HAL
archives-ouvertes.fr

Classification et extension automatique d'annotations d'images en utilisant un réseau Bayésien

Sabine Barrat, Salvatore Tabbone

▶ **To cite this version:**

Sabine Barrat, Salvatore Tabbone. Classification et extension automatique d'annotations d'images en utilisant un réseau Bayésien. Traitement du Signal, Lavoisier, 2009, 26 (5), pp.24. <http://hdl.handle.net/2042/32607>. <inria-00539035>

HAL Id: inria-00539035
https://hal.inria.fr/inria-00539035

Submitted on 23 Nov 2010

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.

Classification et extension automatique d'annotations d'images en utilisant un réseau Bayésien

Sabine Barrat et Salvatore Tabbone

LORIA-UMR7503

Université Nancy 2

BP 239, 54506 Vandœuvre-les-Nancy Cedex

E-mail: {barrat,tabbone}@loria.fr

Résumé

Nous proposons, dans cet article, d'améliorer la classification d'images, en utilisant une approche de classification visuo-textuelle (à base de caractéristiques visuelles et textuelles), et en étendant automatiquement les annotations existantes aux images non annotées. L'approche proposée est dérivée de la théorie des modèles graphiques probabilistes et dédiée aux deux tâches de classification et d'annotation d'images partiellement annotées. Nous considérons une image comme partiellement annotée si elle ne possède pas le nombre maximal de mots-clés disponibles par image dans la vérité-terrain. Grâce à leur capacité à fonctionner en présence de données manquantes, un modèle graphique probabiliste a été proposé pour représenter les images partiellement annotées. Ce modèle est basé sur un mélange de lois multinomiales et de mélanges de Gaussiennes. La distribution des caractéristiques visuelles est estimée par des mélanges de Gaussiennes et celle des mots-clés par une loi multinomiale. Par conséquent, le modèle proposé ne requiert pas que toutes les images soient annotées : lorsqu'une image est partiellement annotées, les mots-clés manquants sont considérés comme des valeurs manquantes. De plus, notre modèle peut automatiquement étendre des annotations existantes à des images partiellement annotées, sans l'intervention de l'utilisateur. L'incertitude autour de l'association entre un ensemble de mots-clés et une image est capturée par une distribution de probabilité jointe (définie par un mélange de lois multinomiales et de mélanges de Gaussiennes) sur le dictionnaire de mots-clés et les caractéristiques visuelles extraites de notre collection d'images. De plus, de façon à résoudre le problème de dimensionnalité dû à la grande dimension des caractéristiques visuelles, nous avons adapté une méthode de sélection de variables. Les résultats de la classification visuo-textuelle, obtenus sur une base d'images collectées sur Internet, partiellement et manuellement annotée, montrent une amélioration de 32.3% *en terme de taux de reconnaissance, par rapport à la classification basée sur l'information visuelle uniquement. Par ailleurs, l'extension automatique*

d'annotations, avec notre modèle, sur des images avec mots-clés manquants, améliore encore la classification visuo-textuelle de 6.8%. *Enfin, la méthode proposée s'est montrée compétitive avec des classificateurs de l'état de l'art.*

Mots-clés

modèles graphiques probabilistes, réseaux Bayésiens, sélection de variables, classification, annotation automatique

Abstract

The rapid growth of Internet and multimedia information has shown a need in the development of multimedia information retrieval techniques, especially in image retrieval. We can distinguish two main trends. The first one, called "text-based image retrieval", consists in applying text-retrieval techniques from fully annotated images. The text describes high-level concepts but this technique presents some drawbacks: it requires a tedious work of annotation. Moreover, annotations could be ambiguous because two users can use different keywords to describe a same image. Consequently some approaches have proposed to use Wordnet in order to reduce these potential ambiguities. The second approach, called "content-based image retrieval" is a younger field. These methods rely on visual features (color, texture or shape) computed automatically, and retrieve images using a similarity measure. However, the obtained performances are not really acceptable, except in the case of well-focused corpus. In order to improve the recognition, a solution consists in combining visual and semantic information. In many vision problems, instead of having fully annotated training data, it is easier to obtain just a subset of data with annotations, because it is less restrictive for the user. This paper deals with modeling, classifying, and annotating weakly annotated images. More precisely, we propose a scheme for image classification optimization, using a joint visual-text clustering approach and automatically extending image annotations. The proposed approach is derived from the probabilistic graphical model theory and dedicated for both tasks of weakly-annotated image classification and annotation. We consider an image as weakly annotated if the number of keywords defined for it is less than the maximum defined in the ground truth. Thanks to their ability to manage missing values, a probabilistic graphical model has been proposed to represent weakly annotated images. We propose a probabilistic graphical model based on a Gaussian-Mixtures and Multinomial mixture. The visual features are estimated by the Gaussian mixtures and the keywords by a Multinomial distribution. Therefore, the proposed model does not require that all images be annotated: when an image is weakly annotated, the missing keywords are considered as missing values. Besides, our model can automatically extend existing annotations to weakly-annotated images, without user intervention. The uncertainty around the association between a set of keywords and an image is tackled by a joint probability distribution (defined from Gaussian-Mixtures and Multinomial mixture) over the dictionary of keywords and the visual features extracted from our collection of images. Moreover, in order to solve the dimensionality problem due to the large dimensions of visual features, we have adapted a variable selection method. Results of visual-textual classification, reported on a database of images collected from the Web, partially and manually annotated, show an improvement of about 32.3% *in terms of recognition rate against only visual information classification. Besides the automatic annotation extension with our model for images with missing keywords outperforms the visual-textual classification of about* 6.8%*. Finally the proposed method is experimentally competitive with the state-of-art classifiers.*

Keywords: *probabilistic graphical models, Bayesian networks, variable selection, image classification, image annotation*

I. Introduction

La croissance rapide d'Internet et de l'information multimédia a engendré un besoin en techniques de recherche d'information multimédia, et plus particulièrement en recherche d'images. On peut distinguer deux tendances. La première, appelée recherche d'images par le texte, consiste à appliquer des techniques de recherche de textes à partir d'ensembles d'images complètement annotés. L'efficacité de ces méthodes est étroitement liée à la qualité de l'indexation des images. Or, les méthodes d'indexation textuelle automatiques sont peu performantes et fournissent des ensembles d'images mal annotées, car elles utilisent l'URL, le titre de la page, ou d'autres attributs ou le texte proche de l'image dans le cas d'images provenant d'Internet, ou alors tout simplement le nom de l'image dans le cas d'images issues de collections personnelles. Quant à l'indexation textuelle manuelle, bien qu'elle soit plus performante que l'indexation textuelle automatique, elle est très coûteuse pour l'utilisateur et se révèle pratiquement inapplicable aux grandes bases d'images.

La seconde approche, appelée recherche d'images par le contenu, est un domaine plus récent et utilise une mesure de similarité (similarité de couleur, forme ou texture) entre une image requête et une image du corpus utilisé. Ces méthodes sont efficaces sur certaines bases d'images, mais leurs performances décroissent sur des bases d'images plus généralistes.

Afin d'améliorer la reconnaissance, une solution consiste à combiner différentes sources d'informations. Dans un premier temps sont apparues des approches de combinaison de caractéristiques [1], [2] et de combinaison de classificateurs[1] [3], [4]. Dans le cas de la combinaison de caractéristiques, un seul classificateur est utilisé pour combiner plusieurs caractéristiques. Au contraire, les approches de combinaison de classificateurs prennent une décision globale à partir des décisions individuelles prises par chaque classificateur. Par ailleurs, des approches de combinaison d'informations visuelles et sémantiques, appelées "approches visuo-textuelles", ont été proposées. L'annotation d'images par mots-clés constitue une manière possible d'associer de la sémantique à une image. En effet, elle consiste à assigner à chaque image, un mot-clé ou un ensemble de mots-clés, destiné(s) à décrire le contenu sémantique de l'image. Ainsi cette opération peut être vue comme une fonction permettant d'associer de l'information visuelle, représentée par les caractéristiques de bas niveau (forme, couleur, texture, ...) de l'image, à de l'information sémantique, représentée par ses mots-clés, dans le but de réduire le fossé sémantique ("semantic gap" en anglais) [5]. Ainsi, il est possible d'obtenir des bases d'images annotées et des approches visuo-textuelles peuvent être mises en place. Par exemple, Barnard et al. [6] segmentent les images en régions. Chaque région est représentée par un ensemble de caractéristiques visuelles et un ensemble de mots-clés. Les images sont alors classifiées en modélisant de façon hiérarchique les distributions de leurs mots-clés et caractéristiques visuelles. Grosky et al. [7] associent des coefficients aux mots afin de réduire la dimensionnalité. Les vecteurs de caractéristiques visuelles et les vecteurs d'indices correspondant aux mots-clés sont concaténés pour procéder à la recherche d'images. Benitez et al. [8] extraient de la connaissance à partir de collections d'images annotées, en classifiant les images représentées

