



HAL
open science

Méthodes d'Apprentissage pour la Recherche d'Images par le Contenu.

Philippe-Henri Gosselin, Micheline Najjar, Matthieu Cord, Christophe
Ambroise, Sylvie Philipp-Foliguet

► **To cite this version:**

Philippe-Henri Gosselin, Micheline Najjar, Matthieu Cord, Christophe Ambroise, Sylvie Philipp-Foliguet. Méthodes d'Apprentissage pour la Recherche d'Images par le Contenu.. 2004. hal-00520277

HAL Id: hal-00520277

<https://hal.science/hal-00520277>

Preprint submitted on 22 Sep 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.

Méthodes d'Apprentissage pour la Recherche d'Images par le Contenu

P.H. Gosselin*

M. Najjar**

M. Cord*

C. Ambroise**

S. Philipp-Foliguet*

*Equipe de Traitement des Images et du Signal / CNRS UMR 81
6, av du Ponceau, 95014 CERGY-PONTOISE Cedex
{gosselin, cord, philipp}@ensea.fr

**HEUDIASYC (UMR6599 CNRS),
Université de Technologie de Compiègne,
Centre de recherche de Royallieu, 60200 Compiègne
{najjar, ambroise}@utc.fr

Résumé

Les techniques de classification statistiques connaissent actuellement un intérêt croissant dans la communauté de recherche d'image par le contenu (CBIR). Nous comparons dans cet article une méthode par discrimination (SVM) avec une méthode par modélisation (Mélange de Gaussiennes), reconnues comme étant parmi les meilleures techniques de classification. Ces deux approches ont été adaptées au contexte du CBIR. Les avantages de chacune d'entre elles ont été dégagés à l'issue de tests menés sur les 500 images de la base généraliste ANN.

Mots Clef

Recherche d'images par le Contenu, SVM, Mélange de Gaussiennes, Interaction

Introduction

A la différence des premiers systèmes de reconnaissance des formes élaborés dans les années 70-80 où l'on recherchait le "tout automatique", les systèmes plus récents font appel à l'utilisateur pour guider, affiner la recherche. C'est notamment le cas dans le domaine de la recherche d'images par le contenu (CBIR) où l'utilisateur est bien souvent intégré dans le système [8] dans une phase dite de "bouclage de pertinence". Lors de ce bouclage, l'utilisateur donne son avis sur les images affichées à l'écran par le système. Différents moyens d'interaction avec l'utilisateur ont été proposés : déplacement des imagerie sur l'écran afin de rapprocher les images similaires, classement des images de la plus pertinente à la moins pertinente, etc. Le type d'annotation dépend aussi de la nature de la requête (image(s), croquis, descriptif ...) et du type de recherche effectuée (recherche d'une seule image ou d'une catégorie d'images).

Nous nous situons dans cet article dans le cas de la recherche de catégorie à partir d'une requête constituée

d'une image fournie par l'utilisateur. L'annotation peut alors être une simple annotation des images "pertinentes", c'est-à-dire appartenant à la catégorie recherchée, peut comporter des annotations positives et négatives selon que l'image appartient ou n'appartient pas à la catégorie, mais elle peut être plus élaborée et demander à l'utilisateur de positionner un curseur ou de donner une note selon que l'image correspond plus ou moins à sa demande. Ce dernier cas est difficile à gérer, car l'utilisateur n'est souvent pas "stable" dans ces jugements ; il juge de manière relative, par rapport aux autres images présentées et peut évoluer au fil des bouclages. L'interaction que nous avons employée est donc une simple annotation des images présentées par le système, annotations qui peuvent être positives ou négatives. L'utilisateur choisit à chaque itération les images qu'il souhaite annoter, certaines restant non annotées.

