient les probabilités *a priori*, i.e les probabilités initiales de chaque noeud du réseau, calculées à partir d'un échantillon d'apprentissage. A partir de nouvelles observations des variables, complètes ou incomplètes (dans le cas où seules les valeurs d'un sous-ensemble des variables du réseau sont observées), que l'on appelle "évidences", et des probabilités conditionnelles entre chaque noeud et chacun de ses non-descendants, on obtient la distribution de probabilité jointe de l'ensemble

des variables du réseau, que l'on peut exprimer par :

$$P(v_1, v_2, \dots, v_n) = \prod_{i=1}^n P(v_i|Pa(v_i)) \quad (1)$$

Par conséquent, une distribution de probabilité jointe assez complexe peut être ramenée à un ensemble de simples probabilités conditionnelles. La distribution de probabilité jointe $P(v_1, v_2, \dots, v_n)$ a pu être réduite à la formule 1 grâce à l'hypothèse d'indépendance conditionnelle de chaque variable étant donné ses non-descendants.

Rappelons que l'on souhaite affecter une instance particulière $f = \{f_1, f_2, \dots, f_n\}$ du vecteur caractéristique $F = \{F_1, F_2, \dots, F_n\}$ à une classe c_i entre k classes. Le principe est alors d'estimer, à partir d'un ensemble de données étiquetées (échantillon d'apprentissage) la probabilité de chaque caractéristique f_1, f_2, \dots, f_n étant donnée la classe c_i . La classification est faite en appliquant la règle de Bayes pour calculer la probabilité de c_i étant donnée l'instance particulière f , et en prédisant la classe qui a la probabilité *a posteriori* la plus grande. C'est-à-dire on cherche la classe $c_i, i \in \{1, 2, \dots, k\}$ telle que $P(c_i|f_1, f_2, \dots, f_n)$ est maximale et où :

$$\begin{aligned} P(c_i|f_1, f_2, \dots, f_n) &= \frac{P(f_1, f_2, \dots, f_n, c_i)}{P(f_1, f_2, \dots, f_n)} \\ &= \frac{P(f_1, f_2, \dots, f_n|c_i)P(c_i)}{P(f_1, f_2, \dots, f_n)} \end{aligned}$$

et

$$P(f_1, f_2, \dots, f_n) = \sum_{j=1}^k P(f_1, f_2, \dots, f_n|c_j)P(c_j)$$

Ce simple classificateur Bayésien est appelé "Naive Bayes" et peut être décrit par la simple structure de la Figure 1 où :

- C désigne la variable aléatoire "Classe"
- F_1, F_2, \dots, F_n sont les variables caractéristiques

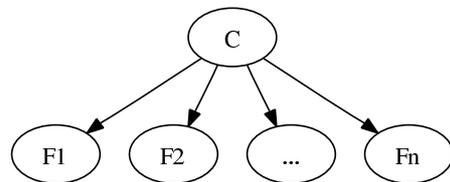


FIG. 1 – Naive Bayes

Il s'agit d'un des classificateurs les plus efficaces, dans le sens que sa performance de prédiction se distingue parmi les classificateurs existants. Il est décrit plus précisément dans [13].

3 Modèle graphique probabiliste comme modèle de mixtures de Gaussiennes

Le Naive Bayes est un modèle efficace, mais il requiert des variables discrètes. Donc, nous avons d’abord discrétisé nos variables, en utilisant la méthode présentée en [2] qui fournit un échantillon optimal à partir des variables continues initiales. Le principe est de construire un histogramme à partir des valeurs, et de fusionner les intervalles de répartition en tenant compte de la perte d’information. Ce modèle est apparu efficace, mais il nécessite un pré-traitement des données, qu’il est possible d’éviter en utilisant les variables continues initiales. De plus, dans le domaine de la reconnaissance d’images, on doit souvent traiter des variables continues. Notre idée est donc de construire un classificateur utilisant un modèle de mixtures de Gaussiennes, représenté par un simple modèle graphique probabiliste.

