Discriminative Classification vs Modeling Methods in CBIR

Philippe-Henri Gosselin, Micheline Najjar, Matthieu Cord, Christophe Ambroise

▶ **To cite this version:**

Philippe-Henri Gosselin, Micheline Najjar, Matthieu Cord, Christophe Ambroise. Discriminative Classification vs Modeling Methods in CBIR. IEEE International Conference on Advanced Concepts for Intelligent Vision Systems, Sep 2004, Belgium. pp.1. hal-00520316

HAL Id: hal-00520316
https://hal.science/hal-00520316v1

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.

DISCRIMINATIVE CLASSIFICATION *VS* MODELING METHODS IN CBIR

[1]*Philippe H. Gosselin,* [2]*Micheline Najjar,* [1]*Matthieu Cord and* [2]*Christophe Ambroise*

gosselin@ensea.fr
[1]ETIS / CNRS UMR 8051, 6 avenue du Ponceau, 95014 Cergy
[2]HEUDIASYC / CNRS UMR 6599, Centre de recherche de Royallieu, 60200 Compiègne

ABSTRACT

Statistical learning methods are currently considered with an increasing interest in the content-based image retrieval (CBIR) community. We compare in this article two leader techniques for classification tasks. The first method uses one-class and two-class SVM to discriminate data. The second approach is based on Gaussian Mixture to model classes. To deal with the specificity of the CBIR classification task, adaptations have been proposed. Experimental tests on a generalist database have been carried out. Advantages and drawbacks are discussed for each method.

1. INTRODUCTION

Content-Based Image Retrieval (CBIR) systems have attracted large amounts of research attention since 1990's.
Contrary to the early systems, focused on "full-automatic" strategies, recent approaches introduce human-computer interaction into CBIR [1]. Starting with a coarse query, the interactive process allows the user to refine his request as long as it is necessary. In this paper, we focus on large image category retrieval, starting with one relevant image.
Many kinds of interaction between the user and the system have been proposed, but most of the time, user provides binary annotations indicating whether or not the image belongs to the desired category.
Interactive methods may be split into two classes, the geometrical and the statistical approaches [2]. The first one aims at updating the query or optimizing the similarity function thanks to the user annotations. Recently, statistical learning approaches have been introduced in CBIR context and have been very successful. As well the techniques modeling the searched category as a density probability function, as the discrimination methods significantly improve the effectiveness of the visual information retrieval task.
However, the CBIR context is a very specific classification task :

- There are very few training data during the retrieval process,
- Unlabeled data are always available during the learning,
- The dimension of the input space is very high,
- Because of interaction, the learning process is an active learning task [3].
- The two training classes do not have the same number of data.

Two methods are presented and compared in this article: a discriminative approach (using Support Vector Machines SVM) against a model-based one (using Gaussian Mixture EMiner). Both are efficient techniques to handle classification tasks. The goal is to see how these techniques may be efficient in this specific CBIR context. We also present an active learning strategy, which can be combined with any classification method to improve retrieval efficiency. A strict protocol is used to evaluate performances and to compare both methods using the same feature vectors on a generalist database.

2. SUPPORT VECTOR MACHINES

Support Vector Machines have shown their capacities in pattern recognition, and today know an increasing interest in CBIR [3, 4], and seems to be a good solution for discriminating between relevant and irrelevant annotations.
This classification method can deal with high dimensionality using the "kernel trick", and does not require a large training set. However, during the first feedback steps, the user gives so few annotations (from 1 to 10) that classifier can not give good results.
This leads us to a similar method "One-Class SVM" which allows density estimation of a vector set. Thus, one can get an estimation of the searched category using only relevant annotations. Substitute One-Class to Two-Class in the first iterations can overcome the lack of annotations during the beginning.