[1] on peut également utiliser le terme "classifieur", par abus de langage par rapport au mot anglais "classifier"

par leurs caractéristiques visuelles et textuelles. Des relations entre les informations visuelles et textuelles sont alors découvertes. La plupart de ces méthodes présentent l'inconvénient majeur de nécessiter que toute la base d'images soit entièrement annotée. Or, de telles bases sont très difficiles à obtenir car elles requièrent un travail coûteux d'annotation manuelle de la base (les méthodes d'indexation textuelle automatique étant moins performantes que l'annotation manuelle). On préfèrera donc s'orienter vers des méthodes dédiées à des bases d'images partiellement annotées. De plus, des techniques d'annotation automatique d'images pourront être utilisées afin de compléter les annotations des images partiellement annotées. En effet, l'annotation automatique d'images peut être utilisée dans les systèmes de recherche d'images, pour organiser et localiser les images recherchées ou pour améliorer la classification visuo-textuelle. Cette méthode peut être vue comme un type de classification multi classes avec un grand nombre de classes, aussi large que la taille du vocabulaire. Plusieurs travaux ont été proposés dans ce sens. On peut citer, sans être exhaustif, les méthodes basées sur la classification [9], [10], les méthodes probabilistes [11], [12] et l'affinement d'annotations [13], [14]. Par exemple, la méthode [13] propose de segmenter les images en blocs visuels. Chaque bloc est décrit par des caractéristiques de couleur, texture et forme. Un algorithme de clustering est appliqué pour regrouper en classes les blocs visuels similaires. Ensuite, pour chaque classe, les caractéristiques les plus pertinentes sont sélectionnées, en se basant sur l'analyse des histogrammes des caractéristiques. Des liens sont alors créés entre certains mots-clés et certaines classes, pour chaque région. L'annotation automatique d'une image se fait selon la distance entre les caractéristiques visuelles de cette image et les caractéristiques visuelles de chaque centre de classe. Chaque bloc visuel est alors associé au mot-clé de la classe la plus proche. Cette méthode, contrairement aux méthodes classiques basées sur la classification, peut être appliquée sur de plus grand corpus de données. Cependant, elle requiert, comme les méthodes classiques, un lourd travail d'annotation manuelle à cause de la segmentation en régions. Dans [15], l'algorithme EM et la règle de Bayes sont utilisés pour connecter chaque caractéristique à des mots-clés. Chaque image est annotée par le mot-clé ayant la plus grande probabilité étant données ses caractéristiques visuelles. Jin et al. [16] proposent un modèle de langage pour annoter des images. Étant donnée une image test, ce modèle de langage estime la probabilité d'un ensemble de mots-clés. L'ensemble de mots-clés ayant la plus grande probabilité est associé à l'image, si cette probabilité dépasse un certain seuil.

Dans cette direction, la contribution de ce papier est de proposer une méthode pour optimiser la classification d'images, en utilisant une approche de classification visuo-textuelle et en étendant automatiquement des annotations existantes. Plus précisément, le modèle présenté ici est dédié aux deux tâches de classification et d'annotation d'images partiellement annotées (images comportant moins de mots-clés que le nombre maximal de mots-clés disponibles dans la vérité-terrain pour une image). En effet, la plupart des méthodes de classification visuo-textuelles sont efficaces, mais requièrent que toutes les images, ou régions d'images, soient annotées. De plus, la plupart des modèles d'annotation automatique existants ne sont pas capables de classifier des images, car ils sont uniquement dédiés à l'annotation. Le modèle que nous proposons ne nécessite pas que toutes les images soient annotées : quand une image est partiellement annotée, les mots-clés manquants sont considérés comme

des données manquantes. Notre modèle permet aussi d'étendre automatiquement des annotations existantes à des images partiellement annotées, sans l'intervention de l'utilisateur. Le modèle [11] est le plus proche de notre approche, car il permet de classifier des images sur la base de caractéristiques visuelles et textuelles, et d'annoter automatiquement de nouvelles images, mais il est moins performant pour l'extension d'annotations.
L'approche proposée est dérivée de la théorie des modèles graphiques probabilistes. Nous introduisons une méthode pour traiter le problème des données manquantes dans le contexte d'images annotées par mots-clés comme défini en [11], [17]. L'incertitude autour de l'association entre un ensemble de mots-clés et une image est représentée par une distribution de probabilité jointe sur le vocabulaire et les caractéristiques visuelles extraites de notre collection d'images couleurs ou à niveaux de gris. Les réseaux Bayésiens sont un moyen simple de représenter une distribution de probabilité jointe d'un ensemble de variables aléatoires, de visualiser les propriétés de dépendance conditionnelle, et ils permettent d'effectuer des calculs complexes comme l'apprentissage des probabilités et l'inférence, avec des manipulations graphiques. Un réseau Bayésien semble donc approprié pour représenter et classifier des images associées à des mots-clés. Enfin, étant donnée la taille importante des caractéristiques visuelles, car nous en combinons plusieurs, un algorithme de sélection de variables est introduit afin de réduire la complexité de notre méthode.
Le reste de ce papier est organisé de la façon suivante : dans la section II, les propriétés des classificateurs basés sur les réseaux Bayésiens sont introduites et nous conduisent à présenter notre propre réseau Bayésien pour la classification et l'extension d'annotations d'images (dans la section III). Les caractéristiques visuelles utilisées pour représenter les images sont décrites section IV. L'algorithme de sélection de caractéristiques, qui nous a permis d'augmenter notre taux de reconnaissance, tout en réduisant la complexité de notre méthode, est expliqué section V. Les résultats expérimentaux sont présentés section VI. Enfin, les conclusions et perspectives de ce travail sont données section VII.

II. Représentation et classification d'images

A. Contexte et objectifs

Dans ce travail, nous nous intéressons à la classification et à l'extension d'annotations d'images partiellement annotées. Étant donnée une base d'images, nous essayons de reconnaître l'objet représenté par l'image. Ce problème de reconnaissance peut être vu comme un problème de classification : notre but est d'affecter chaque image à une classe correspondant à un objet donné. Cependant nous ne disposons pas de modèle pour chaque classe. Par conséquent, nous ne pouvons pas nous contenter de minimiser une distance entre chaque image de la base et chaque modèle. Par contre, cette tâche de classification peut être résolue en utilisant une méthode d'apprentissage supervisée, à partir d'un sous-ensemble des images de la base pour lesquelles les étiquettes de classe sont connues. De plus, de façon à décrire plus précisément les images et à améliorer le taux de reconnaissance, nous proposons de combiner deux descripteurs (un de forme et un de couleur), afin de représenter l'information visuelle contenue dans

l'image, et d'utiliser les mots-clés annotant certaines images afin de prendre en compte l'information sémantique qu'elles véhiculent. Les descripteurs fournissent en général des vecteurs de caractéristiques continues, et les mots-clés sont considérés comme des variables discrètes.

Soit f_j une image requête caractérisée par un ensemble de caractéristiques F composé de :

- m caractéristiques visuelles continues, notées $v_1, ..., v_m$,
- n mots-clés, notés $KW\ 1, ..., KW\ n$.