L'ensemble des images annotées constitue ce que l'on appelle en reconnaissance des formes un ensemble d'apprentissage et puisque l'on recherche des catégories d'images, nous sommes dans le domaine de la classification supervisée, où les individus à classer sont les images et où le nombre de classes est réduit à deux (la catégorie recherchée et son complémentaire). Ce qui caractérise les systèmes de recherche d'images avec bouclage de pertinence est le fait que les exemples qui vont servir à la fabrication des classes ne sont pas disponibles dès le début du processus. Ces exemples arrivent au fur et à mesure des bouclages, l'ensemble d'apprentissage se construit au fil des itérations. De plus, cet ensemble dépend des itérations précédentes de l'algorithme. En effet, si l'on peut supposer que l'utilisateur annotera toujours de la même façon (pertinente ou non pertinente) une même image, les images présentes dans l'ensemble d'apprentissage dépendent de l'affichage des résultats : ne peuvent être incluses dans cet ensemble que les images présentées à l'utili-

sateur pour annotation. Il s'agit donc d'apprentissage actif.

Les méthodes récentes de classification basées sur les Machines à Vecteurs de Support (SVM) effectuent une classification en deux classes, c'est pourquoi elles semblent bien adaptées à la construction de catégories dans les bases d'images sur la base des annotations des utilisateurs. Le but de cet article est de comparer de manière objective cette méthode émergente à un algorithme plus établi basé sur l'estimation paramétrique de densité dans le cadre de la recherche de catégories d'images.

Un protocole très strict a été mis en place pour comparer les deux méthodes sur la même base d'images et avec les mêmes signatures.

1 Machines à Vecteurs de Support

Les Machines à Vecteurs de Support ont démontré leurs capacités dans le domaine de la reconnaissance de formes, et connaissent aujourd'hui un intérêt croissant dans la communauté de recherche d'images par le contenu [1, 4, 5].

Comme nous l'avons mentionné dans l'introduction, les SVM semblent être une bonne solution pour discriminer les annotations pertinentes des annotations non pertinentes. Cependant, lorsque l'utilisateur n'offre que peu d'annotations, le résultat s'avère souvent inattendu. Cela s'explique par le fait que les SVM travaillent sur la frontière qui sépare les deux classes, et qu'un minimum d'exemples est requis pour permettre une bonne discrimination, sachant qu'en CBIR les images pertinentes n'ont pas forcément la même importance que les images non pertinentes.

Ceci nous amène à une technique très proche appelée "SVM à une classe" qui permet l'estimation de la densité d'un ensemble de vecteurs. Ainsi, il est possible d'avoir une estimation de la catégorie recherchée à l'aide uniquement d'annotations pertinentes. Substituer le cas à une classe au cas à deux classes lors des premières itérations peut pallier en partie le manque d'annotations au démarrage.

1.1 SVM à deux classes

Soient $(\mathbf{x}_i)_{i \in [0, n-1]}$, $\mathbf{x}_i \in \mathbb{R}^p$ les vecteurs (signatures) représentant les images annotées, et $(y_i)_{i \in [0, n-1]}$, $y_i \in \{-1, 1\}$ leurs annotations respectives (1 = pertinent, -1 = non pertinent).

L'objectif est de trouver un hyperplan tel que tous les éléments de même annotation soient du même côté. Autrement dit, il faut trouver un vecteur \mathbf{w} et un réel b tel que :

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) > 0, \forall i \in [0, n-1]$$

On peut donc prendre une fonction de décision :

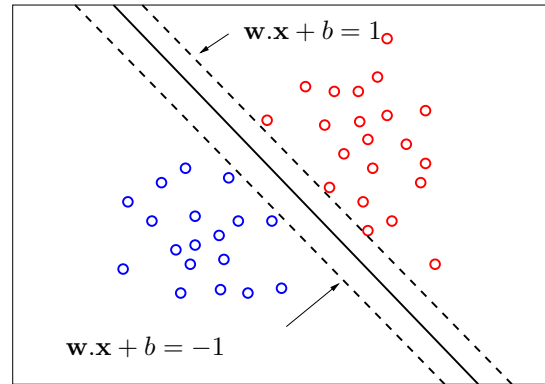


FIG. 1 – Marge des SVM

$$f(\mathbf{x}) = \text{signe}(\mathbf{w} \cdot \mathbf{x} + b)$$

telle que :

$$f(\mathbf{x}_i) = \text{signe}(y_i), \forall i \in [0, n-1]$$

Cette fonction de décision étant invariante par changement d'échelle, on choisit de trouver l'hyperplan tel que $\mathbf{w} \cdot \mathbf{x} + b = \pm 1$ pour les éléments les plus proches de la marge (cf figure 1), ce qui revient à minimiser $\|\mathbf{w}\|^2$ tel que :