Nous présentons ci-dessous comment représenter un modèle de mixtures de Gaussiennes par un modèle graphique probabiliste et comment apprendre ses paramètres. Plus de précisions quant à l’utilisation des modèles de mixtures de Gaussiennes pour la classification sont données dans [5].

3.1 Formulation de la méthode et notations

Pour rappel, on dispose d’un ensemble d’observations dont la classe d’appartenance c_i est connue, et on connaît le nombre C de classes. Chaque observation f_j est caractérisée par n attributs, ainsi on a $f_j = \{f_{j_1}, f_{j_2}, \dots, f_{j_n}\}$. Soit k le nombre de Gaussiennes utilisées pour approximer chaque mixture. Chaque classe c_i correspond donc à k Gaussiennes à n dimensions, associées chacune à des paramètres de mélange, qui diffèrent selon les classes.

Cette structure est représentée dans le modèle graphique probabiliste ci dessous :

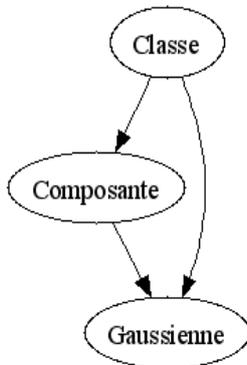


FIG. 2 – Modèle de mixtures de Gaussiennes

- Le nœud “Classe” est un nœud discret, qui peut prendre C valeurs correspondant aux classes prédéfinies.
- Le nœud “Composante” correspond aux composantes de chaque mixture. C’est une variable discrète qui peut prendre k valeurs possibles, i.e le nombre de Gaussiennes nécessaires pour calculer les mixtures. C’est une variable cachée qui représente les poids de chaque Gaussienne dans chaque mixture.
- Enfin la variable “Gaussienne” est continue et représente chaque Gaussienne de chaque mixture, avec ses propres paramètres (moyenne et variance).

3.2 Apprentissage des paramètres

L’apprentissage du modèle se fait à l’aide de l’algorithme EM. Le fonctionnement général de cet algorithme, expliqué plus en détail dans [4], consiste à calculer de manière itérative le maximum de vraisemblance quand les observations peuvent être vues comme données incomplètes : chaque pas d’itération de l’algorithme consiste en une étape de calcul d’espérance suivie par une étape de maximisation, d’où son nom d’algorithme EM (Expectation Maximization). Cet algorithme est remarquable pour sa simplicité et sa généralité, c’est pourquoi nous avons choisi de l’utiliser.

Appliqué à notre cas, pour l’apprentissage des paramètres, l’algorithme EM utilise le principe suivant : il commence par initialiser aléatoirement le modèle, puis, à chaque pas d’itération, met à jour les paramètres de manière à obtenir un optimal local du maximum de vraisemblance. L’algorithme EM est composé de deux étapes : d’abord, chaque point est affecté à chaque composante de la mixture. Ensuite, les paramètres du modèle sont ajustés pour représenter au mieux les données basées sur l’affectation de l’étape précédente.

L’utilisation de l’algorithme EM pour l’apprentissage des modèles de mixtures de Gaussiennes est expliquée plus précisément dans [1].

3.3 Inférence

Une image requête f_j que l’on cherche à classifier, représentée par ses caractéristiques $f_{j_1}, f_{j_2}, \dots, f_{j_n}$, peut être traitée comme évidence et sera instanciée par $P(f_j) = P(f_{j_1}, f_{j_2}, \dots, f_{j_n}) = 1$ quand le réseau sera évalué. Le processus d’inférence consiste alors à calculer les distributions de probabilités *a posteriori* d’un ou plusieurs sous-ensembles d’autres nœuds. Selon la topologie de notre réseau Bayésien, le processus d’inférence propage les valeurs du niveau des caractéristiques de l’image représentées par le nœud “Gaussienne”, en passant par le nœud “Composante” des paramètres de la Gaussienne, jusqu’au niveau du nœud “Classe”. Actuellement il existe plusieurs méthodes d’inférence exactes ou approximatives dans les réseaux Bayésiens. Bien que l’inférence exacte soit en général