Par conséquent il semble approprié de proposer un classificateur qui permet de combiner caractéristiques discrètes et continues. De plus, le classificateur proposé se doit d'être robuste aux données manquantes, car toutes les images de la base ne sont pas annotées ou ne le sont que partiellement. La plupart des méthodes de classification ne permettent de traiter que les données discrètes et requièrent ainsi un pré-traitement de discrétisation des données de façon à transformer chaque variable à valeurs continues en variable à valeurs discrètes. Cependant, il existe quelques méthodes de classification permettant de combiner les deux types de variables. C'est le cas, par exemple, des Machine à Vecteurs Supports SVM [18], des forêts aléatoires [19], de l'algorithme des k plus proches voisins (notés KPPV, ou KNN en anglais), et des classificateurs Bayésiens. Les SVM et les forêts aléatoires sont réputés pour être performants en présence d'un grand nombre de variables. Par contre, l'utilisation des SVM devient difficile lorsque le nombre d'observations de la base d'apprentissage est important. Concernant les KNN, la procédure de classification est lourde car chaque image requête est comparée (sur la base des ses caractéristiques) à toutes les images stockées. Par contre cette méthode a l'avantage de ne pas nécessiter d'apprentissage : c'est l'échantillon qui constitue le modèle. Enfin, les classificateurs Bayésiens, quant à eux, sont sensibles à la dimensionnalité des données. Par contre, ils sont efficaces avec beaucoup de données d'apprentissage. Enfin, les classificateurs Bayésiens sont adaptés à la résolution de problèmes en présence de données manquantes, contrairement aux SVM. Par conséquent, nous avons choisi de construire un classificateur Bayésien pour sa capacité à combiner variables discrètes et continues, en présence de nombreuses données d'apprentissages et de données manquantes. De plus, nous montrerons (voir section VI) que ce classificateur Bayésien, associé à une méthode de sélection de variables, est compétitif avec les SVM, les KNN et le réseau Bayésien décrit dans [11], même en présence d'un grande nombre de variables. Enfin, le modèle proposé sera utilisé pour étendre des annotations existantes à des images sans mots-clés ou partiellement annotées, afin d'augmenter le nombre d'annotations de la base existant, en vue d'effectuer des classifications visuo-textuelles plus efficaces.

B. Les classificateurs Bayésiens

Soit I une image caractérisée par une observation particulière $f = \{f_1, ..., f_n\}$ d'un vecteur caractéristique $F = \{F_1, ..., F_n\}$. Notre but est d'affecter l'image I à la classe c_i parmi k classes. Chaque c_i est une observation particulière de la variable C. Le Naïve Bayes (NB) est un simple algorithme de classification probabiliste qui a montré de bonnes performances dans de nombreux domaines. Ce classificateur encode la distribution $P_{NB}(F_1, ..., F_n, C)$, d'un échantillon d'apprentissage donné (composé de données étiquetées). Le modèle probabiliste résultant peut être

utilisé pour classifier une nouvelle observation I. En effet, la règle de Bayes est appliquée pour calculer la probabilité de c_i étant donnée l'observation f. Le classificateur basé sur le modèle NB retourne la classe c_i, $i \in \{1, ..., k\}$, qui maximise la probabilité *a posteriori* $P_i = P_{NB}(c_i|f_1, ..., f_n)$, où :

$$P_i = \frac{P_{NB}(f_1, ..., f_n|c_i) \times P_{NB}(c_i)}{P_{NB}(f_1, ..., f_n)}$$

et $P_{NB}(f_1, ..., f_n) = \sum_{j=1}^{k} P_{NB}(f_1, ..., f_n|c_j) \times P_{NB}(c_j)$

Cependant, nous nous intéressons aux distributions de probabilités de caractéristiques discrètes et continues, et de leurs relations de dépendance conditionnelle. Considérons chaque composante des vecteurs de caractéristiques continues comme une variable discrète continue et les valeurs discrètes provenant des mots-clés comme des variables discrètes. Ce modèle est trop grand (il possède trop de variables) pour être représenté par une unique distribution de probabilité jointe. Par conséquent, il est nécessaire d'introduire de la connaissance structurelle *a priori* : le Naïve Bayes doit être étendu pour prendre en compte les variables discrètes et continues.
Les modèles graphiques probabilistes, et en particulier les réseaux Bayésiens, sont un bon moyen de résoudre ce genre de problème. En effet, dans un réseau Bayésien, la distribution de probabilité jointe est remplacée par une représentation graphique des relations entre variables, uniquement pour les variables s'influençant les unes les autres. Les interactions indirectes entre variables sont ensuite calculées en propageant la connaissance à travers le graphe de ces connections directes. Par conséquent, les réseaux Bayésiens sont un moyen simple de représenter une distribution de probabilité jointe d'un ensemble de variables, de visualiser les propriétés de dépendance conditionnelle et d'effectuer des calculs complexes comme l'apprentissage ou l'inférence, grâce à des manipulations graphiques.

C. Réseaux Bayésiens

a. Définitions

Formellement, un réseau Bayésien pour un ensemble de variables aléatoires V (continues et/ou discrètes) est un couple $B =< G, \Theta >$. Le premier élément, G, est un graphe direct sans cycle dont les nœuds correspondent à des variables aléatoires $V_1, ..., V_n$, et les arcs représentent des dépendances directes entre variables. Le graphe G encode une hypothèse d'indépendance : chaque variable V_i est indépendante de ses non descendants, étant donnés ses parents dans G. Le second élément du couple, Θ, représente l'ensemble des paramètres qui quantifient le réseau. Cet ensemble contient un paramètre $\theta_{v_i|Pa(v_i)} = P_B(v_i|Pa(v_i))$ pour chaque valeur possible v_i de V_i, et $Pa(v_i)$ de $Pa(V_i)$, où $Pa(V_i)$ représente l'ensemble des parents de V_i dans G. Ainsi, dans son état initial, un réseau Bayésien contient les probabilités *a priori* de chaque nœud du réseau : $P_B(v_i|Pa(v_i))$. Grâce à l'hypothèse d'indépendance conditionnelle de chaque variable étant donnés ses parents, la distribution de probabilité jointe $P_B(V_1, ..., V_n)$ peut être réduite à cette formule :

$$P_B(V_1, ..., V_n) = \prod_{i=1}^{n} P_B(V_i|Pa(V_i)) = \prod_{i=1}^{n} \theta_{v_i|Pa(v_i)}$$

Les réseaux Bayésiens sont associés à un ensemble d'algorithmes pour faire de l'inférence (i.e. calculer les probabilités *a posteriori*) et de l'apprentissage (factorisation des paramètres, estimation des probabilités initiales,...). Les algorithmes que nous avons utilisés sont décrits brièvement ci-dessous.

b. Apprentissage des paramètres

Une fois que la description d'un modèle est établie, à savoir sa structure graphique et les lois de probabilités des variables, on cherche à estimer les valeurs numériques de chaque paramètre. Supposons que l'on dispose de variables continues ou discrètes (ou un mélange des deux), et, dans le cas simple, d'un ensemble de données représentatif de plusieurs cas possibles pour chaque variable. L'ensemble des données peut être complet ou incomplet. Suivant le cas, une solution différente va être utilisée. Dans le cas où l'ensemble des données ne présente pas de données manquantes, la méthode la plus simple et la plus utilisée est l'estimation statistique de la probabilité d'un évènement par la fréquence d'apparition de l'évènement dans la base de données. Cette méthode est appelée "maximum de vraisemblance". Soit $\mathcal{D}$ un ensemble de données, alors $P(d|M)$ est la probabilité qu'une donnée $d \in \mathcal{D}$ soit générée par le modèle M, et est appelée la vraisemblance de M étant donné d. Par conséquent, la vraisemblance de M étant donné l'ensemble complet $\mathcal{D}$ est :

$$L(M|\mathcal{D}) = P(\mathcal{D}|M) = \prod_{d \in \mathbb{D}} P(d|M)$$

Pour des raisons de simplicité de calcul, le logarithme est souvent utilisé à la place de la vraisemblance :

$$L(M|\mathcal{D}) = \sum_{d \in \mathcal{D}} log_2 P(d|M)$$

Par conséquent, le principe du maximum de vraisemblance préfère choisir les paramètres avec la plus grande vraisemblance :

$$\hat{\theta} = argmax_\theta L(M_\theta|\mathcal{D})$$

En général, le maximum de vraisemblance est obtenu en comptant la fréquence de l'évènement dans la base. Dans le cas où l'on ne dispose pas d'assez de données pour représenter tous les cas possibles (présence de données manquantes), un des algorithmes les plus populaire est l'algorithme Espérance-Maximization (EM) ("Expectation Maximization" en anglais). Cet algorithme commence par initialiser aléatoirement les paramètres (distributions de probabilités) du modèle. Ensuite, il 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 (Espérance Maximization). Le principe général de cet algorithme est expliqué en détail dans [20]. Dans un réseau Bayésien, la

première étape de l'algorithme EM peut être faite facilement en utilisant un algorithme d'estimation du maximum *a posteriori* (ou "map" en anglais, de "Maximum a posteriori estimation"). Il s'agit de calculer les valeurs les plus probables pour les données manquantes, étant données les variables observées. La seconde étape de l'algorithme EM (celle de maximisation) est alors exécutée, avec un algorithme d'optimisation, si aucune forme du maximum de vraisemblance n'est connue, ou avec l'approche précédente ("map"). Les deux étapes (E et M), sont répétées jusqu'à convergence.
L'algorithme EM est aussi utilisé pour apprendre les paramètres de distributions Gaussiennes, qui peuvent être considérées comme des données manquantes.