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \forall i \in [0, n-1]$$

En utilisant les Lagrangiens, le problème revient à maximiser W sur α :

$$\text{argmax } W(\alpha) = \sum_{i=0}^{n-1} \alpha_i - \frac{1}{2} \sum_{i,j=0}^{n-1} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad (1)$$

avec :

$$\sum_{i=0}^{n-1} \alpha_i y_i = 0 \quad \text{et} \quad \forall i \in [0, n-1] \quad \alpha_i \geq 0$$

La fonction de décision s'écrit alors :

$$f(\mathbf{x}) = \text{signe}\left(\sum_{i=0}^{n-1} y_i \alpha_i \mathbf{x} \cdot \mathbf{x}_i + b\right) \quad (2)$$

Notons que si l'on omet l'opérateur *signe* dans la fonction de décision, cela nous donne une mesure d'appartenance à la catégorie recherchée.

1.2 Noyau et cas non séparable

Cette première méthode suppose que les données sont linéairement séparables. Afin d'assouplir la discrimination une marge souple peut être introduite, en acceptant la mauvaise classification de certains éléments. Ceci revient à majorer chacun des α_i par une constante C [6].

De plus, la séparation linéaire n'est pas adaptée à tous les problèmes, et il est souvent préférable d'introduire un noyau $k(\mathbf{x}, \mathbf{x}')$ qui remplace le produit scalaire $\mathbf{x} \cdot \mathbf{x}'$. L'expression du problème (1) et la fonction de décision (2) restent les mêmes mis à part les remplacements évoqués ci-dessus.

1.3 SVM à une classe

Une méthode très proche de celle à deux classes permet d'estimer le support d'une distribution d'un ensemble $X = (\mathbf{x}_i)_{i \in [0, n-1]}$ de vecteurs représentant une classe d'images [3], avec de préférence un noyau tel que $k(\mathbf{x}, \mathbf{x}) = 1$.

Le problème peut s'exprimer sous la forme suivante :

$$\begin{aligned} \operatorname{argmax} \quad & W(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{i,j=0}^{n-1} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \\ \text{avec} \quad & \sum_{i=0}^{n-1} \alpha_i = 1 \\ \text{et} \quad & \forall i \in [0, n-1] \quad 0 \leq \alpha_i \leq C \end{aligned}$$

La fonction de distribution est :

$$f(\mathbf{x}) = \sum_{i=0}^{n-1} \alpha_i k(\mathbf{x}, \mathbf{x}_i) - b$$

telle que :

$$\begin{aligned} f(\mathbf{x}) &> 0 \quad \text{si } \mathbf{x} \in X \\ f(\mathbf{x}) &< 0 \quad \text{sinon} \end{aligned}$$

1.4 Choix d'un noyau

Noyaux classiques. On retrouve souvent dans la littérature les noyaux suivants :

$$\begin{aligned} \text{Polynomial} \quad & k(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y} + c)^d \\ \text{Gaussien} \quad & k(\mathbf{x}, \mathbf{y}) = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}} \\ \text{Sigmoïde} \quad & k(\mathbf{x}, \mathbf{y}) = \tanh(\mathbf{x} \cdot \mathbf{y} + \theta) \end{aligned}$$

Compte tenu de la complexité des classes dans l'espace des attributs (nombreux modes dispersés), nous avons fait le choix d'utiliser un noyau gaussien.