2.1. Two-Class SVM

Let $(\mathbf{x}_i)_{i\in[1,n]}$, $\mathbf{x}_i \in \mathbb{R}^p$ be the feature vectors representing labeled images, and $(y_i)_{i\in[0,n-1]}$, $y_i \in \{-1, 1\}$ be their respective annotations (1 = relevant, -1 = irrelevant). The aim of the SVM classification method is to find the best hyperplane separating relevant and irrelevant vectors maximizing the size of the margin. Initial method assumes that relevant and irrelevant vectors are linearly separable. To overcome this problem, kernel $k(.,.)$ have been introduced. It allows to deal with non-linear spaces. Moreover, a soft margin may be used too in order to get better efficiency with noisy configuration. It consists in a very simple adaption by introducing a bound C in the initial equations [5]. The resulting optimization problem may be expressed as the following:

$$\boldsymbol{\alpha}^\star = \underset{\boldsymbol{\alpha}}{\text{argmax}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$
$$\text{with} \begin{cases} \sum_{i=1}^{n} \alpha_i y_i = 0 \\ \forall i \in [1, n] \quad 0 \leq \alpha_i \leq C \end{cases} \tag{1}$$

Thanks to the optimal $\boldsymbol{\alpha}^\star$ value, the distance between a vector x and the separating hyperplane is used to evaluate its relevance to the searched category:

$$f(\mathbf{x}) = \sum_{i=1}^{n} y_i \alpha_i^\star k(\mathbf{x}, \mathbf{x}_i) + b \tag{2}$$

2.2. One-Class SVM

A One-Class SVM method estimates the density support of a vector set $(\mathbf{x}_i)_{i\in[0,n-1]}$ representing an image class [6]. With a kernel $k(\mathbf{x}, \mathbf{x}) = 1$, this lead to the following optimization problem:

$$\boldsymbol{\alpha}^\star = \underset{\boldsymbol{\alpha}}{\text{argmax}} \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j)$$
$$\text{with} \begin{cases} \sum_{i=1}^{n} \alpha_i = 1 \\ \forall i \in [1, n] \quad 0 \leq \alpha_i \leq C \end{cases} \tag{3}$$

The function $f(.)$ (Eq. 2) can also be used in the One-Class context.

2.3. Kernel

In our experiments, we use a gaussian radial basis function kernel, which has always given us the best results:

$$k(\mathbf{x}, \mathbf{y}) = e^{-\frac{1}{2}\left(\frac{d(\mathbf{x},\mathbf{y})}{\sigma}\right)^2} \tag{4}$$

The distance $d(.,.)$ depends on feature vectors, and will be discuted in section 5.

3. EMINER : MIXTURE MODELS FOR DATA RETRIEVAL

The Gaussian mixture model approach is a flexible statistical approach to model complex and in-homogeneous categories [7]. The authors consider that observations of class c have the following density:

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

where the mixing proportions π_{cr} sum to one, G_{cr} is a Gaussian distribution, and μ_{cr} and Σ_{cr} are the class center and the covariance matrix respectively.

3.1. Semi-supervised mixture model

Interactive image retrieval is characterized by a small number of labeled observations (images). However, the mixture model approach does not seem adapted because it requires many parameters estimation. Using few observations to estimate many parameters can be impossible to carry out or leads at least to non robust estimation. In order to cure this problem, we propose two solutions:

- Consider diagonal covariance matrix. This assumption allows to reduce the number of parameters to be estimated (but keeps the possibility of a relatively complex class) and increases the estimation robustness.
- Use all images of the database (labeled and unlabeled) as training set. This approach lies within the semi-supervised learning framework [8].