c. Inférence

Un algorithme d'inférence est nécessaire pour calculer les distributions de probabilités *a posteriori* des nœuds non observés. Selon la topologie du réseau Bayésien, le processus d'inférence propage les valeurs du niveau des feuilles du graphe jusqu'au nœud inferré. Plusieurs algorithmes peuvent être utilisés [21]. Le plus populaire est l'algorithme de passage de messages [22]. Dans cette technique, chaque nœud est associé à un processeur qui peut envoyer des messages de façon asynchrone à ses voisins jusqu'à ce qu'un équilibre soit atteint.

d. Les réseaux Bayésiens comme classificateurs

Les réseaux Bayésiens peuvent être utilisés comme classificateurs. Par exemple, le Naïve Bayes peut être représenté par la structure de la Figure 1, où :

- C représente la variable classe,
- $F_1, ..., F_n$ sont les variables caractéristiques.

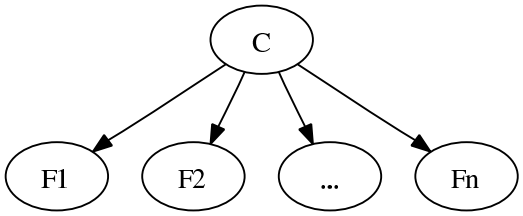


Fig. 1. Naïve Bayes

Le Naïve Bayes est un modèle simple et efficace mais il requiert des variables discrètes. Comme nous devons traiter des variables discrètes (valeurs correspondant aux mots-clés) et continues (valeurs provenant des descripteurs), ce modèle doit être étendu pour les prendre en compte.

III. Un modèle de mélange de lois multinomiales et de densités à mélange de Gaussiennes

Nous présentons un modèle hiérarchique probabiliste multimodal (images et mots-clés associés) pour classifier de grandes bases de données d'images annotées. Nous rappelons que les caractéristiques visuelles sont considérées comme des variables continues, et les éventuels mots-clés associés comme des variables discrètes. De plus, on considère que notre échantillon de caractéristiques visuelles suit une loi dont la fonction de densité est une densité de mélange de Gaussiennes. Les variables discrètes sont supposées suivre une distribution multinomiale sur le vocabulaire des mots-clés : chaque mot-clé d'un vocabulaire de taille N est représenté par un entier entre 1 et N. Nous proposons d'étendre le Naïve Bayes afin de prendre en compte ces distributions de probabilités : le modèle proposé est un modèle de mélange de lois multinomiales et de densités à mélange de Gaussiennes (noté "modèle de mélange GM-Mult"). La structure du Naïve Bayes est conservée c'est-à-dire que l'on dispose d'une variable "Classe", connectée à chaque variable caractéristique (cf. Figure 3).

Soit F un échantillon d'apprentissage composé de m individus $f_{1_i}, ..., f_{m_i}, \forall i \in \{1, ..., n\}$, où n est la dimension des signatures obtenues par concaténation des vecteurs caractéristiques issus du calcul des descripteurs sur chaque image de l'échantillon. Chaque individu $f_j, \forall j \in \{1, ..., m\}$ est caractérisé par n variables continues. Comme nous l'avons vu dans la section II-A, nous sommes dans le cadre d'une classification supervisée. Les m individus sont donc divisés en k classes $c_1, ..., c_k$. Soient $G_1, ..., G_g$ les g groupes dont chacun a une densité Gaussienne avec une moyenne $\mu_l, \forall l \in \{1, ..., g\}$ et une matrice de covariance $\sum_l$. De plus, soient $\pi_1, ..., \pi_g$ les proportions des différents groupes, $\theta_l = (\mu_l, \sum_l)$ le paramètre de chaque Gaussienne et $\Phi = (\pi_1, \pi_1, ..., \pi_g, \theta_1, ..., \theta_g)$ le paramètre global du mélange. Alors la densité de probabilité de F conditionnellement à la classe $c_i, \forall i \in \{1, ..., k\}$ est définie par $P(f, \Phi) = \sum_{l=1}^{g} \pi_l p(f, \theta_l)$ où $p(f, \theta_l)$ est la Gaussienne multivariée définie par le paramètre θ_l.

Ainsi, nous avons un modèle de mélange de Gaussiennes (GMM) par classe. Ce problème peut être représenté par le modèle probabiliste de la Figure 2, où :

- Le nœud "Classe" est un nœud discret, pouvant prendre k valeurs correspondant aux classes prédéfinies $c_1, ..., c_k$.
- Le nœud "Composante" est un nœud discret correspondant aux composantes (i.e les groupes $G_1, ..., G_g$) des mélanges. Cette variable peut prendre g valeurs, i.e le nombre de Gaussiennes utilisé pour calculer les mélanges. Il s'agit d'une variable latente qui représente le poids de chaque groupe (i.e les $\pi_l, \forall l \in \{1, ..., g\}$).
- Le nœud "Gaussienne" est une variable continue représentant chaque Gaussienne $G_l, \forall l \in \{1, ..., g\}$ avec son propre paramètre ($\theta_l = (\mu_l, \sum_l)$). Il correspond à l'ensemble des vecteurs caractéristiques dans chaque classe.
- Enfin, les arêtes représentent l'effet de la classe sur le paramètre de chaque Gaussienne et son poids associé. Le cercle vert sert à montrer la relation entre le modèle graphique proposé et les GMMs : nous avons un GMM (entouré en vert), composé de Gaussiennes et de leur poids associé, par classe. Chaque GMM a son propre paramètre global.

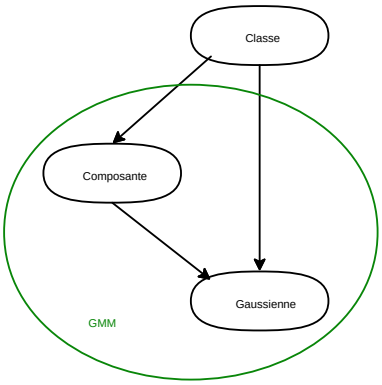


Fig. 2. GMMs représentés par un modèle graphique probabiliste

Maintenant le modèle peut être complété par les variables discrètes, notées $KW\ 1, ..., KW\ n$, correspondant aux éventuels mots-clés associés aux images. Des *a priori* de Dirichlet [23], ont été utilisés pour l'estimation de ces variables. Plus précisément, on introduit des pseudo comptes supplémentaires à chaque instance de façon à ce qu'elles soient toutes virtuellement représentées dans l'échantillon d'apprentissage. Ainsi, chaque observation, même si elle n'est pas représentée dans l'échantillon d'apprentissage, aura une probabilité non nulle. Comme les variables continues correspondant aux caractéristiques visuelles, les variables discrètes correspondant aux mots-clés sont incluses dans le réseau en les connectant à la variable classe.

Notre classificateur peut alors être décrit par la Figure 3. La variable latente "α" montre qu'un *a priori* de Dirichlet a été utilisé. La boîte englobante autour de la variable KW indique n répétitions de KW, pour chaque mot-clé.