un problème NP-difficile [3], elle reste efficace sur de nombreux réseaux Bayésiens. De plus, la complexité en temps des méthodes d'inférence exactes est calculable à l'avance. Quand le résultat dépasse une limite raisonnable, on préfère alors utiliser une méthode d'approximation [8, 10]. Dans notre cas, étant donné que la taille de notre réseau n'est pas très grande, nous pouvons utiliser une méthode d'inférence exacte. Une des méthodes d'inférence exactes les plus influente est l'algorithme JLO, aussi appelé algorithme de l'arbre de jonction, développé dans [9, 14]. Cet algorithme transforme le réseau en un arbre de jonction : les variables sont regroupées de telle façon que les boucles du réseau soient supprimées et que les groupes de variables soient les plus petits possibles. Après la transformation du réseau en arbre, un algorithme de passage de messages [12] est appliqué au réseau. Dans cette technique, chaque noeud est associé à un processeur qui peut envoyer des messages de façon asynchrone à ses voisins jusqu'à ce qu'un équilibre soit atteint. Les probabilités de chaque noeud sont alors mises à jour en fonction de l'observation de l'évidence. Après la propagation de l'inférence, on connaît la probabilité *a posteriori* $P(c_i | f_{j_1}, f_{j_2}, \dots, f_{j_n}), \forall i \in \{1, 2, \dots, C\}$. L'image requête f_j est affectée à la classe c_i pour laquelle cette probabilité est maximale.

4 Reconnaissance de symboles

Cette section explique comment nous avons adapté la méthode théorique développée ci-dessus au problème de la reconnaissance de symboles. Nous y présentons les caractéristiques utilisées ainsi que les méthodes que nous avons dû utiliser pour résoudre le problème de la dimensionnalité.

4.1 Les caractéristiques

Les images à classifier étant des images en noir et blanc, nous avons choisi les trois descripteurs de forme suivants : Generic Fourier Descriptor (GFD), Zernike et la \mathcal{R} -signature 1D, pour calculer les vecteurs de caractéristiques.

Generic Fourier Descriptor. Generic Fourier Descriptor (GFD) est basé sur la transformée de Fourier. Il s'agit d'un descripteur pixels. La transformée n'étant pas invariante à la rotation, Dengsheng Zhang et Guojun Lu [18] proposent d'utiliser la transformée de Fourier modifiée polairement (MPFT). La MPFT est définie par :

$$PF(\rho, \phi) = \sum_r \sum_i f(r, \theta_i) \exp[j2\pi(\frac{r}{R}\rho + \frac{2\pi i}{T}\phi)]$$

où $0 \leq r = [(x - x_c)^2 + (y - y_c)^2]^{\frac{1}{2}} < R$ et $\theta_i = i(2\pi/T) (0 \leq i < T)$, x_c et y_c sont les coordonnées du centre de la forme ; $0 \leq \rho < R$, $0 \leq \phi < T$

où R et T sont les résolutions angulaire et radiale. Après normalisation, $PF(\rho, \phi)$ décrit une signature invariante à la rotation et à l'échelle.

Zernike. Le descripteur Zernike [7] est un descripteur pixels basé sur les moments de Zernike. Ces moments sont construits à partir de polynômes complexes (formule 2) et forment un ensemble orthogonal complet défini sur le disque unité.

$$A_{mn} = \frac{m+1}{\pi} \int_x \int_y I(x, y) [V_{mn}(x, y)] dy dx \quad (2)$$

m et n représentent l'ordre du moment et $I(x, y)$ le niveau de gris d'un pixel sur l'image. V_{mn} , le polynôme de Zernike, est exprimé en coordonnées polaires :

$$V_{mn}(r, \theta) = R_{mn}(r) e^{-jn\theta}$$

où R_{mn} est le polynôme radial orthogonal :

$$R_{mn}(r) = \sum_{s=0}^{\frac{m-|n|}{2}} (-1)^s \frac{(m-s)!}{s! (\frac{m+|n|}{2} - s)! (\frac{m-|n|}{2} - s)!} r^{m-2s}$$

Une fois les moments calculés, on obtient un vecteur pour chaque image. Ce descripteur est invariant à la rotation et au changement d'échelle.