De plus, ce noyau offre la possibilité de jouer sur la distance d placée en exponentielle :

$$k(\mathbf{x}, \mathbf{y}) = e^{-\frac{d(\mathbf{x}, \mathbf{y})^2}{2\sigma^2}}$$

et le fait que chaque vecteur se trouve sur une sphère (pour le cas à une classe) :

$$k(\mathbf{x}, \mathbf{x}) = 1$$

1.5 Interaction

Le classifieur SVM à deux classes (classe des images pertinentes et classe des images non pertinentes) est utilisé pour déterminer l'appartenance de chaque image à la classe recherchée. L'ensemble utilisé pour entraîner le classifieur est formé par les signatures des images annotées. La distance à la marge est ensuite utilisée pour mesurer le degré d'appartenance de

chaque image à la classe recherchée. Les images sont triées puis affichées selon cette mesure.

Dans le cas où l'utilisateur n'a effectué qu'un seul type d'annotation (uniquement pertinent par ex.), l'estimateur de densité SVM à une classe est utilisée pour mesurer l'appartenance de chaque image à la classe recherchée.

2 EMiner : les modèles de mélange pour la fouille de données

Une approche statistique souple, pour modéliser des classes de formes complexes, est l'approche mélange de gaussiennes [10], qui considère que les observations (index des images) de la classe c sont issus d'une loi :

$$f_c(\mathbf{x}) = P(X = \mathbf{x} | Y = c) = \sum_{r=1}^{R_c} \pi_{cr} G_{cr}(X; \mu_{cr}, \Sigma_{cr})$$

avec $\sum_r \pi_{cr} = 1$ et G_{cr} une loi gaussienne de moyenne μ_{cr} et de covariance Σ_{cr} .

L'une des caractéristiques de la recherche interactive d'images est la rareté des observations (images) annotées. L'approche mélange ne semble pas adaptée car elle nécessite l'estimation de nombreux paramètres. Estimer des nombreux paramètres en utilisant peu de données peut se révéler impossible ou bien mener à des estimations très peu robustes. Pour pallier ce problème nous proposons deux solutions :

- considérer des modèles de variance-covariances simples. Cette solution permet de réduire considérablement le nombre de paramètres à estimer ;
- utiliser l'ensemble des images (annotées et non annotées) de la base pour estimer les paramètres. Ce type d'approche rentre dans le cadre de l'apprentissage semi-supervisé [9].

Dans le cadre d'une approche semi-supervisée les paramètres des densités de classes sont estimés en tirant parti de toutes les images de la base (de nombre N), c.à.d. des images annotées et non annotées. Les signatures des images sont supposés générés par un modèle de mélange à R composantes. Par souci de simplification des formules, nous noterons $g_r(\cdot)$ les densités des composantes gaussiennes, qui ne sont pas les densités de classes. La densité de probabilité du vecteur \mathbf{x}_i s'écrit :

$$g(\mathbf{x}_i | \Phi) = \sum_{r=1}^R \pi_r g_r(\mathbf{x}_i | \theta_r) \quad (3)$$

avec π_r , la proportion de la composante c_r , ($0 < \pi_r < 1$ et $\sum_{r=1}^R \pi_r = 1$) et Φ le vecteur de l'ensemble de paramètres $(\pi_1, \dots, \pi_R, \theta_1, \dots, \theta_R)$.

Pour permettre (1) la recherche de catégories multimodales et (2) la modélisation de la classe C_2 des images

non-pertinentes également multi-modales, nous avons choisi de modéliser les classes C_1 (images pertinentes) et C_2 (images non pertinentes) par un mélange de R_1 et R_2 composantes gaussiennes respectivement. Il est donc évident que $R = R_1 + R_2$. La densité \mathbf{x}_i sera écrite :

$$g(\mathbf{x}_i|\Phi) = \sum_{r=1}^{R_1} \pi_r g_r(\mathbf{x}_i|\theta_r) + \sum_{r=R_1+1}^{R_1+R_2} \pi_r g_r(\mathbf{x}_i|\theta_r) \quad (4)$$

Les annotations de l'utilisateur seront représentées par les vecteurs annotations ou étiquettes $\mathbf{z}_i \in \{0, 1\}^{R_1+R_2}$ qui sont codés comme suit :