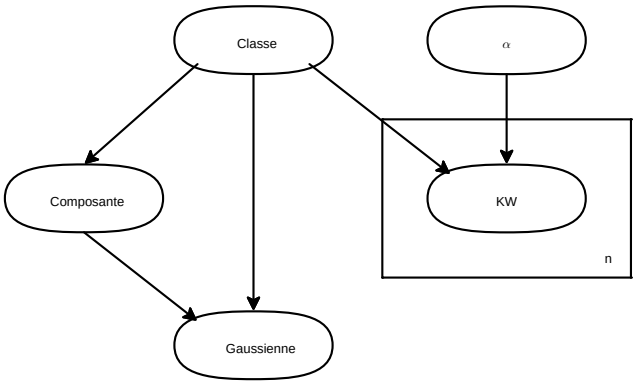


Fig. 3. Modèle de mélange GM-Mult

a. Classification

Pour classifier une nouvelle image f_j, le nœud classe C est inferré grâce à l'algorithme de passage de messages [22]. Ainsi, une image requête f_j, représentée par ses caractéristique visuelles $v_{j_1}, ..., v_{j_m}$ et ses éventuels mots-clés

$KW\ 1_j, ..., KW\ n_j$, est considérée comme une observation (aussi appelée "évidence") représentée par :

$$P(f_j) = P(v_{j_1}, ..., v_{j_m}, KW\ 1_j, ..., KW\ n_j) = 1$$

quand le réseau est évalué. En effet, une évidence correspond à une information nous donnant avec une certitude absolue la valeur d'une variable, d'où la probabilité égale à 1. Grâce à l'algorithme d'inférence (i.e. l'algorithme de passage de messages [22]), les probabilités de chaque nœud sont mises à jour en fonction de cette évidence. On parle de "propagation de croyance ou de propagation de l'évidence". Il s'agit de la phase de calcul probabiliste à proprement parler où les nouvelles informations concernant les variables observées sont propagées à l'ensemble du réseau, de manière à mettre à jour l'ensemble des distributions de probabilités du réseau. Ceci se fait en passant des messages contenant une information de mise à jour entre les nœuds du réseau. A la fin de cette phase, le réseau contiendra la distribution de probabilité sachant les nouvelles informations.
Après la propagation de croyances, on connaît donc, $\forall i \in \{1, ..., k\}$, la probabilité *a posteriori* :

$$P(c_i|f_j) = P(c_i|v_{j_1}, ..., v_{j_m}, DF\ 1_j, ..., DF\ n_j)$$

L'image requête f_j est affectée à la classe c_i maximisant cette probabilité.

b. Extension automatique d'annotations d'images

Étant donnée une image sans mot-clé, ou partiellement annotée, le modèle proposé peut être utilisé pour calculer une distribution des mots-clés conditionnellement à une image et ses éventuels mots-clés existants. En effet, pour une image f_j annotée par $k, \forall k \in \{0, ..., n\}$ mots-clés, où n est le nombre maximum de mots-clés par image, l'algorithme d'inférence permet de calculer la probabilité *a posteriori* $P(KW\ i_j|f_j, KW\ 1_j, ..., KW\ k_j), \forall i \in \{k+1, ..., n\}$. Cette distribution représente une prédiction des mots-clés manquants d'une image. Pour chaque annotation manquante, le mot-clé du vocabulaire ayant la plus grande probabilité est retenu, si cette probabilité atteint un certain seuil. Ainsi, toutes les images ne seront pas annotées par le même nombre de mots-clés à l'issue de l'extension automatique d'annotations.
Par exemple, considérons le tableau I présentant 3 images avec leurs éventuels mots-clés existants et les mots-clés obtenus après l'extension automatique d'annotations. La première image, sans mot-clé, a été automatiquement annotée par deux mots-clés appropriés. De même, la seconde image, annotée au départ par deux mots-clés, a vu son annotation s'étendre à trois mots-clés. Le nouveau mot-clé, "coucher de soleil" est approprié. Enfin, la troisième image, initialement annotée par un mot-clé, a été complétée par deux nouveaux mots-clés. Le premier nouveau mot-clé, "nuage", est correct. Par contre, le second, "coucher de soleil", ne convient pas. Cette erreur est due au grand nombre d'images de la base annotées par les trois mots-clés "pont", "nuage" et "coucher de soleil" (donc la probabilité jointe de ces trois mots-clés est grande), et à l'algorithme d'inférence.
Nous verrons de façon plus complète, dans la partie expérimentale, l'intérêt d'étendre automatiquement des annotations.

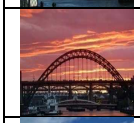
image	mots-clés initiaux	mots-clés après extension automatique d'annotations
		pont eau
	pont nuage	pont nuage coucher de soleil
	pont	pont nuage coucher de soleil

TABLE I. Exemple d'images et de leurs éventuels mots-clés, avant et après extension automatique d'annotations

IV. Description des images

Dans cette section, nous présentons les caractéristiques visuelles que nous avons utilisées : un descripteur de forme, et un de couleur. Notre choix s'est porté sur ces caractéristiques car elles étaient facilement disponibles et/ou rapides à implémenter. De plus, ce choix n'est pas vraiment important dans le sens où le but de cet article est de montrer que la combinaison de caractéristiques visuelles et sémantiques améliore le taux de reconnaissance, quelles que soient les caractéristiques visuelles utilisées.

A. Un descripteur de forme : La $\mathcal{R}$-signature $1D$

La $\mathcal{R}$-signature $1D$ [24] 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. 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 est créée. 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.
La $\mathcal{R}$-signature $1D$ nous fournit un vecteur de 180 caractéristiques par image.

B. Un descripteur de couleur : l'histogramme des composantes RGB

Il existe un grande nombre de modes de représentation de l'espace des couleurs (par exemple RGB et HSI). L'espace RGB a été largement utilisé grâce à la grande disponibilité d'images au format RGB à partir d'images scannées. Quel que soit l'espace de représentation, l'information couleur d'une image peut être représentée par un seul histogramme $3D$ ou 3 histogramme $1D$ [25]. Ces modes de représentation de la couleur ont l'avantage d'être invariants à la translation et à la rotation. De plus une simple normalisation de l'histogramme fournit aussi

l'invariance à l'échelle. Finalement, pour chaque image, le vecteur de caractéristiques couleur correspond à la concaténation des 3 histogrammes $1D$ normalisés (un pour chaque composante R, G et B), de 16 valeurs chacun. On obtient ainsi un vecteur caractéristiques de 48 valeurs par image.

V. Réduction de dimensionnalité

Une fois que les deux descripteurs présentés section IV ont été calculés sur l'ensemble des images de la base, nous disposons d'un vecteur caractéristique de dimension 228, par image.

Cette large dimension des vecteurs de caractéristiques visuelles engendre un problème de dimensionnalité. En effet, une trop grande dimension des vecteurs caractéristiques augmente le temps de calcul de la classification et un mauvais apprentissage des mélanges de Gaussiennes, car il y a une disproportion entre la taille de l'échantillon d'apprentissage et la dimension des vecteurs (problème "Small Sample Size" (SSS)). Pour pallier ce problème et réduire la complexité de notre méthode, nous avons utilisé une méthode de réduction de dimensionnalité. De nombreuses méthodes ont été proposées dans la littérature, afin de réduire la dimension des vecteurs [26], [27]. Parmi les méthodes de réduction de dimension, nous nous intéressons plus particulièrement aux méthodes de sélection de caractéristiques, car elles permettent de réduire la dimension, tout en sélectionnant un sous-ensemble des caractéristiques initiales, contrairement aux méthodes qui réduisent la dimension mais en fournissant de nouvelles variables issues de combinaisons des variables initiales [28]. Les méthodes de sélection de variables sont donc plus appropriées à notre problème, car notre objectif est de réduire le nombre de variables caractéristiques, afin de réduire la taille de notre réseau et la complexité de notre méthode. Les méthodes de sélection de variables les plus populaires sont des heuristiques basées sur des parcours séquentiels, consistant à rajouter ou éliminer itérativement des variables [29]. Dans ces approches, il est possible de partir d'un ensemble de variables vide et d'ajouter des variables à celles déjà sélectionnées (il s'agit de la méthode Sequential Forward Selection (SFS) [30]) ou de partir de l'ensemble de toutes les variables et d'éliminer des variables parmi celles déjà sélectionnées (dans ce cas on parle de Sequential Backward Selection (SBS)). Ces méthodes sont connues pour leur simplicité de mise en œuvre et leur rapidité. Néanmoins, elles sont aussi connues pour leur instabilité. De plus, comme elles n'explorent pas tous les sous-ensembles possibles de variables et ne permettent pas de retour arrière pendant la recherche, elles sont donc sous-optimales.