\mathcal{R} -signature 1D. La \mathcal{R} -signature [15] se base sur la transformée de Radon pour représenter une image. La transformée de Radon est la projection d'une image dans un plan particulier. Elle est définie par la formule 3 où $\delta(\cdot)$ est la fonction telle que $\delta(x) = 1$ si $x = 0$ et 0 sinon, $\theta \in [0, \pi[$ et $\rho \in]-\infty, \infty[$.

$$T_{R^f}(\rho, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - \rho) dx dy \quad (3)$$

Cette projection possède des propriétés géométriques intéressantes qui font d'elle un bon descripteur de forme. Suivant ces propriétés géométriques, une signature de la transformée, notée \mathcal{R}_f , est créée :

$$\mathcal{R}_f(\theta) = \int_{-\infty}^{\infty} T_{R^f}^2(\rho, \theta) d\rho$$

Cette signature vérifie les propriétés d'invariance à certaines transformations géométriques, telles que la translation et le changement d'échelle (après normalisation). Par contre l'invariance à la rotation est restaurée par permutation cyclique de la signature ou directement à partir de sa transformée de Fourier.

4.2 Réduction de la dimensionnalité

Une fois que l'ensemble des $n, \forall n \in \{1, 2, 3\}$ descripteurs que l'on souhaite combiner est calculé sur chaque image, on dispose de n signatures par image. Chaque

image sera alors représentée par un vecteur de caractéristiques correspondant à la concaténation des n signatures obtenues. Ce vecteur de caractéristiques est de dimension égale à la somme des dimensions des signatures qui le composent. Les signatures étant déjà de grandes dimensions, leur concaténation nous place face à un problème de dimensionnalité. En effet, une trop grande dimension des caractéristiques implique un apprentissage incorrect des mélanges de Gaussiennes, car il y a une disproportion entre la taille de l'échantillon d'apprentissage et la taille des vecteurs caractéristiques. Pour pallier ce problème, nous utilisons une méthode de réduction de dimensionnalité, avec une perte minimale d'information, en extrayant des vecteurs initiaux seulement les caractéristiques les plus pertinentes et discriminantes. Le LASSO est une méthode de sélection de variables et de diminution de coefficients pour la régression linéaire, introduite par R. Tibshirani [16]. L'objectif est de minimiser la somme des erreurs quadratiques, avec un seuil associé à la somme des valeurs absolues des coefficients :

$$\beta^{lasso} = \arg \min_{\beta} \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2$$

avec la contrainte $\sum_{j=1}^p |\beta_j| \leq s$

Le LASSO utilise une pénalité $L_1 : \sum_{j=1}^p |\beta_j|$. Cette contrainte implique que pour des petites valeurs de s , $s \geq 0$, certains coefficients β_j seront égaux à 0. Ces coefficients nuls correspondent aux variables que l'on va supprimer de notre base et seules les variables correspondant aux coefficients non nuls sont sélectionnées.

Le calcul des solutions du LASSO est un problème de programmation quadratique, qui peut être traité par des algorithmes standards d'analyse numérique. Un des plus adaptés est l'algorithme de Least Angle Regression (LAR). En effet, il exploite la structure particulière du problème du LASSO et fournit un moyen efficace de calculer simultanément les solutions pour toutes les valeurs de s . Cet algorithme est présenté en détail dans [6].

Nous avons appliqué cette méthode à nos données d'apprentissage : y représente donc notre variable classe. Pour chaque classe c_i , sa valeur y_i est la somme des caractéristiques du modèle de la classe c_i . Et $x_j = \{x_{j1}, x_{j2}, \dots, x_{jp}\}$ le vecteur des p caractéristiques de l'observation j .

Cette méthode s'est avérée plus efficace expérimentalement, du point de vue du taux de reconnaissance, qu'une approche par analyse en composantes principales (ACP).

5 Résultats expérimentaux

5.1 Base GREC

Nous avons utilisé des symboles de la base GREC pour effectuer nos tests [17]. Cette base (voir FIG. 3), a été créée spécialement dans le cadre du concours de reconnaissance de symboles à GREC'2005.

