Fusion après sélection mot-clé dépendante des traits visuels et de leur hétérogénéité par LDA approximée pour filtrer l'indexation d'images

Word dependant fusion and selection of visual features and their heterogeneity by approximation of LDA for image keyword filtering

Sabrina Tollari(1), Hervé Glotin(1), Pascale Giraudet(2)
(1)Laboratoire LSIS UMR CNRS 6168
(2)Département de biologie
Université du Sud Toulon-Var
BP 20132, F-83957 La Garde cedex
{tollari,glotin,giraudet}@univ-tln.fr

Résumé

Pour améliorer les performances des systèmes automatiques d'annotations d'images à partir de données réalistes, nous proposons tout d'abord une méthode permettant de sélectionner les traits visuels les plus discriminants pour un mot-clé donné. Cette méthode est basée sur une analyse factorielle discriminante (LDA), mais étendue par approximation (ALDA) au cas de bases mal annotées comme le sont les grandes bases d'images. Nous validons l'ALDA théoriquement et expérimentalement sur COREL (10000 images). D'autre part, nous proposons de dériver pour chaque trait visuel leur hétérogénéité (ou entropie) dans chaque image associée à chaque concept visuel (ou mot clé). Nous appliquons l'ALDA sur ce nouveau jeu de traits, et nous mesurons pour chaque mot les gains de classification après sélection de traits dans chacun des deux espaces, et dans leur fusion. Enfin, nous appliquons notre système de classification adaptative au filtrage de l'indexation d'image. Nous montrons que les deux informations visuelles (classique et hétérogénéité) sont complémentaires. Nous comparons donc des méthodes de fusion précoces et tardives pour améliorer ce filtrage et démontrons un gain de l'ordre de +60% de classification pour une réduction de 80% du nombre de dimensions.

Mots clés

Recherche d'images, indexation multidimensionnelle, CAH, LDA, hétérogénéité

Abstract

In order to enhance real automatic image indexing we propose a method reducing features space. We estimate the most discriminant visual features for a given keyword, by approximating Fisher discriminant analysis (LDA) on the real not well labelled image databases (e.g. where there is many to many relations between visual concept and keyword). Then, we use a non-supervised clustering algorithm to build visual clusters : using all the features of the visual space, or several subspaces made up of the most discriminant features depending of each keyword. Comparisons of indexing scores on COREL show an indexing enhancement going up to 60%, while reducing the number of dimensions of 80%, and show how a meta visual feature called heterogeneity could improve indexation systems.

Keywords

Images retrieval, multi-dimensional indexing, clustering, LDA, heterogeneity

1 Introduction

Les systèmes d'indexation d'images (ou plus largement audiovisuels) se séparent en deux grandes catégories : (1) ceux qui sont dédiés à la détection de concepts spécifiques et (2) ceux qui sont construits pour des données traitant de sujets généralistes. Dans le premier cas, les systèmes sont spécialisés pour le traitement d'images dans le but d'en tirer une information précise, comme la présence de certains objets (voitures, armes, visages...) ce qui permet d'optimiser les procédures de traitements visuels. Dans le second cas, les systèmes sont construits pour donner les meilleurs résultats pour tous les concepts en moyenne, mais pas pour chaque concept pris indépendamment.

Nous constatons que peu de systèmes modélisent efficacement le couplage visuo-textuel pour permettre une indexation et une recherche d'informations réellement adaptées aux images. Cette pénurie peut être due à difficulté de modéliser les relations entre les attributs

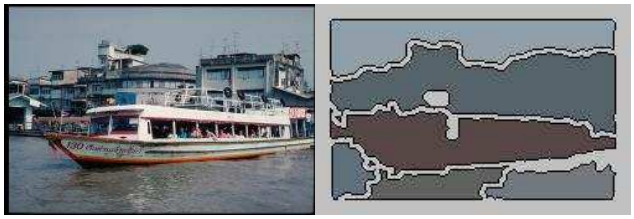

FIG. 1 – Exemple de segmentation d'une image. Chaque segment est appelé « blob ». Chaque image de la base COREL est annotée manuellement par des mots de référence (pour cette image, « water », « boat », « harbor » et « building »).

visuels et les mots d'une requête sémantique de l'utilisateur (paradigme dit du « fossé sémantique »), qui est d'autant plus délicate à aborder que les données pour la construction des modèles ne semblent pas encore adaptées au problème. Dans ce cadre, certains travaux récents proposent des méthodes lourdes d'indexation textuelle automatique d'images par le contenu visuel (auto-annotation) [3], champs de Markov [15, 29], Probabilistic Latent Semantic Analysis [18], systèmes de traduction de langues appliqués sur les espaces textuel et visuel [13]. Les performances de ces systèmes dépassent rarement de 25% un système aléatoire et ne sont donc pas actuellement applicables pour le grand public. Nous avons proposé un autre type de système filtrant l'information textuelle par le contenu visuel d'une image ou bien encore de fusions précoces et tardives des informations textuelles et visuelles [24, 23, 22], mais ces systèmes restent lourds de par la grande dimensionnalité de l'espace visuel traité. Aucun des systèmes cités ne propose une sélection adaptative des traits visuels en amont de leur traitement stochastique, alors que leur complexité est non linéaire en fonction du nombre de traits. Cette lacune peut être due à l'inexistence de vastes[1] bases de données précisément annotées, c'est-à-dire spécifiant une relation univoque entre un mot-clé et une région ou un objet de l'image. En effet, la base la plus largement diffusée pour ces qualités et largement utilisée pour le développement des modèles précités est une base d'images généralistes nommée COREL [27, 19, 28] qui ne fournit qu'une indexation textuelle globale pour chaque image et non un mot par objet de l'image.
Dans la base COREL traité par Arizona Université chaque image est segmentée automatiquement par l'algorithme « Normalized Cuts » [21] en régions appelées « blob », et chaque blob est associé à un vecteur de 40 dimensions comportant divers types de traits visuels : couleurs, textures, formes. Ces chercheurs font cependant bien remarquer : « It remains an interesting open question to construct feature sets that (...) offer very good performance for a particular vision task » [3].

[1] plus de 10000 images, 200 mots-clés

En effet, il est difficile de prédire à partir de ce genre de matériel quels traits sont vraiment pertinents ou informatifs pour un concept donné.
Dans cet article, nous montrons comment on peut améliorer les performances de notre modèle en sélectionnant automatiquement les traits visuels les plus discriminants pour le mot traité. Pour cela, nous utilisons une méthode statistique simple qui permet de réduire l'espace visuel aux traits les plus pertinents pour un mot-clé donné. Nous validons notre méthode sur une base de données de l'état de l'art qui théoriquement ne se prête pas à l'application de cette méthode, mais qui pratiquement est efficace grâce au grand nombre de données traitées.
Nous proposons donc dans ce papier de (i) ramener automatiquement l'espace visuel dimensionnel élevé aux traits les plus efficaces, et simultanément de le (ii) compléter en dérivant une nouvelle caractéristique propre à une analyse contextuelle : l'hétérogénéité. Enfin, une application au filtrage des légendes textuelles des images à l'aide du visuel est présentée.