Image indexes are supposed to be generated from a Gaussian mixture with R components. To simplify the problem formulation, we note $g_r()$ the Gaussian density of the mixture component r. Thus the probability density of vector $\mathbf{x}_i$ is:

$$g(\mathbf{x}_i|\boldsymbol{\Phi}) = \sum_{r=1}^{R} \pi_r g_r(\mathbf{x}_i|\boldsymbol{\theta}_r) \tag{5}$$

where π_r is the proportion of component c_r, ($0 < \pi_r < 1$ and $\sum_{r=1}^{R} \pi_r = 1$). $\boldsymbol{\Phi}$ is the parameters vector to be estimated $(\pi_1, ..., \pi_R, \boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_R)$.
We aim (1) to retrieve multi-modal categories of relevant images and (2) to model the heterogeneous class C_2, of irrelevant images. Thus, we choose to model the relevant images class (noted C_1) and the irrelevant images class (noted C_2) by a mixture of R_1 and R_2 Gaussian components respectively. The density of $\mathbf{x}_i$ is:

$$g(\mathbf{x}_i|\boldsymbol{\Phi}) = \sum_{r=1}^{R_1} \pi_r g_r(\mathbf{x}_i|\boldsymbol{\theta}_r) + \sum_{r=R_1+1}^{R_1+R_2} \pi_r g_r(\mathbf{x}_i|\boldsymbol{\theta}_r) \tag{6}$$

The annotation vectors or labels $\mathbf{z}_i \in \{0,1\}^{R_1+R_2}$ are coded as follows:

- $\mathbf{z}_i = (\underbrace{1,\ldots,1}_{R_1 \text{ times}},\underbrace{0,\ldots,0}_{R_2 \text{ times}})$, for $\mathbf{x}_i$ *labeled* relevant,
- $\mathbf{z}_i = (\underbrace{0,\ldots,0}_{R_1 \text{ times}},\underbrace{1,\ldots,1}_{R_2 \text{ times}})$, for $\mathbf{x}_i$ *labeled* irrelevant,
- $\mathbf{z}_i = (\underbrace{1,\ldots,1}_{R_1+R_2 \text{ times}})$ if $\mathbf{x}_i$ is *unlabeled*.

The posterior probability that $\mathbf{x}_i$ belongs to class C_1 is given by:

$$P(\mathbf{x}_i \in C_1|\mathbf{x}_i,\mathbf{z}_i;\mathbf{\Phi}) \quad = \quad \sum_{r=1}^{R_1} p(r|\mathbf{x}_i,\mathbf{z}_i;\mathbf{\Phi}) \quad (7)$$

with

$$p(r|\mathbf{x}_i,\mathbf{z}_i;\mathbf{\Phi}) = \frac{z_{ir}\pi_r g_r(\mathbf{x}_i|\theta_r)}{\sum_{s=1}^{R} z_{is}\pi_s g_s(\mathbf{x}_i|\theta_s)}$$

The images are returned to the user by descending order of:

$$f(\mathbf{x}_i) = \frac{P(\mathbf{x}_i \in C_1|\mathbf{x}_i,\mathbf{z}_i;\mathbf{\Phi})}{P(\mathbf{x}_i \in C_2|\mathbf{x}_i,\mathbf{z}_i;\mathbf{\Phi})} \quad (8)$$

3.2. Mixture parameter estimation

In the context of our application [9], the mixture parameters can be estimated by maximizing the likelihood knowing the indexes and their labels. The classical and natural method for computing the maximum-likelihood estimates for mixture distributions is the EM algorithm [10], which is known to converge to a local maximum. EM alternates between the two Expectation and Maximization steps, at each iteration q:

- E-step: for each component r and each image $\mathbf{x}_i$, compute

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

- M-step: Compute the parameters $\mathbf{\Phi}^{(q+1)}$, which maximize

$$Q(\mathbf{\Phi}|\mathbf{\Phi}^{(q)}) = \sum_{i=1}^{N}\sum_{r=1}^{R} c_{ir}^{(q)} \log \pi_r g_r(\mathbf{x}_i|\boldsymbol{\theta}_r)$$

4. RETIN ACTIVE LEARNING STRATEGY

Performances of inductive classification depend on the training data set. In interactive CBIR, all the images labeled during the retrieval session are added to the training set used for classification. As a result, the choice of these labeled images will change system performances. For instance, labeling images very close to ones already labeled will not change the current classification.