- $\mathbf{z}_i = (\underbrace{1, \dots, 1}_{R_1 \text{ fois}}, \underbrace{0, \dots, 0}_{R_2 \text{ fois}})$, pour \mathbf{x}_i *annotée* pertinente c'est à dire appartenant à la classe C_1 ,
- $\mathbf{z}_i = (\underbrace{0, \dots, 0}_{R_1 \text{ fois}}, \underbrace{1, \dots, 1}_{R_2 \text{ fois}})$, pour \mathbf{x}_i *annotée* non-pertinente (appartenant à la classe C_2),
- $\mathbf{z}_i = (\underbrace{1, \dots, 1}_{R_1+R_2 \text{ fois}})$ si \mathbf{x}_i est non annotée (classe inconnue).

La probabilité d'appartenance à la classe C_1 s'exprime comme :

$$P(\mathbf{x}_i \in C_1 | \mathbf{x}_i, \mathbf{z}_i; \Phi) = \sum_{r=1}^{R_1} p(r | \mathbf{x}_r, \mathbf{z}_r; \Phi) \quad (5)$$

et celle de la classe C_2 :

$$P(\mathbf{x}_i \in C_2 | \mathbf{x}_i, \mathbf{z}_i; \Phi) = \sum_{r=R_1+1}^{R_1+R_2} p(r | \mathbf{x}_r, \mathbf{z}_r; \Phi) \quad (6)$$

2.1 Estimation des paramètres du mélange

Dans le contexte de notre application [12], les paramètres du modèle peuvent être estimés en maximisant la log-vraisemblance des paramètres connaissant les index et les annotations. Cette maximisation, non-triviale, peut être réalisée en utilisant l'algorithme EM [11]. Les étapes d'estimation (E) et de maximisation (M) de chaque itération q de l'algorithme sont les suivantes :

- Etape E : Pour chaque composante r et chaque image \mathbf{x}_i , calculer

$$c_{ir}^{(q)} = p(r | \mathbf{x}_i, \mathbf{z}_i; \Phi^{(q)}) = \frac{z_{ir} \pi_r^{(q)} g_r(\mathbf{x}_i | \theta_r^{(q)})}{\sum_{l=1}^R z_{il} \pi_l^{(q)} g_l(\mathbf{x}_i | \theta_l^{(q)})}$$

- Etape M : Calcul des nouveaux paramètres $\Phi^{(q+1)}$ qui maximise

$$Q(\Phi | \Phi^{(q)}) = \sum_{i=0}^{N-1} \sum_{r=1}^R c_{ir}^{(q)} \log \pi_r g_r(\mathbf{x}_i | \theta_r).$$

3 Expérimentations

3.1 Protocole d'évaluation

Les tests sont effectués sur la base ANN composée d'environ 500 images partitionnée en 11 catégories de 25 à 50 images. Deux attributs sont calculés (Couleurs Lab et Filtres de Gabor) pour former les distributions représentant chaque image [7].

Les performances des systèmes d'apprentissage sont mesurées à l'aide de courbes de précision/rappel sur une catégorie recherchée. Si on note A l'ensemble des images appartenant à la catégorie, et B l'ensemble des images affichées par le système, on a :

$$\begin{aligned} \text{Précision} &= \frac{|A \cap B|}{|B|} \\ \text{Rappel} &= \frac{|A \cap B|}{|A|} \end{aligned}$$

Le nombre d'images affichées par ordre d'appartenance à la catégorie cherchée varie de 1 à N . A chaque bouclage de pertinence, nous simulons l'annotation de $m = 5$ images. Nous calculons un point de précision/rappel moyen de la catégorie pour chaque nombre d'images affichées par le système. Nous obtenons ainsi une courbe de précision en fonction du rappel pour une catégorie et un nombre d'itération donnés. Le Tableau 1 décrit plus en détail le protocole d'évaluation.