2 Sélection des traits visuels

Des travaux récents ont posé le problème de la sélection des descripteurs efficaces qui contiennent un nombre minimum de dimensions pour permettre des calculs et une recherche rapide [4]. [12] ont montré qu'il est possible par le critère du Maximum Entropy Discrimination de sélectionner les traits au sein d'un processus de classification linéaire ou de régression. D'autres travaux traitant du sujet [11] ont déjà montré par une méthode de fouilles de données images que la réduction du nombre de dimensions de l'espace visuel n'affectait pas forcement l'efficacité des associations signal/symbole, mais sur une base de données de quelques centaines d'images seulement.
Dans un système de recherche d'information basé sur le contenu d'images, l'utilisateur peut fournir une image comme requête et rechercher des images semblables (recherche de l'image la plus proche (NNS)). Mais ces dispositifs multidimensionnels NNS ne sont pas efficaces du fait du problème des grandes dimensions [2, 5]. Un autre mode de recherche est basé sur une requête textuelle sur des images classées par mot-clé. Ce mode serait efficace si l'indexation d'image pouvait être faite automatiquement et correctement, mais ce n'est pas le cas. En fait des images sont grossièrement classées par des moteurs de recherche sur le web à partir des mots de la page contenant l'image [6]. Cependant, si la recherche d'images basée sur leur contenu (CBIR) a motivé de nombreux travaux [3, 15, 29, 24], elle reste très délicate et actuellement inopérante sur de grandes bases d'images. Une des explications de ce relatif échec repose sur l'absence de base de données d'images finement annotés avec des mots-clés marquant l'objet dans l'image (également

appelés visems). Les bases existantes sont marquées de façon peu précise comme illustré dans figure 1.
Une des approches récentes pour réduire l'espace sémantique est l'Apprentissage Actif, demandant à l'utilisateur de marquer quelques images les plus proches de la frontière d'un classificateur comme par exemple un SVM [25, 10]. Malheureusement, cet apprentissage actif requière des rétroactions manuelles d'utilisateur, des centaines pour environ seulement 10 visems [10], et donc cette méthode ne peut pas être appliquée à une grande base de données d'images avec un grand lexique de visems.
Dans cet article, nous proposons d'augmenter automatiquement la pertinence de l'espace visuel en fonction de la sémantique traitée par une étape de prétraitement avant indexation de classification, sans aucune intervention d'utilisateur. D'ailleurs, notre stratégie ne se fonde pas sur un choix des images difficiles à classifier, mais sur la relation entre traits visuels et la sémantique visuelle de mot-clé.
La méthode la plus célèbre de réduction de dimensionnalité est l'Analyse en Composantes Principales (ACP). Mais l'ACP n'inclut pas l'information de classification des données. Bien que l'ACP trouve les composants qui sont utiles pour représenter les données, il n'y a aucune raison de supposer que ces composants doivent être utiles pour distinguer les données dans différentes classes. L'Analyse Discriminante Linéaire de Fisher (LDA) permet de trouver la meilleure projection pour une classification ou indexation. Là où une ACP cherche les composantes efficaces pour la représentation, la LDA cherche les directions qui sont efficaces pour la discrimination [7]. Ainsi, dans la prochaine section nous adaptons la LDA en l'approximant (ALDA) au cas réel des visems incorrectement annotés.

3 Sélection de traits visuels mots-clés dépendantes sur des données mal indexées

Du fait du problème des grandes dimensions [2, 5], une bonne indexation visuelle doit être construite à partir des traits visuels qui ont la plus grande capacité discriminante. Mais déterminer quels sont les traits visuels discriminants pour rechercher des concepts dans des images est un problème difficile parce que les données disponibles, souvent mal annotées, ne correspondent pas aux méthodes statistiques traditionnels. Des travaux antérieurs ont montré que des méthodes simples comme la LDA (Analyse Discriminante Linéaire) ou la MMD[2] (Maximum Marginal Diversity) [26] peuvent discriminer un signal acoustique [20] ou bien des traits visuels [30], mais ces méthodes ont été appliquées sur

[2]Une approximation du critère MMD sera discutée dans la conclusion.

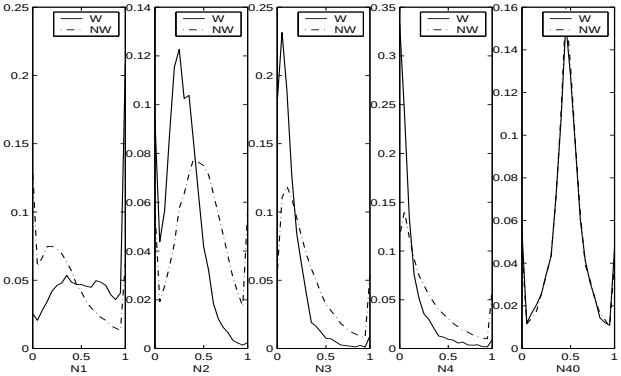

FIG. 2 – Vraisemblances de 5 traits sachant les classes MOT(W) ou NONMOT(NW) du mot-clé « snow ». Les traits sont ordonnés du plus au moins discriminants : N1 (« B » de RGB), N2 (« B » de LAB), N3 (« écart type du A » de LAB), N4 (« écart type du G » de RGS) et N40 (« texture 15 » où W et NW se confondent).

des corpus bien annotée, c'est-à-dire décrivant une relation univoque entre la classe d'un concept et un trait visuel. Or dans les grands corpus d'images, les images sont indexées globalement, et non pas une annotation pour chaque blobs, mais plutôt un ensemble de mots par image (voir figure 1). Nous faisons cependant l'hypothèse suivante : *si la base d'images présente chaque concept avec une variété contextuelle assez large, alors les analyses de variances des traits visuels ne seront significatives que pour le concept récurrent considéré.*
Nous allons donc appliquer la méthode LDA « à la limite » [9] pour déterminer les traits les plus pertinents pour décrire chaque mot-clé de la base COREL, et nous vérifierons la validité des résultats obtenus en mesurant les gains d'une classification hiérarchique ascendante des concepts.
Pour appliquer ces méthodes, nous construisons une bipartition des images d'apprentissage : l'ensemble des images qui sont annotées par ce mot (appelé classe MOT) et l'ensemble des images qui ne sont pas annotées par ce mot (appelé classe NONMOT). La figure 2 donne un exemple des distributions obtenues pour les classes MOT et NONMOT du mot « snow »pour certains des traits visuels.