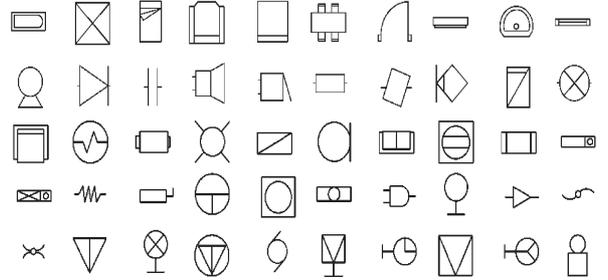


FIG. 3 – Base de symboles GREC.

Les symboles de cette base sont issus principalement de deux domaines d'application, architecture et électronique, car ces symboles sont le plus largement utilisés par les équipes et représentent un grand nombre de formes différentes. Cinquante modèles différents ont été utilisés, plusieurs d'entre eux ayant des formes similaires.

On dispose donc d'une base de 50 symboles parfaits, sur lesquels nous avons effectué des dégradations basées sur le modèle Kanungo [11]. Ces dégradations sont similaires au bruit obtenu quand un document est scanné, imprimé ou photocopie (bruits de type global et local et fermeture morphologique). Nous avons aussi appliqué aux symboles des rotations de différents angles et différents zooms, de façon à obtenir une base de 3600 symboles, constituée de 72 images différentes par modèle. Sur cette base nous avons défini plusieurs tests d'apprentissage : nous avons constitué des échantillons d'apprentissage et de test variés et de différentes tailles. Nous avons testé la méthode en utilisant les signatures issues de un, deux ou des trois descripteurs et en effectuant toutes les combinaisons possibles.

Les tableaux 1 à 3 présentent les taux de reconnaissance (en pourcentage) obtenus par les différents tests. Dans ces tableaux, G désigne le descripteur GFD, Z le descripteur Zernike et R la \mathcal{R} -signature. Enfin l'opérateur + indique que l'on combine les descripteurs (représentés par les opérandes). La sélection de variables par la méthode du LASSO nous a permis de sélectionner 13 variables parmi 225 variables initiales pour GFD, 15 parmi 34 pour Zernike et 13 parmi 180 pour la \mathcal{R} -signature. Les résultats présentés dans le tableau 1 ont été obtenus en prenant le même nombre de variables que celles obtenues avec le LASSO, mais en sélectionnant les premières variables de chaque signa-

ture, i.e on utilise les 13 premières composantes de la signature GFD, les 15 premières de la signature Zernike et les 13 premières de la \mathcal{R} -signature.

Le tableau 2 présente les résultats obtenus après la sélection automatique, avec la méthode du LASSO, d'un sous-ensemble de variables pertinentes. Les résultats obtenus après sélection de variables par le LASSO sont meilleurs, à nombre de variables égal, que ceux obtenus sans sélection de variables. De plus, ces résultats montrent l'intérêt de la combinaison de descripteurs : quels que soient les taux de reconnaissance obtenus par un seul descripteur, en combinant 2 ou 3 descripteurs, nous égalons ou améliorons ces taux. De même, quels que soient les taux obtenus en combinant 2 descripteurs, en utilisant les 3 descripteurs, nous égalons ou améliorons ces taux. On remarque en plus que même lorsque l'on obtient un taux de reconnaissance élevé voire de 100% avec le descripteur Zernike, ce taux ne subit pas de baisse lorsque l'on combine le descripteur avec un ou deux autres descripteurs, quels qu'ils soient, même lorsque les descripteurs avec lesquels on le combine présentent un taux moyen, comme c'est le cas avec la \mathcal{R} -signature. Par exemple, prenons le test où l'échantillon d'apprentissage est constitué des images 19 à 72 et celui de test des images 1 à 18. Dans ce cas la \mathcal{R} -signature affiche un taux de reconnaissance de seulement 40.9% et Zernike un taux de 100%. En combinant ces deux descripteurs, on conserve un taux de 100% ce qui signifie que le mauvais comportement d'un descripteur n'entrave pas le bon comportement d'un autre.