Nous avons donc choisi une méthode de sélection de caractéristiques, qui permet d'extraire les caractéristiques les plus pertinentes et discriminantes, avec une perte minimale d'information. La méthode de régression, appelée LASSO (Least Absolute Shrinkage and Selection Operator) [31], a été choisie pour sa stabilité et sa facilité de mise en œuvre. De plus, cette méthode permet de sélectionner des variables, et prend en compte les valeurs de la variable classe afin de sélectionner un sous-ensemble de variables. Dans la section VI, nous comparerons les résultats obtenus sur notre base d'images, par le LASSO, par rapport aux résultats obtenus par la méthode SFS, qui reste un des plus populaires.

Le principe du LASSO est de réduire les coefficients de régression en imposant une pénalité sur leur taille (on parle également de méthode de rétrécissement). Ces coefficients minimisent 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

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

La forme linéaire du LASSO a été utilisée dans une étape de prétraitement, sur les caractéristiques visuelles, totalement indépendamment de notre classificateur Bayésien. Pour adapter cette méthode à notre problème, nous considérons nos données d'apprentissage où y_i représente la somme des caractéristiques du vecteur moyen de la classe c_i, et $x_j = \{x_{j_1}, ..., x_{j_p}\}$ les p caractéristiques de l'individu j. Ensuite, seul le sous-ensemble de variables sélectionnées est utilisé dans notre modèle.
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 vont s'annuler. On choisit pour s la valeur maximale telle que au moins un des coefficients β_j s'annule. Les variables correspondant aux coefficients non nuls sont sélectionnées. Les solutions du LASSO ont été calculées avec l'algorithme "Least Angle Regression" (LAR) [32]. Cet algorithme exploite la structure particulière du LASSO, et fournit un moyen efficace de calculer simultanément les solutions pour toutes les valeurs de s.

Ainsi, nous distinguons deux phases indépendantes dans notre méthodes : dans un premier temps le LASSO est utilisé pour sélectionner les caractéristiques visuelles les plus pertinentes. Ensuite, notre modèle de mélange de Gaussiennes et de lois multinomiales est utilisé pour classifier de nouvelles images (images requêtes), représentées par l'ensemble des caractéristiques visuelles pré-sélectionnées et les éventuels mots-clés.

VI. Résultats expérimentaux

Dans cette section, nous présentons une évaluation de notre modèle sur plus de 3000 images provenant d'Internet, et fournies par Kherfi et al. [17]. Ces images sont réparties en 16 classes. Par exemple, la Figure 4 présente quatre images de la classe "cheval".

Fig. 4. Exemples d'images de la classe "cheval"

Cette base nous a été fournie partiellement annotée : 65% de la base est annotée par 1 mot-clé, 28% par 2 mots-clés et 6% par trois mots-clés, en utilisant un vocabulaire de 39 mots-clés. La notion de mot-clé est à différencier de la notion de classe. En effet, un même mot-clé peut être présent dans les annotations d'images de classes différentes. Par exemple, le mot-clé nature annote plusieurs images des classes "feuilles", "fleurs" et nature. De même, on pourra trouver le mot-clé "eau" dans les annotations d'images des classes "chutes d'eau", "ponts" et "mer". Enfin, plusieurs classes se chevauchent, i.e. certaines images de la base font partie de deux classes différentes : par exemple certaines images de la classe "feuilles" peuvent faire partie de la classe "forêts" et *vice versa*. Les images annotées ont été choisies aléatoirement et les mots-clés sont distribués non uniformément, parmi les images et les classes : c'est-à-dire que toutes les images ne sont pas annotées par le même nombre de mots-clés. De même toutes les classes n'ont pas le même nombre d'images annotées. Par exemple, parmi les quatre images de la Figure 4, la première est annotée par 2 mots-clés, "animal" et "cheval". La seconde est annotée par 1 mot-clé seulement : "animal". Les deux autres images n'ont aucune annotation. On rappelle que les caractéristiques visuelles utilisées sont issues d'un descripteur de couleur, un histogramme de couleurs, et d'un descripteur de forme basé sur la transformée de Radon. Notre méthode a été évaluée en effectuant cinq validations croisées, dont chaque proportion de l'échantillon d'apprentissage est fixée à 25%, 35%, 50%, 65% et 75% de la base. Les 75%, 65%, 50%, 35% et 25% respectivement restants sont retenus pour l'échantillon de test. Dans chaque cas, les tests ont été répétés 10 fois, de façon à ce que chaque observation ait été utilisée au moins une fois pour l'apprentissage et les tests. Pour chacune des 5 tailles de l'échantillon d'apprentissage, nous calculons le taux de reconnaissance moyen en effectuant la moyenne des taux de reconnaissance obtenus pour les 10 tests. Dans tous les tests, notre modèle de mélange GM-Mult a été exécuté avec des mélanges de 2 Gaussiennes et des matrices de covariances diagonales.

Tout d'abord, considérons le tableau II, présentant le nombre de variables sélectionnées pour chaque descripteur avec la méthode du LASSO, comparé à celui obtenu avec une autre méthode de sélection de variables : "Sequential Forward Selection" (notée SFS) [30]. Ce tableau montre que la sélection de variables avec la méthode du LASSO nous a permis de réduire significativement le nombre de variables issues de la $\mathcal{R}$-signature $1D$. Par contre, seulement 3 variables ont été supprimées de l'ensemble de caractéristiques issues de l'histogramme de couleur. Ceci peut s'expliquer par la faible dimension initiale de l'histogramme de couleurs (16 valeurs pour chaque composante R, G et B) : globalement, il y a plus de variables pertinentes et moins de redondance entre les variables, dans un ensemble de petite taille. De plus, nous pouvons constater que la méthode du LASSO nous a permis de sélectionner plus de variables que la méthode SFS. En effet, la méthode SFS sélectionne les variables en les ajoutant une par une, de manière itérative, à un ensemble de variables déjà sélectionnées. Au contraire, le LASSO est une méthode d'optimisation globale : il vise à faire émerger des sous-groupes de variables pertinentes et sélectionne, de ce fait, plus de variables que les méthodes itératives.

Le tableau III montre l'impact de la méthode de sélection de variables sur la qualité de la classification. De façon à mesurer cet impact, une classification a été effectuée sur les caractéristiques visuelles avec notre modèle

Nombre de variables	Descripteur couleur	Descripteur de forme
Sans sélection	48	180
SFS	11	7
LASSO	45	23

TABLE II. Nombre moyen de variables en fonction de la méthode de sélection de variables

(noté mélange GM-Mult), et trois autres classificateurs : un classificateur SVM classique [33], un algorithme flou des k plus proches voisins (noté FKNN) [34] et le modèle de mélange de lois multinomiales et Gaussiennes (noté GM-Mixture) [11]. Ces classificateurs ont été choisis pour leur aptitude à traiter à la fois les données discrètes et continues et leur efficacité en présence de grandes dimensions. Le modèle GM-Mixture présente les avantages supplémentaires d'être efficace en présence de données manquantes et de pouvoir être utilisé en annotation. De plus, il est très proche de notre modèle de mélange GM-Mult. Le modèle GM-Mixture a été utilisé sans segmentation des images : le descripteur de couleur et celui de forme ont été calculés sur les images entières, et les mots-clés sont également associés aux images entières. De plus, comme nous considérons, dans ce papier, un problème de classification supervisé, la variable discrète z, utilisée dans [11] pour représenter la classification jointe d'une image et de sa légende, n'est pas cachée pour les images des échantillons d'apprentissage. De même le nombre de clusters est connu. En fait, cette variable discrète z correspond à notre variable classe "Classe". Nous avons comparé les taux de reconnaissance pour ces 4 classificateurs sans sélection de variables préalable et après la sélection d'un sous-ensemble de variables avec les méthodes SFS et LASSO. L'algorithme flou des k plus proches voisins a été exécuté avec $k = 1$ et $k = m$, où m désigne le nombre moyen d'images par classe dans l'échantillon d'apprentissage. De plus, les résultats du tableau III montrent que la sélection de variables avec la méthode LASSO améliore le taux de reconnaissance de 1.8% en moyenne par rapport à celui obtenu sans sélection de variables préalable, et de 6.9% en moyenne comparé à celui obtenu après sélection de variables avec la méthode SFS. De plus, ce résultat est vérifié quel que soit le classificateur utilisé. En effet, les méthodes de rétrécissement comme le LASSO (le LASSO, en plus de la sélection de groupes de variables, opère un rétrécissement sur les variables à l'intérieur de ces groupes (voir section V)) sont réputées pour être plus stables que les méthodes itératives, pour sélectionner des variables dans un grand ensemble de variables mais avec peu d'exemples. Ainsi, la méthode du LASSO s'est montrée plus robuste expérimentalement, sur cette base d'images, que la méthode SFS. Enfin, nous pouvons remarquer que l'utilisation du LASSO a permis de réduire significativement les temps de calculs (tableau VI). Ainsi, seules les variables sélectionnées avec la méthode du LASSO ont été utilisées dans la suite des expérimentations.