3.1 Approximation de l'Analyse Discriminante Linéaire

Pour mesurer la dispersion entre les deux classes, la LDA propose de calculer pour chaque mot w_i et pour chaque trait visuel v_j, la variance interclasse $B(v_j; w_i)$ (variance des moyennes de chaque classe) et la variance intraclasse $W(v_j; w_i)$ (moyenne des variances de chaque classe) entre les deux classes MOT et NON-

MOT. Puis, elle calcule pour chaque mot w_i, le pouvoir discriminant $F(v_j; w_i)$ du trait v_j défini par :

$$F(v_j; w_i) = \frac{B(v_j; w_i)}{B(v_j; w_i) + W(v_j; w_i)}. \quad (1)$$

Nous proposons d'utiliser la LDA sur des données mal indexées en calculant, pour chaque mot et pour chaque trait, le pouvoir discriminant approximé $\hat{F}(v_j; w_i)$. Cette méthode, appelée ALDA (Approximation de la LDA), a été théoriquement et expérimentalement démontré dans [9]. La démonstration montre que les erreurs ordonnées due à l'approximation sont petites si les exemples d'apprentissages sont assez nombreux et si le concept considéré est présent avec des contextes variés.

3.2 Estimation du biais de l'ALDA

Nous estimons dans cette partie l'erreur induite dans la valeur des pouvoirs discriminants par l'approximation de l'ALDA par rapport à une LDA qui serait calculée sur une base complètement renseignée notée Q (avec un lien univoque entre chaque mot et chaque objet de l'image).
Nous définissons pour cela 4 ensembles : S, T, T_G and G. S est l'ensemble des valeurs pour un trait visuel calculé sur les images de Q et étiquetées par w_i. On note pour tout ensemble de trait E, c_E son cardinal, μ_E sa moyenne pour toutes les valeurs x_j de $x \in E$ et v_E sa variance. Soit T l'ensemble des valeurs x *de tous les blobs* inclus dans toutes les images étiquetées par w_i. On a T qui contient S. Soit T_G tel que $T = T_G \cup S$ et $T_G \cap S = \emptyset$. On suppose alors $c_{T_G} \neq 0$ Soit G l'ensemble des valeurs de x de tous les blobs contenus dans les images qui ne sont pas étiquetées par w_i. Nous faisons alors les hypothèses suivantes : (hyp. 1) $\mu_{T_G} = \mu_G$ *and* $v_{T_G} = v_G$, ce qui est lié à l'indépendance du contexte d'un concept dans une grande base d'images. On note B_{DE} (resp. W_{DE}) la variance interclasse (resp. la variance intraclasse) entre D et E.
La LDA classique calcule le pouvoir discriminant d'un trait comme : $F(x; w_i) = \frac{1}{1+V(x;w_i)}$ où $V(x; w_i) = \frac{W_{SG}}{B_{SG}}$. On montre alors dans [9], que $\hat{V}(x; w_i) = \frac{W_{TG}}{B_{TG}}$ est une bonne estimation de $V(x; w_i)$, et que l'ordre des V pour un mot w_i est le même que celui des $\hat{V}$, au moins pour les traits les plus discriminants x. En effet la sélection des meilleurs traits pour un mots donné peut être effectuée en calculant $\hat{F}(x; w_i) = \frac{1}{1+\hat{V}(x;w_i)}$.
Nous avons d'après les hypothèses précédentes [9] :

$$\begin{aligned}\hat{V}(x; w_i) &= \frac{\frac{(c_T - c_S + c_G).(c_S + c_G)}{c_G.(c_T + c_G)}.W_{SG} - \frac{c_S.(c_T - c_S)}{c_G.(c_T + c_G)}.v_S}{\frac{c_S.(c_S+c_G)2}{c_T.(c_T+c_G)^2}.B_{SG}} \\ &\quad + \frac{\frac{c_S.(c_T-c_S)}{c_T.(c_T+c_G)}.(\mu_G - \mu_S)2}{\frac{c_S.(c_S+c_G)2}{c_T.(c_T+c_G)^2}.B_{SG}} \\ &= \frac{c_T(c_T - c_S + c_G)(c_T + c_G)}{c_G.c_S(c_S + c_G)} \frac{W_{SG}}{B_{SG}} \\ &\quad + \frac{(c_T - c_S)(c_T + c_G)}{c_S.c_G}\left(1 - \frac{c_T}{c_G}\frac{v_S}{(\mu_G - \mu_S)2}\right)\end{aligned}$$

$$\hat{V}(x; w_i) = A(w_i).V(x; w_i) + B(w_i).\Big(1 - C(x; w_i)\Big)$$

où A et B sont des constantes positives indépendantes de x, fonctions des cardinaux de T, S, G (A et B sont proches de 10 dans COREL).
Donc pour tout mot w_i, $\hat{V}(x; w_i)$ est une fonction linéaire de $V(x; w_i)$ si $C(x; w_i)$ est négligeable par rapport à 1. C'est le cas si (hyp. 2) $\frac{c_T}{c_G}$ est petit, ce qui est vérifié dans COREL (proche de 0.01 pour la plupart des mots, et n'excédant jamais 0.2). Une base peut de toute manière être construite pour que $C_T << C_G$ et si (hyp. 3) v_S est petit devant $(\mu_G - \mu_S)2$ ce qui est le cas quand x est un trait pertinent pour séparer G de S. Donc les ordres de $\hat{V}$ approximent bien ceux de V, en tous les cas pour les premiers rangs.

3.3 Détermination automatique du nombre de traits

Nous déterminons automatiquement pour chaque mot w_i les N dimensions les plus discriminantes (méthode NADAPTτ) telles que, après classement par ordre décroissant des pouvoirs discriminants $\hat{F}$, la somme de leurs pouvoirs discriminants cumule τ% de la somme totale des pouvoirs discriminants de tous les traits pour ce mot :

$$\sum_{j=1}^{N} \hat{F}(v_j; w_i) = \tau \sum_{j=1}^{\delta} \hat{F}(v_j; w_i). \quad (2)$$