Enfin le tableau 3 présente les résultats de la combinaison des 3 descripteurs avec notre approche, comparés à ceux d'une méthode de combinaison de descripteurs simple. Cette méthode empirique consiste à calculer une distance euclidienne pour chaque descripteur entre l'image à classer et le modèle "parfait" de chaque classe. On effectue la moyenne des distances obtenues pour chaque descripteur et on affecte l'image à la classe qui minimise cette distance moyenne. Cette expérience confirme l'efficacité de notre méthode. En effet, quel que soit l'échantillon d'apprentissage, notre modèle offre de meilleurs résultats que la méthode de combinaison par moyenne. De plus on observe pour notre approche un taux de reconnaissance supérieur de 15% en moyenne à celui obtenu par la méthode de type moyenne.

5.2 Base Ubyte

Ubyte est une base de 2500 chiffres manuscrits : on dispose de 250 images par chiffre. Les 10 modèles sont présentés dans le tableau 4.

Sur cette base la méthode du LASSO nous fournit 15 variables pour GFD, à partir des 225 initiales et 30 pour Zernike, à partir des 34 initiales. Nous avons effectué les mêmes tests que précédemment, avec seule-

ment deux descripteurs, GFD et Zernike. Les taux de reconnaissance (cf tableau 5) sont bien inférieurs à ceux que l'on a pu obtenir sur la base GREC, car nous n'avons pas écarté la propriété d'invariance à la rotation des descripteurs, qui apporte de la confusion dans la reconnaissance de certains chiffres (par exemple entre le 6 et le 9). Par contre, comme sur la base GREC, on remarque qu'en combinant les descripteurs GFD et Zernike, nous améliorons le taux de reconnaissance obtenu par GFD seul ou Zernike seul.

6 Conclusion et perspectives

Dans ce papier nous avons proposé une adaptation originale de la théorie Bayésienne associée à la sélection de variables par la méthode du LASSO. Les résultats obtenus sont prometteurs et montrent que notre méthode permet d'améliorer le taux de reconnaissance en combinant des descripteurs. Ces résultats montrent également une amélioration du taux de reconnaissance par rapport à un opérateur de combinaison de type moyenne. De plus, nous avons montré qu'utiliser la méthode du LASSO résout non seulement le problème de la dimensionnalité des vecteurs caractéristiques, mais permet en plus d'améliorer le taux de reconnaissance.

Nous souhaitons intégrer dans nos futurs travaux une démarche de "relevance feedback". Plus précisément, il s'agit de prendre en compte les informations données par l'utilisateur, sur certaines requêtes, pour modifier les paramètres du réseau après inférence sur la classe de l'image requête. Les réseaux Bayésiens peuvent permettre de n'utiliser le relevance feedback que pour certaines requêtes, de par leur capacité à traiter des données incomplètes. De même, nous souhaitons pouvoir utiliser l'information donnée par des mots clés associés à un sous-ensemble des données d'apprentissages. Ceci est possible car les réseaux Bayésiens permettent également de traiter au sein d'un même réseau des informations de différents types, en l'occurrence, utiliser différents médias.

Références

- [1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [2] O. Colot, C. Olivier, P. Courtellemont, A. El-Matouat, and D. de Brucq. Information criteria and abrupt changes in probability laws. *Signal Processing VII : Theories and Applications*, 1 :387–391, 2004.
- [3] G. F. Cooper. The computational complexity of probabilistic inference using bayesian belief networks (research note). *Artificial Intelligence*, 42(2-3) :393–405, 1990.
- [4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the

Spécifications		G	Z	R	G+Z	G+R	Z+R	G+Z+R
apprentissage	test							
ims 1-54	ims 55-72	80.3	97.1	35.7	97.3	81.2	97.4	97.5
ims 19-72	ims 1-18	80.2	96.8	35.5	97.7	81.2	96.8	97.7
ims 1-36	ims 37-72	79.7	95.4	34.8	95.6	80.7	96.2	95.8
ims 37-72	ims 1-36	80.1	96.5	34.9	97.7	80.4	96.7	97.9
ims 1-18	ims 19-72	77.8	93.7	33.9	95.9	79.6	94.2	98.3
ims 55-72	ims 1-54	75.4	92.4	34.7	96.0	77.9	92.5	95.2