Notations. Let $(\mathbf{x}_i)_{i\in[1,N]}$, $\mathbf{x}_i \in \mathbb{R}^p$ be the feature vectors representing images from the whole database, and $\mathbf{x}_{(i)}$ the permuted vectors after a sort according to the function f (Eq. 2).

Starting from the SVM$_{active}$ method [3], we present an active learning strategy to deal with these aspects: RETIN AL (Active Learning). At the feedback iteration j, we propose to label $m = 2p+1$ images using a rank s_j:

$$\underbrace{\mathbf{x}_{(1),j}}_{\text{most relevant}}, \mathbf{x}_{(2),j}, ..., \underbrace{\mathbf{x}_{(s_j-p),j}, ..., \mathbf{x}_{(s_{j+p}),j}}_{m \text{ images to label}}, ..., \underbrace{\mathbf{x}_{(N),j}}_{\text{less relevant}}$$

The problem is to handle s in order to get a balanced training data. The user gives new annotations for images $\mathbf{x}_{(s_j-p),j}, ...\mathbf{x}_{(s_{j+p}),j}$. Let us note $r_{rel}(j)$ and $r_{irr}(j)$ the numbers of relevant and irrelevant annotations. To obtain balanced training sets, s has to be increased if $r_{rel}(j) > r_{irr}(j)$, and decreased otherwise. We adopt the following upgrade rule for s_{j+1}: $s_{j+1} = s_j + k \times (r_{rel}(j) - r_{irr}(j))$ For now, we have used this relation with $k = 2$ in all our experiments.

Once s_{j+1} is computed, the system should propose to the user the m images from $\mathbf{x}_{(s_{j+1}-p),j+1}$ to $\mathbf{x}_{(s_{j+1}+p),j+1}$. Actually, we also want to increase the sparseness of the training data. Indeed, nothing prevents an image close to another (already labeled or selected) to be selected. To overcome this problem, we consider the same strategy but working no more on images but on clusters of images: we compute m clusters of images from $\mathbf{x}_{(s_j-p),j}$ to $\mathbf{x}_{(s_j-p+M-1),j}$ (where $M = 10 \times m$ for instance), using an enhanced version of the LBG algorithm [11]. Next, the system selects for labeling the most relevant image in each cluster. Thus, images close to each other in the feature space will not be selected together for labeling.

5. EXPERIMENTS

5.1. Features

Color and texture information are exploited. $L^*a^*b^*$ space is used for color, and Gabor filters, in twelve different scales and orientations, are used for texture analysis. Both spaces are clustered using an enhanced version of LBG algorithm [11]. We take the same class number for both spaces. Tests have shown that $c = 25$ classes is a

category	size	description
birds	219	birds from all around the world
castles	191	modern and middle ages castles
caverns	121	inside caverns
dogs	111	dogs of any species
doors	199	doors of Paris and San Francisco
Europe	627	European cities and countryside
flowers	506	flowers from all around the world
food	315	dishes and fruits
mountains	265	mountains
objects	116	single objects
savana	408	animals in African savana

Table 1: COREL categories for evaluation

category	SVM/RETIN AL	EMiner/RETIN AL
birds	**13**	11
castles	**24**	23
caverns	**61**	59
dogs	**38**	34
doors	82	**86**
Europe	**28**	25
flowers	**57**	27
food	**51**	22
mountains	**34**	32
objects	**53**	29
savana	**51**	47

Table 2: COREL evaluation: system performances with L_2 distance estimated with the P_a metric (sum of Precision/Recall function), at the end of the interactive learning process.

good choice for all our feature spaces [12]. The image signature is composed of one vector representing the image color and texture distributions. The input size p is then 50 in our experiments.