4 Construction de traits par hétérogénéité

En utilisant l'ALDA, nous mettons en évidence les dépendances entre les visems et les traits visuels classiques (couleurs, textures, formes). Les travaux en psychovision de J. Martinet ([16, 17] page 117) montrent que le critère d'hétérogénéité appliqué aux surfaces a plus ou moins d'impact sur la description visuelle des objets. La valeur de l'hétérogénéité de la dimension v_j pour l'image d qui contient le blob b_p qui a pour valeur $b_{p,j}$ à la dimension v_j est l'entropie :

$$H_j = -\sum_{b_p \in d} b_{p,j} \times \log_2(b_{p,j}). \quad (3)$$

En nous fondant sur les travaux en neurobiologie de [1], nous proposons d'étendre le concept d'hétérogénéité à tous les traits visuels. En effet, nous pouvons supposer que comme pour le critère de l'aire, certains visems seront consistants sur les traits visuels classiques, et d'autres non. Par exemple, la valeur du trait visuel rouge est stable pour le visem 'tomato', tandis que pour le visem 'market' ce sont les traits d'hétérogénéité qui sont plus discriminants. Nous pouvons supposer de même pour le visem 'people'. Plus généralement, de récents travaux en science cognitive montrent que la perception humaine est basée

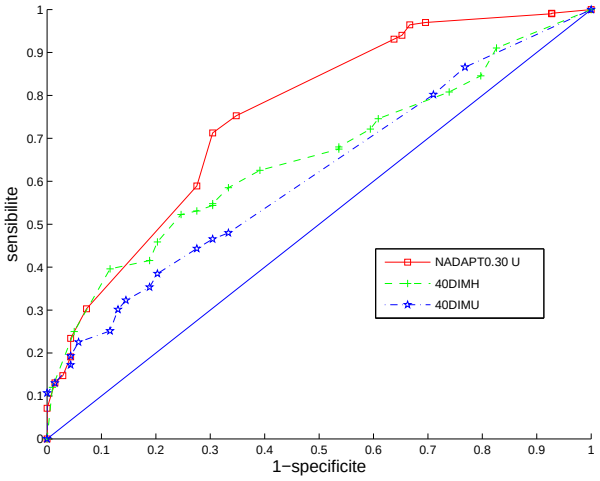

FIG. 3 – ROC de la CAH pour le mot *woman* appliqué par différentes méthodes sur les espaces U et H sur DEV. Entre deux points de la courbe, 5% des points de la courbe sont agrégés par HAC.

sur l'analyse visuelle en contexte. L'hétérogénéité est l'une des caractéristiques qui permet une analyse en contexte. C'est pourquoi nous étendons l'espace visuel par ces nouveaux traits d'hétérogénéité. Nous allons également appliquer l'ALDA sur ce nouvel espace et vérifier expérimentalement si un gain de classification peut être généré en utilisant l'espace d'hétérogénéité.

5 Construction et évaluation de classes visuelles

Une fois que nous avons déterminé pour chaque mot les traits les plus discriminants nous réalisons plusieurs expériences pour valider nos hypothèses. Pour cela, nous construisons d'abord des classes visuelles à l'aide d'un ensemble d'images d'apprentissage, qui nous servent de classes de références, afin de modéliser les dépendances entre les traits visuels pour chaque mot. Puis, nous évaluons la qualité de cette classification, c'est-à-dire de l'association entre un mot et ses classes visuelles, en calculant le score obtenu par classification supervisée des images d'une base de test. La méthode utilisée est décrite plus en détails dans [22], nous la résumons ci-après.

5.1 Construction

Pour construire les classes visuelles d'un mot, nous allons chercher des regroupements de blobs de la base d'apprentissage dans l'espace multidimensionnel visuel au moyen d'une Classification Ascendante Hiérarchique (CAH) [14] (construction non-supervisée de clusters). Pour chaque mot, nous construisons un sous-ensemble de la base d'apprentissage composée des images possédant ce mot, sur lequel nous réalisons une CAH. Nous déterminons alors la valeur d'arrêt de la CAH en choisissant celle qui donne le meilleur score (le calcul du score est expliqué section 5.2). La figure 3 donne des exemples d'apprentissage pour le mot *woman*. Nous gardons alors seulement les classes qui contiennent un nombre significatif de blobs. Nous associons ainsi des classes visuelles à un mot. Chaque classe est représentée uniquement par un couple de vecteurs de mémé dimension :

- le vecteur centroïde de la classe visuelle dans l'espace multidimensionnel,
- le vecteur des écarts types de la classe pour chaque dimension de l'espace,

ainsi que par une constante déterminée de manière empirique pour donner le meilleur score pour chaque mot, optimisée entre la valeur 2 et la valeur 4 (voir [22]).
Le fait que cette méthode de construction de classes visuelles puisse produire plusieurs classes visuelles pour un même mot est intéressant, car un mot peut avoir plusieurs sens, et chacun de ces sens peut avoir plusieurs représentations visuelles. Par exemple, le mot anglais « plants » peut correspondre à un végétal (une plante) ou bien à un bâtiment (une usine). De plus, la plupart des plantes sont vertes, mais elles peuvent être aussi de différentes couleurs.

5.2 Évaluation

Les classes visuelles que nous venons d'obtenir représentent les régions de l'espace visuel correspondant aux caractéristiques visuelles possibles d'un mot. C'est pourquoi lors de la phase de test, un mot sera associé à un blob d'une image de l'ensemble de test si le vecteur visuel du blob appartient à l'une des classes visuelles du mot. Un mot sera associé à une image de test si au moins un blob de l'image a été associé avec ce mot. Chaque image de la base possède initialement un ensemble de mots de référence, nous pouvons donc faire un calcul de score. Nous utilisons le score « Normalized Score » (noté NS par la suite) employé dans [3, 18]. Le score NS est défini par :

$$\begin{aligned} NS &= \frac{right}{n} - \frac{wrong}{N-n} \\ &= sensibilité + specificité - 1 \end{aligned} \quad (4)$$

où *right* est le nombre d'images qui avaient le mot comme mot de référence et auxquelles le système a associé ce mot, *wrong* est le nombre d'images qui n'étaient pas indexées par ce mot mais pour lesquelles le système a associé ce mot, N est le nombre total d'images dans la base de test et n est le nombre d'images dans la base de développement ayant le mot. Le rapport $right/n$ (aussi appelé rappel ou sensibilité) donne le taux de bonnes indexations, le rapport $wrong/(N-n)$ (équivalent à : un moins spécificité) donne le taux de mauvaises indexations. Le score NS mesure donc la différence entre ces deux rapports. On remarque que $-1 \leq NS \leq 1$. Le score vaut 1 quand on trouve les n mots de référence, et aucun des autres

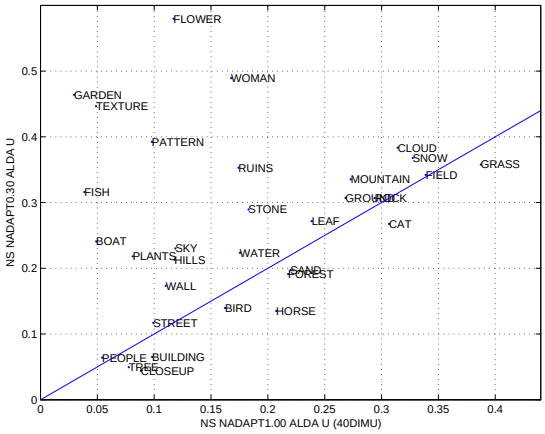

FIG. 4 – Représentation de la consistance visuelle (score NS) des mots pour 40DIMU (sans sélection de traits visuels) en abscisse, et NADAPT0.30 U en ordonnée. NADAPT0.30 U donne de meilleurs résultats que 40DIMU sauf pour les mots situés en dessous de la ligne : *closeup, tree, building, bird, horse, forest, sand, cat, grass*. La méthode ALDA est donc validée.