TAB. 1 – Taux de reconnaissance sans sélection de variables - base GREC

Spécifications		G	Z	R	G+Z	G+R	Z+R	G+Z+R
apprentissage	test							
ims 1-54	ims 55-72	99.1	100	51.3	100	99.4	100	100
ims 19-72	ims 1-18	98.9	100	40.9	100	99.2	100	100
ims 1-36	ims 37-72	98.3	100	44.3	100	99	100	100
ims 37-72	ims 1-36	98.6	100	45.5	100	99.2	100	100
ims 1-18	ims 19-72	98.4	100	46	100	98.6	100	100
ims 55-72	ims 1-54	98.3	100	44.1	100	98.3	100	100

TAB. 2 – Taux de reconnaissance avec sélection de variables par méthode du LASSO - base GREC

Spécifications		Combinaison par moyenne	Méthode proposée
apprentissage	test		
ims 1-54	ims 55-72	91.1	100
ims 19-72	ims 1-18	74.8	100
ims 1-36	ims 37-72	91.2	100
ims 37-72	ims 1-36	87.1	100
ims 1-18	ims 19-72	90.3	100
ims 55-72	ims 1-54	75.9	100

TAB. 3 – Taux de reconnaissance méthode proposée vs. combinaison par moyenne - base GREC

			
0	1	2	3
			
4	5	6	7
			
8	9		

TAB. 4 – Base Ubyte

Spécifications		G	Z	G+Z
apprentissage	test			
ims 1-188	ims 189-250	34.7	65.3	66.6
ims 63-250	ims 1-62	31.2	63.3	65.2
ims 1-125	ims 126-250	31.5	70.2	73.1
ims 126-250	ims 1-125	31.3	66.6	67.9
ims 1-62	ims 63-250	30.1	69.7	72.4
ims 189-250	ims 1-188	29.9	67.4	68.7

TAB. 5 – Taux de reconnaissance avec sélection de variables par la méthode du LASSO - base UBYTE

em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1) :1–38, 1977.

- [5] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification, Second Edition*. Wiley-Interscience, 2001.
- [6] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32 :407–499, 2004.
- [7] Kim H.-K., J.-D. Kim, D.-G. Sim, and D.-Il Oh. A modified zernike moment shape descriptor invariant to translation, rotation and scale for similarity-based image retrieval. *IEEE International Conference On Multimedia And Expo*, 1 :307–310, 2000.
- [8] T. Jaakkola and M. I. Jordan. Variational probabilistic inference and the QMR-DT network. *Journal of Artificial Intelligence Research*, 10 :291–322, 1999.
- [9] F. Jensen, S.L. Lauritzen, and K. Olesen. Bayesian updating in recursive graphical models by local computations. *Computational Statistical Quarterly*, 4 :269–282, 1990.
- [10] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2) :183–233, 1999.
- [11] T. Kanungo, R. Haralick, H. Baird, W. Stuezel, and D. Madigan. A statistical, nonparametric methodology for document degradation model validation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11) :1209–1223, 2000.
- [12] J. H. Kim and J. Pearl. A computational model for combined causal and diagnostic reasoning in inference systems. In *IJCAI-83*, pages 190–193, 1983.
- [13] P. Langley, W. Iba, and K. Thompson. An analysis of bayesian classifiers. In *National Conference on Artificial Intelligence*, pages 223–228, 1992.
- [14] S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 50(2) :157–224, 1988.
- [15] S. Tabbone, L. Wendling, and J.-P. Salmon. A new shape descriptor defined on the radon transform. *Computer Vision and Image Understanding*, 102(1) :42–51, 2006.
- [16] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1) :267–288, 1996.
- [17] E. Valveny and P. Dosch. Symbol recognition contest : A synthesis. In *Graphic Recognition, volume 3088 of Lecture Notes in Computer Science*, pages 368–385, 2004.
- [18] D. S. Zhang and G. Lu. Shape-based image retrieval using general fourier descriptor. *Signal Processing : Image Communication*, 17(10) :825–848, 2002.