3.2 Résultats

Paramétrage SVM. Le noyau utilisé est un noyau gaussien de distance L^2 sur des données normalisées ($\sigma = 1$). Nous choisissons la distance L^2 puisque elle est très proche de celle utilisée par EMiner. La normalisation des distributions des images de la base ANN est très importante. Nous constatons que tout autre choix de pondération des distributions induit une instabilité dans les résultats.

Paramétrage EMiner. Lorsque l'utilisateur a choisi son image requête, les paramètres des lois attachées aux classes C_1 et C_2 sont estimés en utilisant l'image requête comme seule image annotée pertinente. Toutes les autres images de la base restent non annotées mais sont utilisées dans l'estimation. Ensuite, les probabilités conditionnelles d'appartenance des images à la classe des images pertinentes sont calculées et les meilleures images résultats sont affichées à l'écran par ordre décroissant d'intérêt au sens du rapport des probabilités conditionnelles :

$$f(\mathbf{x}_i) = \frac{P(\mathbf{x}_i \in C_1 | \mathbf{x}_i, \mathbf{z}_i; \Phi)}{P(\mathbf{x}_i \in C_2 | \mathbf{x}_i, \mathbf{z}_i; \Phi)}$$

L'utilisateur annoté les images qui lui semblent pertinentes et les images restantes (parmi les m) sont classées automatiquement comme non pertinentes. Chaque bouclage de pertinence affine la requête en ré-estimant les paramètres du modèle avec des nouvelles annotations.

Par souci de concevoir un algorithme de recherche interactive en ligne simple et rapide, le nombre de composantes du modèle de mélange a été fixé a priori de telle sorte que le nombre de composantes de la classe des images pertinentes soit égal au nombre d'images annotées positives. Le nombre de composantes de la classe des images non-pertinentes est fixé à un nombre relativement élevé (20 par exemple).

Courbes. Les différentes courbes 2, 3 et 4 représentent la précision/rappel d'une simulation sur la catégorie donnée pour 1, 2, 3, 4 et 5 bouclages à raison de $m = 5$ annotations par bouclage.

Au vu des résultats, chacune des deux méthodes semble avoir son domaine de prédilection.

Par exemple, sur la classe 'geneva' (figure 5), EMiner est plus performant que SVM (figure 2). En revanche, sur la classe 'cherries' (figure 6), c'est l'effet contraire qui est constaté (figure 3). Notons que la classe 'geneva' est composée de 25 images plus dispersées dans l'espace des attributs que la classe des 50 images 'cherries'. De plus, une autre classe 'springflower' (figure 7) est très proche de 'cherries'. Ceci permet peut-être d'expliquer les résultats obtenus pour chaque méthode.

En effet, le SVM travaille sur la frontière qui sépare la classe des images pertinentes de la classe des images non pertinentes, alors que le EMiner se concentre davantage sur les centres des deux classes. Le classifieur SVM découpe l'espace en deux morceaux, et si le nombre d'annotations est faible, cette séparation peut être inattendue (pour l'utilisateur). Par exemple, sur la catégorie 'geneva', rien n'empêche le fait que lors des premières annotations, certains modes de la classe recherchée soient classés comme non pertinents. L'EMiner, au contraire, puisqu'il se concentre sur les centres des classes, n'a pas ce genre de problème. Il recherche des images proches du centre et non des images proches d'une frontière mal établie par le SVM lors des premières itérations.

Pour la classe 'cherries', SVM est plus performant que EMiner. Comme nous l'avons fait remarquer précédemment, la classe 'cherries' est plutôt unimodale et mélangée dans l'espace des attributs avec la classe 'springflower'. Dans ce cas la précision apportée par la frontière des SVM peut expliquer ce résultat, puisqu'il sépare les deux classes sans difficulté. De plus, le caractère plutôt unimodal de la classe ne contribue pas à la détermination d'une frontière inattendue. Le EMiner, par contre, semble moins précis pour discriminer avec précision, et sa capacité à mieux modéliser des classes multi-modales lors des premières annotations ne peut être exploitée compte tenu de la forme de la classe.