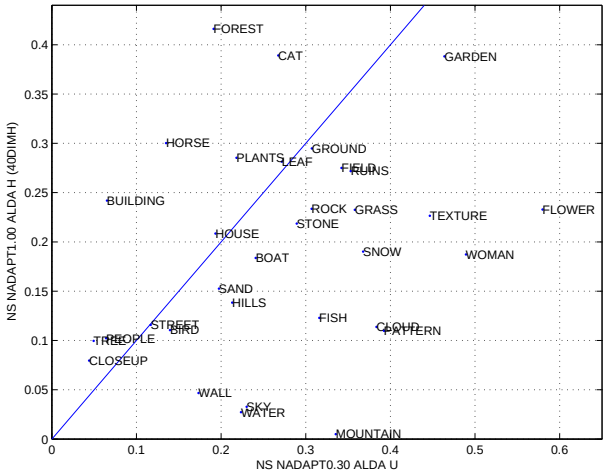

FIG. 5 – Représentation de la consistance visuelle (score NS) des mots pour NADAPT0.30 U (en abscisse) et 40DIMH (en ordonnée). NADAPT0.30 U donne de meilleurs résultats en moyenne que 40DIMH, cependant, certains mots sont mieux discriminés par les traits d'hétérogénéité : *closeup, tree, people, building, house, horse, plants, forest, cat*. Donc l'utilisation des traits d'hétérogénéité permet d'améliorer la reconnaissance de certains concepts.

mots, -1 quand on ne trouve que les mots qui ne sont pas de référence, 0 quand on trouve tous les mots.

6 Expérimentations

Les expérimentations consistent tout d'abord à réaliser une CAH sur un espace contenant tous les traits visuels et à calculer les scores NS pour l'ensemble des mots du lexique, puis à effectuer des CAH sur plusieurs sous-espaces visuels et enfin à comparer les scores NS obtenus avec la première expérience afin de voir si le choix d'utiliser le pouvoir discriminant des traits visuels par rapport aux mots permet d'améliorer les scores NS et donc de faire de meilleures associations entre classes visuelles et mots-clés.

6.1 Corpus

La base d'images utilisée est un sous-ensemble de COREL [27, 19, 28]. Elle est composée de 10000 images. Chaque image possède de 1 à 5 mots-clés choisis manuellement parmi un ensemble de 250 mots environ. En moyenne, il y a 3,6 mots-clés par image. Les images ont été prétraitées par des chercheurs du Computer Vision Group de l'université de California (Berkeley) et du Computing Science Department de l'université d'Arizona [3]. Chaque image a été segmentée en utilisant l'algorithme « normalized cuts » [21] et les 10 plus grands blobs ainsi créés ont été conservés. En moyenne sur notre corpus, il y a 9,5 blobs par image. La figure 1 donne un exemple de segmentation par « normalized cuts ». Les auteurs ont choisi d'extraire des caractéristiques visuelles générales calculables sur tout type de segments. Chaque blob est donc décrit par un vecteur de 40 dimensions, composées de :

- 6 dimensions de formes (aire, périmètre sur aire, convexité, moment d'inertie, position en x et y du barycentre du segment),
- 18 dimensions de couleurs (RGB, RGS, LAB et leurs écarts types),
- 16 dimensions de texture.

Nous avons ensuite normalisé les vecteurs visuels par estimation MLE de distributions Gamma. Finalement, chaque blob est représenté par un vecteur de 40 dimensions dont chaque composante est dans $[0, 1]$. Nous avons choisis de réduire le lexique aux mots-clés ayant plus de 60 occurrences dans la base d'apprentissage, il est donc finalement composé d'un ensemble de 52 mots. Le corpus est ensuite séparé aléatoirement en un ensemble d'apprentissage TRAIN de 5000 images, un ensemble de développement DEV de 2500 images et un ensemble de test TEST de 2500 images.

6.2 Résultats

Nous utilisons l'ensemble TRAIN pour construire les classes visuelles de chaque mot, puis nous utilisons DEV pour apprendre les paramètres (taille des clusters) qui donnent les meilleurs scores NS. La figure 3 montre plusieurs apprentissages sur DEV pour le mot *woman*. Finalement, les scores NS sont calculés sur l'ensemble TEST de 2500 images qui permet de valider les expériences sur les images n'ayant jamais servi à les optimiser.

Nous effectuons tout d'abord une CAH sur les 40 dimensions visuelles classiques (expérience 40DIMU abs-

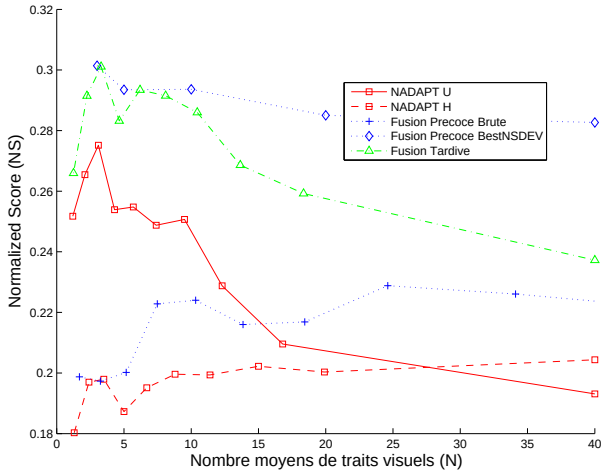

FIG. 6 – Moyennes des scores NS pour les 52 mots les plus fréquents et 2500 images de TEST, en fonction du nombre de traits visuels utilisés. Chaque point d'une courbe est pour τ variant de 10% à 100%.

cisse de la figure 4) comme dans [22][3]. Cette classification nous servira de référence. Les mots qui ont un fort score NS sont : *grass, field, snow, cloud*. Nous remarquons que dans l'ensemble ce sont des mots qui ont une forte consistance visuelle ou bien un fort contexte discriminant. Par exemple, l'herbe est souvent verte, la neige est souvent blanche. Les mots qui ont un faible score NS sont : *garden, fish, texture, boat, people, tree, plants, closeup, building, street, pattern, flower*. Les faibles scores obtenus pour ces mots peuvent peut-être s'expliquer par le bruit ajouté par certaines dimensions. Par exemple, une fleur peut être de différentes couleurs, on ne peut donc pas associer une couleur particulière au mot fleur, les traits couleurs apportent certainement plus de bruit aux classes visuelles du mot fleur que d'information. Pour 40DIMU, la moyenne des scores NS des 52 mots les plus fréquents sur 2500 images de TEST est de 0.192 (première ligne du tableau 1).