Considérons maintenant le tableau IV. La notation "C + F" signifie que les descripteurs de forme et de couleur ("C" pour couleur et "F" pour forme) ont été combinés. La notation "C + F + KW" indique la combinaison des informations visuelles et textuelles. Les taux de reconnaissance confirment que la combinaison des caractéristiques visuelles et sémantiques est toujours plus performante que l'utilisation d'un seul type d'information. En effet, nous observons que la combinaison des caractéristiques visuelles et des mots-clés (quand ils sont disponibles) augmente

Méthode de sélection de variables	SVM	FKNN $k = 1$	FKNN $k = m$	GM-Mixture	**mélange GM-Mult**
Sans sélection	32.2	43.2	39	36.1	40.7
SFS	30.5	33.7	35	32.5	33.8
LASSO	32.6	44.1	39.3	38.9	45.2

TABLE III. Taux de reconnaissance moyens (en %), en classification visuelle, pour les classificateurs SVM, FKNN, GM-Mixture et le mélange GM-Mult, en fonction de la méthode de sélection de variables

le taux de reconnaissance d'environ 38.5% comparé aux résultats obtenus avec le descripteur couleur seul, de 58% comparé à la classification basée sur le descripteur de forme et de 37% par rapport à la classification utilisant uniquement l'information textuelle. De plus, on peut noter que pour toutes les expérimentations, combiner les deux descripteurs visuels apporte en moyenne une amélioration de 16% du taux de reconnaissance, comparé à l'utilisation d'un seul. Enfin, la classification visuo-textuelle montre une amélioration d'environ 32.3% en terme de taux de reconnaissance, par rapport à la classification basée sur l'information visuelle seule.

Spécifications		Couleur	Forme	Mots-clés	C + F	C + F + KW
proportion apprentissage	proportion test					
25%	75%	35	17.8	36.6	39.4	69.7
35%	65%	36.9	18.1	38.9	42.2	74.4
50%	50%	38.7	18.5	41.1	45	79.1
65%	35%	41.1	20.6	41.5	46.6	81.7
75%	25%	43.5	21.8	45.1	52.9	82.9

TABLE IV. Taux de reconnaissance (en %) de la classification visuelle vs. classification visuo-textuelle (avec mélange GM-Mult)

Ensuite, le tableau V montre l'efficacité de notre approche (mélange GM-Mult) comparée aux classificateurs SVM, FKNN et GM-Mixture. Les résultats ont été obtenus en utilisant les deux caractéristiques visuelles et les éventuels mots-clés associés. Il apparaît que les résultats du mélange GM-Mult sont meilleurs que ceux du SVM, du FKNN et du GM-Mixture. Plus précisément, le mélange GM-Mult se montre sensiblement supérieur aux classificateurs SVM et FKNN. Ces résultats ne sont pas surprenants car les SVM et le FKNN sont peu adaptés au traitement des données manquantes. Par contre, les résultats de notre modèle de mélange GM-Mult sont très proches de ceux obtenus par le modèle GM-Mixture. Ceci est dû à la similarité de ces deux modèles. En effet, dans notre mélange GM-Mult, un mélange de Gaussiennes multivariées est utilisé pour estimer la distribution des caractéristiques visuelles, là où le modèle GM-Mixture utilise une Gaussienne multivariée. Ceci explique aussi la légère supériorité de notre modèle, plus précis. Cette différence de précision se révèle plus significative dans l'extension d'annotations.

Enfin, le tableau VI montre les temps CPU du modèle GM-Mixture comparés à ceux de notre modèle de mélange

Spécifications		SVM	FKNN $k=1$	FKNN $k=m$	GM-Mixture	**mélange GM-Mult**
proportion apprentissage	proportion test					
25%	75%	38.3	59.1	58.5	68.1	69.7
35%	65%	41.3	62.3	58.3	73.9	74.4
50%	50%	39.9	68.2	58.2	75.9	79.1
65%	35%	40.5	72.9	67	80.7	81.7
75%	25%	41.9	73.2	69.3	81	82.9

TABLE V. Taux de reconnaissance (en %) des classificateurs SVM, FKNN et GM-Mixture vs. notre mélange GM-Mult

GM-Mult, pour les phases d'apprentissage et de test, dans les mêmes conditions expérimentales que dans le tableau V. Les expérimentations ont été menées sur PC doté d'un processeur Intel Core 2 Duo 2,40 GHz, 2 Go RAM, Windows OS. Les deux classificateurs ont été exécutés avec Matlab©. Le temps CPU est plus élevé pour le modèle de mélange GM-Mult, car il dépend du nombre de Gaussiennes (dans ce cas, 2) et de la précision de l'algorithme EM, mais il reste faible de l'ordre de 0.04s par image en phase de classification hors apprentissage. Le modèle GM-Mixture est plus rapide car il n'utilise qu'une Gaussienne multivariée. Cependant, la différence en temps de calcul entre les deux méthodes est négligeable (inférieure à 0.015s par image).

App	GM-Mixture sans SV		**mélange GM-Mult** sans SV		GM-Mixture avec SV		**mélange GM-Mult** avec SV	
	app	test	app	test	app	test	app	test
25%	91.1	86.5	137.3	145	80.8	53.7	101.5	77
35%	127.2	80.6	178.4	128.3	106	47.9	134.3	71.7
50%	177	56.7	284.3	113	156.5	34.9	202.8	55.5
65%	255.9	40.5	388.4	75.3	209.2	25.1	277.7	37.4
75%	263.4	30.3	460.7	58.5	227.6	15	300	26.5

TABLE VI. Temps CPU (en secondes) du modèle GM-Mixture vs. modèle mélange GM-Mult, avec ou sans sélection de variables avec le LASSO (SV). Les temps CPU sont donnés pour la classification visuo-textuelle de toutes les images test

Considérons donc maintenant le problème d'extension d'annotations. Il est nécessaire que chaque annotation comprenne au moins un mot-clé pour comparer les annotations après l'extension automatique d'annotations à la vérité terrain. 99% de la base d'images, annotée par au moins 1 mot-clé (pour rappel 65% de la base est annotée par 1 mot-clé, 28% par 2 mots-clés et 6% par 3 mots-clés), a donc été sélectionnée comme vérité terrain. Afin d'évaluer la qualité de l'extension d'annotations, une validation croisée a été effectuée. La proportion de chaque échantillon d'apprentissage est fixée à 50% du sous-ensemble pré-selectionné de la base comme vérité-terrain. Les 50% restants sont retenus pour l'échantillon de test. Les tests ont été répétés 10 fois, de façon à ce que chaque observation ait été utilisée au moins une fois pour l'apprentissage et les tests. Le taux moyen de bonnes annotations

est obtenu en effectuant la moyenne des taux de bonnes annotations obtenus pour les 10 tests. Pour chaque test, le taux de bonnes annotations correspond à la proportion de mots-clés corrects parmi les mots-clés obtenus par extension. Le tableau VII compare les taux de bonnes annotations obtenus par notre approche (mélange GM-Mult) par rapport au modèle de mélange de lois multinomiales et Gaussiennes (GM-Mixture). On observe que notre modèle est sensiblement meilleur que le modèle GM-Mixture. Comme en classification, cette supériorité est due au fait que notre modèle utilise un mélange de Gaussiennes multivariées pour estimer la distribution des caractéristiques visuelles, là où le modèle GM-Mixture utilise une Gaussienne multivariée. De plus, le tableau VIII montre les temps CPU du modèle GM-Mixture comparés à ceux de notre modèle de mélange GM-Mult, pour les phases d'apprentissage et de test, dans les mêmes conditions expérimentales que dans le tableau VII. Les deux classificateurs ont été exécutés avec Matlab©. Le temps CPU de notre modèle GM-Mult est environ 2 fois supérieur à celui du modèle GM-Mixture, pour les mêmes raisons que celles évoquées précédemment (nombre de gaussiennes et précision dans l'algorithme EM) mais il reste inférieur à 0.04s par image. Compte tenu de l'écart significatif entre les taux de bonnes annotations des deux méthodes, il nous semble que le compromis entre précision d'annotation et temps de calcul est meilleur pour notre modèle de mélange GM-Mult.