Enfin, pour la classe 'green lake' (figure 6), nous sommes face à un cas situé entre les deux précédents, à la fois dispersé dans l'espace des attributs et proche d'une autre classe. Dans ce cas, il semble délicat d'affirmer la supériorité d'un algorithme par rapport à l'autre (figure 4).

Soient :

- N la taille de la base (le nombre total des images) ;
- b_{max} le nombre total de bouclage de pertinence effectué ;
- m le nombre des images annotées à chaque boucle de pertinence ;
- lot l'ensemble des images parmi lesquelles sont choisies les meilleures, $N - b \times m$ est le nombre des images du lot.

Ainsi pour m (ici égal à 5) et pour une catégorie donnée :

Pour toutes les images de la catégorie, **faire** :

o Initialisation :

- image requête \leftarrow image courante ;
- classifier (avec SVM ou EMiner) les images de la base.

o Pour $b = 1$ à b_{max} , **faire** :

1. sélection des m meilleures images parmi le lot ;
2. annotation de m images non précédemment annotées ;
3. les images annotées sont retirées du lot pour le prochain tirage ;
4. classification (SVM ou EMiner) des images de la base ;

La précision et le rappel calculés sont moyennés sur toutes les images requêtes de la catégorie.

Conclusion et Perspectives

L'intérêt croissant de la communauté CBIR pour la classification par SVM semble justifié. Cependant, face à un classifieur paramétrique comme EMiner, l'utilisation directe de la méthode n'est pas suffisante.

Au vu des résultats obtenus, les deux approches semblent plus se compléter que se concurrencer. D'une part, EMiner modélise mieux des catégories multimodales complexes. D'autre part, SVM a une plus grande capacité pour discriminer de manière précise une classe donnée.

Un mélange des deux stratégies est une perspective à envisager. Par exemple, l'utilisation d'un EMiner lors des premières itérations afin de collecter un nombre suffisant de modes, suivi d'un SVM pour parfaire la discrimination, nous semble être une approche prometteuse.

Références

- [1] S. Tong, *Active Learning : Theory and Application*, Thèse, Université de Stanford, 2001.

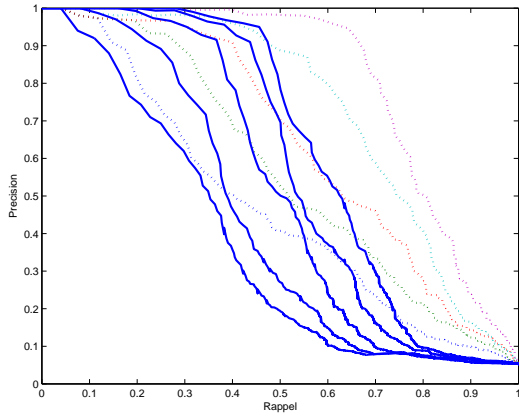


FIG. 2 – Comparaison des courbes précision/rappel pour 1, 2, 3, 4, 5 bouclages de pertinence sur les images de la catégorie geneva. Les courbes obtenues par SVM sont en trait plein et les courbes obtenus par EMiner en pointillés.

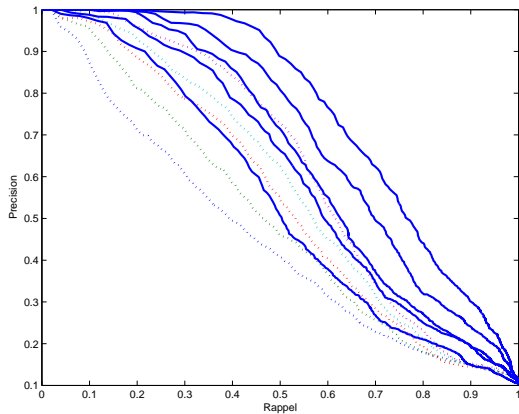


FIG. 3 – Comparaison des courbes précision/rappel pour 1, 2, 3, 4, 5 bouclages de pertinence sur les images de la catégorie cherries. Les courbes obtenues par SVM sont en trait plein et les courbes obtenus par EMiner en pointillés.