Nous souhaitons réduire le nombre de dimensions de l'espace visuel tout en améliorant les scores obtenus pour chacun des mots, mais nous ne savons pas *a priori* quelles dimensions garder. Nous avons essayé tout d'abord des associations de traits visuels naïves (couleurs seules, textures seules...) qui ont donné des scores NS plus faibles que l'expérience 40DIMU.

Nous réalisons ensuite d'autres expériences en choisissant pour chaque mot les traits visuels les plus discriminants d'après l'ALDA. L'expérience NADAPTτ consiste à prendre pour chaque mot les N dimensions les plus discriminantes telles qu'elles cumulent τ% des valeurs de pouvoir discriminant de l'ensemble des traits visuels du mot considéré (comme expliqué dans la partie 3.3). Nous faisons varier τ de 10% à 100%, et nous réalisons une CAH sur les sous espaces à N dimensions construits à partir des espaces U et H. La figure 6 montre la moyenne des scores NS obtenus pour les 52 mots les plus fréquents et 2500 images de TEST. Nous remarquons qu'appliquée sur U la méthode ALDA permet une nette amélioration des scores NS moyens. La figure 5 montre le détail mot par mot : la plupart des mots sont mieux discriminés par NADAPT0.30 U que par 40DIMU. La méthode ALDA est donc expérimentalement validée. Par contre, l'ALDA appliquée sur H ne permet pas une augmentation des scores NS moyens. Nous pouvons supposer que l'ALDA sur H ne fonctionne pas parce que nous n'avons pas assez de données, en effet pour H nous ne disposons que d'un vecteur par image, au lieu d'un vecteur par blob pour U. Les meilleurs résultats sur H sont obtenus pour NADAPT1.00 H (40DIMH), et le score moyen est même supérieur à celui obtenu pour 40DIMU (0.204>0.192). Les scores sur H sont donc en général plus mauvais que sur U, cependant, si nous étudions les scores NS obtenus sur H mot par mot, nous remarquons que certains mots ont un bien meilleur score sur H que sur U. La figure 5 compare les scores NS des mots obtenus pour les meilleurs méthodes de sélection de traits visuels sur H et sur U. Nous remarquons que certains mots (ceux situés au dessus de la droite) : *closeup, tree, people, building, house, horse, plants, forest, cat*, sont mieux discriminés sur H que sur U. L'utilisation de l'hétérogénéité sur les traits visuels est donc une méthode qui permet de mieux discriminer certains concepts que les traits classiques.

6.3 Fusion précoce

L'expérience 80DIM réalise une CAH sur l'ensemble des 80 traits. Le score NS moyen obtenu (0.208) est légèrement supérieur à 40DIMU et à 40DIMH, mais en utilisant 40 dimensions de plus (tableau 1). Nous appliquons ensuite l'ALDA sur les 80 traits. La difficulté principale lors de cette fusion précoce est que les rangs sont bien estimés par l'ALDA, mais les ordres de grandeur des pouvoirs discriminants sur U et sur H sont différents. Pour essayer de surmonter ce problème, nous avons réalisé une estimation MLE de distribution Gamma sur les pouvoirs discriminants des traits classiques et des traits d'hétérogénéité. Nous obtenons la courbe *Fusion Précoce Brute* de la figure 6. Cette courbe est inférieure à la courbe NADAPT U jusqu'à un nombre de traits moyens égale à 15. Puis elle lui est légèrement supérieure. Cette fusion ne fonctionne donc pas efficacement les traits U et H, car les pouvoirs discriminants estimés de U et H même normalisés

[3]Dans [22] la base d'apprentissage est de 7000 images au lieu de 5000 et nous n'avions pas utiliser d'ensemble de TEST. Les scores NS obtenus pour 40DIMU pour 35 mots et 3000 images de DEV étaient de 0.22, alors que nous obtenons un score NS de 0.192 pour 52 mots et 2500 images de TEST.

		DIMENSION		CLASSIFICATION			FILTRAGE		
		N	réduct.	NS	NS	gain	NS	NS	gain
Méthode	τ	moyen	%	moyen	écart type	%	moyen	écart type	%
40DIMU (référence)	1.00	40	-	**0.192**	0.129	-	**0.229**	-	-
NADAPT U	0.30	3.1	-92	0.275	0.151	+43	0.297	0.332	+30
NADAPT U	Bestτ	7.8	-81	**0.293**	0.162	+52	**0.304**	0.354	+33
NADAPT H (40DIMH)	1.00	40	+0	0.204	0.116	+6	0.239	0.320	+4
NADAPT H	Bestτ	9.4	-77	0.211	0.137	+9	0.286	0.330	+25
80DIM	1.00	80	+100	0.208	0.121	+8	0.182	0.323	-21
Fusion Précoce Brute	0.80	24.6	-39	0.229	0.126	+19	0.221	0.329	-3
Fusion Précoce BestNSDEV		3	-93	0.301	0.124	+56	0.258	0.337	+13
Fusion Tardive	0.30	3.3	-92	0.301	0.133	+56	**0.318**	0.339	+39
Fusion Tardive	Bestτ	8.4	-79	**0.325**	0.139	+69	**0.317**	0.349	+38

TAB. 1 – Meilleurs résultats pour chaque type d'expérience comparés à l'expérience de référence 40DIMU réalisée sans sélection des traits visuels, pour les 52 mots les plus fréquents et 2500 images de TEST.

n'ont pas le même ordre de grandeur et les traits de H sont plus souvent choisis par l'ALDA que ceux de U.
Nous avons ensuite construit des sous-espaces de nombres variables de traits en provenance de U ou de H, avec une proportion variable calculée par le rapport des scores NS obtenus avec les vecteurs de même longueur sur U et sur H d'après l'ensemble d'image de développement DEV [23]. Soit $NS_{DEV}U$ ce score pour une longueur de vecteur Z dans U donnée, alors la proportion de traits U dans le vecteur mélange est $\frac{Z_U}{Z_H} = \frac{NS_{DEV}(U)}{NS_{DEV}(H)}$ avec $Z = Z_U + Z_H$. Par exemple, si pour un mot on a $NS = 0.4$ avec la méthode NADAPT U, et $NS = 0.2$ avec NADAPT H, alors le vecteur de fusion précoce de longueur 6 contient les 4 meilleurs traits de U et les 2 meilleurs traits de H. La courbe *Fusion Précoce BestNSDEV* de la figure 6 montre que cette fusion précoce des traits visuels augmente sensiblement les scores NS moyens pour toutes les valeurs de τ. Cette méthode de fusion est donc efficace.