GM-Mixture	**mélange GM-Mult**
36	77.5

TABLE VII. Taux moyens de bonnes annotations (en %), obtenus par le modèle GM-Mixture vs. notre modèle mélange GM-Mult

GM-Mixture		**mélange GM-Mult**	
apprentissage	test	apprentissage	test
121.3	27.6	231.5	54

TABLE VIII. Temps CPU (en secondes) du modèle GM-Mixture vs. modèle mélange GM-Mult. Les temps CPU sont donnés pour l'extension d'annotations de toutes les images test

Enfin, des annotations ont été ajoutées automatiquement à toutes les images de la base de façon à ce que chacune soit annotée par trois mots-clés. Puis, afin d'évaluer la qualité de cette extension d'annotations, la classification visuo-textuelle a été répétée avec les mêmes spécifications que dans le tableau IV. Le tableau IX montre l'efficacité de notre extension automatique d'annotations. En effet, les taux de reconnaissance après l'extension d'annotations sont toujours meilleurs qu'avant. De plus, l'extension automatique d'annotations améliore le taux de reconnaissance de 6.8% en moyenne.

Spécifications		Avant extension d'annotations	Après extension d'annotations
proportion apprentissage	proportion test		
25%	75%	69.7	77
35%	65%	74.4	79.3
50%	50%	79.1	85.4
65%	35%	81.7	87.6
75%	25%	82.9	92.7

TABLE IX. Taux de reconnaissance (en %) de la classification visuo-textuelle (avec mélange GM-Mult) avant et après extension automatique d'annotations

VII. Conclusion et perspectives

Nous avons proposé un modèle efficace permettant de combiner l'information visuelle et textuelle, de traiter les données manquantes et d'étendre des annotations existantes à d'autres images. Afin de diminuer la complexité de notre méthode, nous avons adapté une méthode de sélection de caractéristiques, qui a montré expérimentalement son efficacité. Nos expérimentations ont été effectuées sur une base d'images partiellement annotées provenant d'Internet. Les résultats montrent que la classification visuo-textuelle a amélioré le taux de reconnaissance comparée à la classification basée sur l'information visuelle seule. De plus, notre réseau Bayésien a été utilisé pour étendre des annotations à d'autres images, ce qui a encore amélioré le taux de reconnaissance. Enfin, la méthode proposée s'est montrée compétitive par rapport à des classificateurs classiques, aussi bien en classification qu'en extension automatique d'annotations. Les futurs travaux seront dédiés à la considération, dans le réseau, d'éventuelles relations sémantiques entre mots-clés, de façon à intégrer une hiérarchie de concepts sémantiques dans la description des images.

References

[1] O. Terrades, S. Tabbone, and E. Valveny, "A review of shape descriptors for document analysis," *International Conference on Document Analysis and Recognition*, vol. 1, pp. 227–231, 2007.

[2] L. Wendling, J. Rendek, and P. Matsakis, "Selection of suitable set of decision rules using choquet integral," in *SSPR/SPR*, 2008, pp. 947–955.

[3] V. Gunes, M. Ménard, P. Loonis, and S. Petit-Renaud, "Combination, cooperation and selection of classifiers: A state of the art," *IJPRAI*, vol. 17, no. 8, pp. 1303–1324, 2003.

[4] O. R. Terrades, E. Valveny, and S. Tabbone, "Optimal classifier fusion in a non-bayesian probabilistic framework," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 9, pp. 1630–1644, 2009. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2008.224

[5] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, December 2000. [Online]. Available: http://dx.doi.org/10.1109/34.895972

[6] K. Barnard, P. Duygulu, D. Forsyth, N. De Freitas, D. M. Blei, and M. I. Jordan, "Matching words and pictures," *Journal of Machine Learning Research*, vol. 3, no. 6, pp. 1107–1135, 2003.

[7] W. I. Grosky and R. Zhao, "Negotiating the semantic gap: From feature maps to semantic landscapes," in *SOFSEM '01*, 2001, pp. 33–52.

[8] A. Benitez and C. Shih-Fu, "Perceptual knowledge construction from annotated image collections," *in ICME '02*, vol. 1, pp. 189–192, 2002.

[9] Y. Gao, J. Fan, X. Xue, and R. Jain, "Automatic image annotation by incorporating feature hierarchy and boosting to scale up svm classifiers," in *ACM MULTIMEDIA '06*, 2006, pp. 901–910.

[10] C. Yang, M. Dong, and J. Hua, "Region-based image annotation using asymmetrical support vector machine-based multiple-instance learning," in *CVPR '06*, 2006, pp. 2057–2063.

[11] D. M. Blei and M. I. Jordan, "Modeling annotated data," in *SIGIR '03*, 2003, pp. 127–134.

[12] S. Feng, R. Manmatha, and V. Lavrenko, "Multiple bernoulli relevance models for image and video annotation," *in CVPR '04*, vol. 2, pp. 1002–1009, 2004.

[13] C. Wang, F. Jing, L. Zhang, and H.-J. Zhang, "Image annotation refinement using random walk with restarts," in *ACM MULTIMEDIA '06*, 2006, pp. 647–650.

[14] X. Rui, M. Li, Z. Li, W.-Y. Ma, and N. Yu, "Bipartite graph reinforcement model for web image annotation," in *MULTIMEDIA '07*, 2007, pp. 585–594.

[15] R. Zhang, Z. M. Zhang, M. Li, W.-Y. Ma, and H.-J. Zhang, "A probabilistic semantic model for image annotation and multi-modal image retrieval," in *ICCV '05*, vol. 1, 2005, pp. 846–851.

[16] R. Jin, J. Y. Chai, and L. Si, "Effective automatic image annotation via a coherent language model and active learning," in *MULTIMEDIA '04*, 2004, pp. 892–899.

[17] M. L. Kherfi, D. Brahmi, and D. Ziou, "Combining visual features with semantics for a more effective image retrieval," in *ICPR'04*, vol. 2, 2004, pp. 961–964.

[18] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[19] L. Breiman, "Random forests," in *Machine Learning*, 2001, pp. 5–32.

[20] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[21] M. I. Jordan, Ed., *Learning in graphical models*. Cambridge, MA, USA: MIT Press, 1999.

[22] J. H. Kim and J. Pearl, "A computational model for combined causal and diagnostic reasoning in inference systems," in *IJCAI-83*, 1983, pp. 190–193.

[23] C. Robert, *A decision-Theoretic Motivation*. Springer-Verlag, 1997.

[24] S. Tabbone and L. Wendling, "Technical symbols recognition using the two-dimensional radon transform," in *ICPR'02*, vol. 2, aug 2002, pp. 200–203.

[25] M. J. Swain and D. H. Ballard, "Color indexing," *Int. J. Comput. Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[26] T. Denoeux and M.-H. Masson, "Dimensionality reduction and visualization of interval and fuzzy data: a survey," in *Proceedings of the 56th session of the International Statistical Institute (ISI Ś07)*, Lisboa, Portugal, August 2007.

[27] M. Piccardi, H. Gunes, and A. F. Otoom, "Maximum-likelihood dimensionality reduction in gaussian mixture models with an application to object classification," in *ICPR'08*, 2008, pp. 1–4.

[28] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, Second Edition*. Wiley-Interscience, 2001.

[29] P. A. Devijver and J. Kittler, *Pattern recognition: A statistical approach*. Prentice Hall, 1982.

[30] P. Pudil, J. Novovičová, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognition Letters*, vol. 15, no. 11, pp. 1119–1125, 1994.

[31] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[32] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of Statitics*, vol. 32, pp. 407–499, 2004.

[33] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001. [Online]. Available: http://www.csie.ntu.edu.tw/ cjlin/libsvm

[34] J. Keller, M. Gray, and J. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15, no. 4, pp. 580–585, 1985.