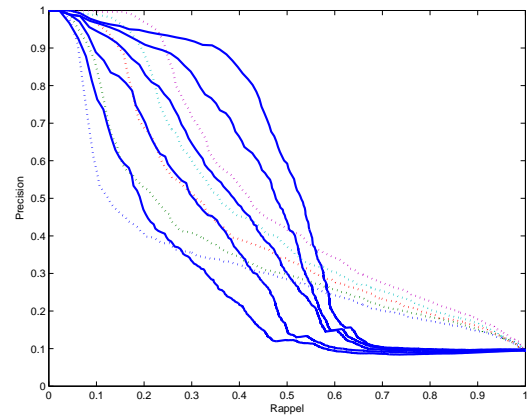


FIG. 4 – Comparaison des courbes précision/rappel pour 1, 2, 3, 4, 5 bouclages de pertinence sur les images de la catégorie greenlake. Les courbes obtenues par SVM sont en trait plein et les courbes obtenus par EMiner en pointillés.

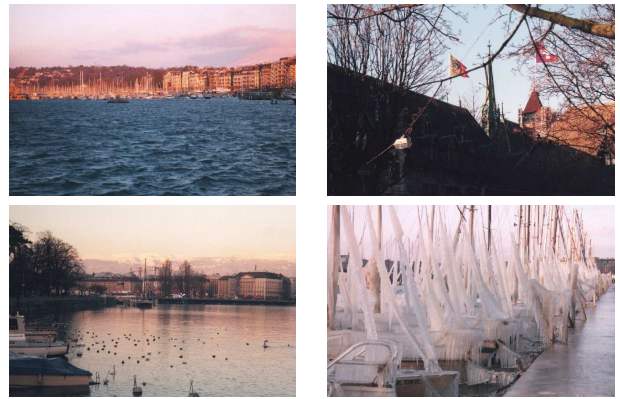


FIG. 5 – Images de la catégorie 'geneva'



FIG. 6 – Images de la catégorie 'cherries'



FIG. 7 – Images de la catégorie 'springflowers'

- [2] A. Smola, *Learning with Kernels*, Thèse, Université de Berlin, 1998.
- [3] B. Scholkopf, *Estimating the Support of High-Dimensional Distribution*, Rapport, Microsoft Research, 1999.
- [4] L. Wang, *Image Retrieval with SVM Active Learning Embedding Euclidian Search*, Article, ICIP 2003, 2003.
- [5] B. Le Saux, *Classification non exclusive et personnalisation par apprentissage : Application la navigation dans les bases d'images*, Thèse, 2003.
- [6] K. Veropoulos, *Controlling the Sensivity of Support Vector Machines*, Article, Proceedings of the International Joint Conference on Artificial Intelligence, (IJCAI99), Stockholm, Sweden, 1999.
- [7] J. Fournier, *Indexation d'images par le contenu et recherche interactive dans les bases généralistes*, Thèse, Université de Cergy-Pointoise, 2002.
- [8] R.C. Veltkamp, *Content-Based Image Retrieval System : A Survey*, Rapport Technique, 2000.
- [9] K. Nigam, A. McCallum, S. Thrun, T. Mitchell, *Text Classification from Labeled and Unlabeled Documents using EM*, Machine learning, Vol.39, n2/3, pp. 135-167, 2000.
- [10] T. Hastie, R. Tibshirani, *Discriminant Analysis by Gaussian Mixtures*, Journal of the Royal Statistical Society B, Vol. 58, pp. 155-176, 1996.
- [11] A. Dempster, N. Laird, D. Rubin, *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society, Vol 39, pp .1-38, 1977.
- [12] M. Najjar, C. Ambroise, J-P. Cocquerez, *Image Retrieval using Mixture Models and EM Algorithm*, 13'th Scandinavian Conference on Image Analysis, Gteborg, Sweden, 29 June - 2 July 2003.



FIG. 8 – Images de la catégorie 'greenlake'