6.4 Fusion tardive

Pour fusionner tardivement U et H, nous choisissons à τ fixé, de réaliser pour chaque mot la CAH soit sur U soit sur H (sans fusion précoce) en fonction des scores obtenus pour chacune des méthodes sur DEV. La courbe *Fusion Tardive* de la figure 6 montre que cette fusion est aussi efficace que la fusion précoce pour un faible nombre de dimensions utilisées, puis les scores moyens diminues, mais restent supérieurs à *Fusion Précoce Brute*. Nous proposons également d'apprendre sur DEV pour chaque mot quelle est la valeur de τ qui donne le meilleur score (*Fusion Tardive Best*τ). Le tableau 1 montre que cette expérience donne les meilleurs résultats de toutes celles vus précédemment avec un score NS moyen de 0.325 (+69%) et une réduction du nombre de dimensions par rapport à 40DIMU de -79%. Si on la compare à NADAPT U Bestτ, qui apprend sur DEV pour chaque mot quelle valeur de τ donne un meilleur NS, on remarque que l'utilisation de l'hétérogénéité dans la fusion tardive, apporte un gain de 0.325-0.293=0.032 points, soit +11%.

6.5 Application : filtrage de l'indexation textuelle d'images

Sur le Web, les images sont associées avec un ensemble de mots-clés parmi lesquels certains sont pertinents et d'autres non : les images sont mal légendées. Nous souhaitons filtrer les mots en fonction de leur pertinence visuelle par rapport à une image. Malheureusement, nous ne disposons pas de bases d'images permettant de valider ce filtrage. De plus, il faudrait une validation de l'efficacité du filtrage par des utilisateurs. Nous proposons donc une autre méthode. De la même manière que dans [22], nous associons tous les mots du lexique à chacune des images de TEST. Puis nous les filtrons à l'aide des classes visuelles construites sur TRAIN et optimisées sur DEV. Nous supposons que les mots de la légende initiale des images de TEST sont les mots pertinents pour l'image, et que tous les autres mots représentent les mots non-pertinents de la page Web. Cette méthode peut être comparer à de l'autoannotation d'images à partir du contenu visuel, mais notre système est construit pour travailler sur des images mal légendées et pour permettre un filtrage grossier des mots qu'il est visuellement impossible d'associer à une image donnée de TEST.
Pour chaque image de TEST, nous calculons le score NS (décrit partie 5.2) en prenant *right* comme étant le nombre de mots de la légende initiale de l'image qui ont été associés à l'image par le système, et *wrong* le nombre de mots qui n'étaient pas dans la légende mais que le système a quand même associé à l'image. Nous faisons ensuite la moyenne des scores NS obtenus pour toutes les images de TEST. La partie FILTRAGE du tableau 1 donne les résultats et la figure 7 montre un

Image 172052 (10 blobs)
Légende (3 sur 3)
water(OK) mountain(OK) coast(OK)
sensi=1.00 specif=0.65
preci=0.15 NS=0.65

20 mots associés par le système
desert(7) water(6) sky(6) wave(6) hills(6) closeup(6) mountain(6) coast(6) tree(6) beach(6) boat(5) branch(5) temple(5) fish(4) sand(4) forest(4) cloud(4) people(4) horizon(3) valley(3)

32 mots non associés
snow(2) statue(2) vegetable(1) rock(1) bird(1) wall(1) flower(1) head(1) building(1) window(1) woman(1) street(1) plants(1) field(1) cat cougar food fungus garden grass ground horse house ice leaf mushroom ocean pattern ruins stone texture

Total: 52 mots

FIG. 7 – Exemple de filtrage d'indexation textuelle d'images. Les 52 mots les plus fréquents sont associés à cette image de TEST, puis filtrés par les classes visuelles de NADAPT0.30 U. Entre parenthèses, le nombre de blobs de l'image indexée par le mot. Les mots *wave, hills, boat, cloud* associés par le système à cette image pourraient annoter cette image.

exemple de filtrage pour une image de TEST.

Les scores NS de filtrage obtenus sont différents de ceux de la partie CLASSIFICATION, car ils dépendent de la fréquence des mots dans l'ensemble de TEST. Si une méthode permet un fort score NS pour les mots très fréquents, alors les scores seront forts, faibles sinon. Par exemple, l'expérience 40DIMH a un score faible de classification (0.211), mais un score fort de filtrage (0.286), peut-être car comme sur H il n'y a qu'un seul vecteur visuel par image, nous n'avons pas assez de données pour bien construire les classes, seuls les mots très fréquents possèdent assez de données pour être bien discriminés. Inversement, pour *Fusion Précoce BestNSDEV*, le score de classification est plus fort que celui de filtrage, cette méthode favorise les mots peu fréquents. Par contre, pour la fusion tardive les scores de classification et de filtrage sont proches montrant que cette méthode est efficace.

Pour filtrer efficacement les mots associés à une image du web, nous utiliserons les classes visuelles des mots ayant un fort score NS, car nous savons que pour ces mots notre système est efficace, pour les mots ayant un faible score NS nous ne pourrons pas faire confiance au contenu visuel de l'image. Pour le filtrage par la méthode *Fusion Tardive Best*τ, nous avons une sensibilité de 0.47 et une spécificité de 0.85, ce qui signifie que le système supprime en moyenne 85% des mots qui ne légendent pas cette image et garde 47% des mots de la légende initiale. Nous pourrions également choisir les classes visuelles pour que le système soit plus sensible afin de garder plus de mots de la légende, mais en filtrant moins de mots qui ne sont pas dans la légende.

7 Discussion et conclusion

Ce papier apporte premièrement une méthode (ALDA) efficace pour sélectionner les traits visuels les plus discriminants en fonction des concepts visuels abordés, et ce en ne travaillant que sur des données réelles bruitées. Nous observons sur la figure 6 que les plus grandes dégradations sont obtenues pour un nombre moyen de dimensions égal à 15, confirmant les résultats théoriques obtenus dans [5].

D'autre part, nous avons étendu la notion d'hétérogénéité à tous les traits visuels, et nous avons mesuré le gain de ce nouveau trait dans des tâches de classification. Nous prouvons alors que quelques mots sont bien distingués par des traits visuels usuels ou par sélection de bons traits d'hétérogénéité ou par les deux. Cette dimension 'contextuelle' de l'analyse de scène a été abordée par certaines études neurobiologiques comme dans [1]. Cette étude montre que le système visuel doit, pour analyser une image, la décomposer et doit intégrer l'information de toutes les régions de l'image et non pas les analyser séparément. Notre article montre que l'hétérogénéité est une information riche pour l'interprétation perceptuelle des régions ambiguës d'image comme le suggère [1] : « cette transformation contexte-dépendante de l'image pour la perception a des implications profondes mais fréquemment sous-appréciées pour des études neurophysiologiques de la vision ». Les systèmes CBIR devraient utiliser de telles approches ; nous les testerons dans notre système d'auto-annotation d'images [8]. Troisièmement, nous proposons des méthodes de fusions efficaces qui permettent un gain de classification de 60% et une réduction du nombre de dimensions de 80%. Finalement, nous montrons comment nous pouvons utiliser les classes construites pour filtrer les mots associés à une image en fonction de leur pertinence par rapport au contenu visuel de l'image.

Nous voulons maintenant utiliser la MMD [26] dans le cas des bases d'images mal indexées pour sélectionner les traits les plus pertinents. Les premiers résultats obtenus montrent que l'hypothèse de gaussianité des données posée par la LDA, et donc l'ALDA, nuit peu aux performances de l'ALDA en comparaison de la méthode AMMD qui ne pose pas cette hypothèse.

Remerciements

Nous remercions K. Barnard [3] et J. Wang [15].

Références

[1] Thomas D. Albright. Why do things look as they do ? : Contextual influences on visual processing. *Journal of Vision*, 2(10), 12 2002.

[2] L. Amsaleg, P. Gros, and S.-A. Berrani. Robust object recognition in images and the related database problems. *Multimedia Tools and applications*, 23(3), 2004.

[3] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. I. Jordan. Matching words

and pictures. In *Journal of Machine Learning Research*, volume 3, pages 1107–1135, 2003.

[4] S.-A. Berrani, L. Amsaleg, and P. Gros. Recherche par similarités dans les bases de données multidimensionnelles : panorama des techniques d'indexation. *Ingénierie des systèmes d'information (RSTI série ISI-NIS)*, 7(5-6) : 65–90, 2002.

[5] K. S. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft. When is "nearest neighbor" meaningful ? In *International Conference on Database Theory (ICDT)*. Springer-Verlag, 1999.

[6] T. A. S. Coelho, P. P. Calado, L. V. Souza, B. Ribeiro-Neto, and R. Muntz. Image retrieval using multiple evidence ranking. In *IEEE Knowledge and Data Engineering*, pages 408–417, 2004.

[7] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, Inc., 2000.

[8] H. Glotin and S. Tollari. Image auto-annotation method using dichotomic visual clustering for CBIR. In *IEEE 4th Inter. Workshop on Content-Based Multimedia Indexing (CBMI2005)*, 2005.

[9] H. Glotin, S. Tollari, and P. Giraudet. Approximation of linear discriminant analysis for word dependent visual features selection. In *IEEE Advanced Concepts for Intelligent Vision Systems (ACIVS2005)*, pages 170–177, september 2005.

[10] Philippe H. Gosselin and Matthieu Cord. A comparison of active classification methods for content-based image retrieval. In *1st International Workshop on Computer Vision Meets Databases (CVDB2004) lié à SIGMOD2004*, 2004.

[11] Hatem Haddad and Philippe Mulhem. Utilisation de la fouille de données images pour l'indexation automatique des images. In *Inforsid'2001*, 2001.

[12] T. Jebara and T. Jaakkola. Feature selection and dualities in maximum entropy discrimination. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2000.

[13] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *ACM SIGIR*, 2003.

[14] G.N. Lance and W.T. Williams. A general theory of classificatory sorting strategies : I. hierarchical systems. *Computer Journal*, 9 :373–380, 1967.

[15] Jia Li and James Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(9) :1075–1088, 2003.

[16] J. Martinet, Y. Chiaramella, and P. Mulhem. A model for weighting image objects in home photographs. In *ACM CIKM'05*, Germany, 2005.

[17] Jean Martinet. *Un modèle vectoriel relationnel de recherche d'information adapté aux images*. Phd thesis, Université Joseph Fourier, Grenoble, 2004.

[18] F. Monay and D. Gatica-Perez. On image auto-annotation with latent space models. In *ACM Multimedia (ACM MM)*, pages 275–278, 2003.

[19] Henning Muller, Stéphane Marchand-Maillet, and Thierry Pun. The truth about corel - evaluation in image retrieval. In *The Challenge of Image and Video Retrieval (CIVR02)*, 2002.

[20] C. Neti, G. Potamianos, J. Luettin, I. Matthews, H. Glotin, and D. Vergyri. Large-vocabulary audio-visual speech recognition : A summary of the Johns Hopkins Summer 2000 Workshop. In *IEEE Work. Multimedia Signal Process.*, 2001.

[21] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Pattern Analysis and Machine Intelligence*, 22(8) : 888–905, 2000.

[22] S. Tollari. Filtrage de l'indexation textuelle d'une image au moyen du contenu visuel pour un moteur de recherche d'images sur le web. In *Actes de CORIA'05*, pages 261–275, mars 2005.

[23] S. Tollari, H. Glotin, and J. Le Maitre. Rehaussement de la classification textuelle d'images par leur contenu visuel. In *Actes du 14ème Congrès Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle*, pages 1383–1392, janvier 2004.

[24] S. Tollari, H. Glotin, and J. Le Maitre. Enhancement of textual images classification using segmented visual contents for image search engine. *Multimedia Tools and Applications*, 25(3) : 405–417, march 2005.

[25] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. In *ACM Multimedia*, 2001.

[26] Nuno Vasconcelos. Feature selection by maximum marginal diversity : optimality and implications for visual recognition. In *IEEE ICIP*, 2003.

[27] J. Wang. http ://wang.ist.psu.edu/docs, 2004.

[28] J. Z. Wang, J. Li, and G. Wiederhold. Simplicity : Semantics-sensitive integrated matching for picture libraries. *IEEE Pattern Analysis and Machine Intelligence*, 23(9) : 947–963, 2001.

[29] X. S. Zhou and T. S. Huang. Unifying keywords and visual contents in image retrieval. *IEEE Multimedia*, 9, 2002.

[30] F. Zuo, P. H. N. de With, and M. van der Veen. Multistage face recognition using adaptative feature selection and classification. In *Proc. of ACIVS2005*, 2005.