5.2. Database and evaluation protocol

Tests are carried out on the generalist COREL photo database, which contains more than $50,000$ pictures organized in categories. To get tractable computation for the statistical evaluation, we randomly selected 77 of the COREL folders, to obtain a database of $6,000$ images. We built 11 categories[1] (*cf.* Table 1) from this database to get sets with different sizes and complexities.
The CBIR system performances are measured using the average precision P_a, which represents the value of the Precision/Recall integral function on a required category. This metric is used in the TREC VIDEO conference[2], and gives a global evaluation of the system. Let us note A the set of images belonging to the category, and B the set of best similar images returned by the system to the user, then: Precision $= \frac{|A \cap B|}{|B|}$ and Recall $= \frac{|A \cap B|}{|A|}$. $|B|$, the cardinal of B, varies from 1 to N.

5.3. Experiments

Each simulation is initialized with one relevant image, and at each one of the 10 feedback steps, 20 images are labeled using the active learning strategy. The training set contains 201 images at the end of the interactive learning process. The classification performances are then provided for systems trained with only 3% of the whole database.
We first experiment the following systems:

- SVM classifier with gaussian L_2 kernel;
- EMiner classifier assuming a gaussian mixture.

These systems use a RETIN AL strategy. Results are shown by Table 2.
Considering only SVM and EMiner with a L_2 metric, SVM has higher performances overall, except for the *doors* category. *Doors* category is well represented by horizontal and vertical textures. As features used for these experiments are color and Gabor filters, images from this category are very concentrated in feature space. In such a case, one can suppose that the gaussian mixture model is well suited. Focusing on other categories, images are really sparse in feature space. There are no large clusters, and the high accuracy of discriminative classifiers as SVM makes the difference.
Next we experiment the following systems, using a χ^2 metric:

- SVM classifier with gaussian χ^2 kernel, and RETIN AL strategy;
- SVM classifier with gaussian χ^2 kernel, and SVM$_{active}$ strategy.

For EMiner, the EM algorithm needs adaptations that we do not present here. Results are shown by Table 3.
The use of a χ^2 distance improves results for all categories. This is not surprising, because input vectors are distributions. This shows that the choice of the metric is significant for system performances. Because SVM classifiers are methods which can be easily "kernelized", changing this metric is easy. For EMiner, this is not the case: the EM algorithm must be adapted. For instance, one can implement an EM assuming a Laplace mixture.

[1] A description of this database and the 11 categories can be found at: http://www-etis.ensea.fr/∼cord/data/mcorel.tar.gz. This archive contains lists of image file names for all the categories.

[2] http://www-nlpir.nist.gov/projects/trecvid/

category	SVM/RETIN AL	SVM/SVM$_{active}$
birds	**31**	**31**
castles	**38**	**38**
caverns	**78**	75
dogs	**58**	**58**
doors	**93**	83
Europe	**35**	**35**
flowers	**67**	57
food	**71**	59
mountains	**54**	**54**
objects	**78**	76
savana	**68**	56

Table 3: COREL evaluation: system performances with χ^2 distance estimated with the P_a metric (sum of Precision/Recall function), at the end of the interactive learning process.

A last point concerns the semi-supervised aspect of EMiner. Semi-supervised methods use unlabeled data to improve classification, but in the CBIR context, it seems that the lack of structure does not allow any significant improvements. The same results can be expected from the semi-supervised versions of SVM, such as Transductive SVM [13]. As these techniques use the whole database, the time required for their computation is huge in comparison to other inductive methods. For instance, computations for EMiner in these experiments are about fifty times more expensive.

6. CONCLUSION

Increasing interest of the CBIR community for SVM classifiers and kernel-based methods seems to be justified. Because most of the categories searched by users are not necessarily structured, discriminative classifiers as SVM are better adapted to the CBIR context. Futhermore, the use of a kernel in an algorithm provides an easy tuning to a specific database feature vectors. Because the adaptation of EM is complex, kernel-based methods should be preferred in this context.

7. REFERENCES

[1] R.C. Veltkamp, "Content-based image retrieval system: A survey," Tech. Rep., University of Utrecht, 2002.

[2] N. Vasconcelos and M. Kunt, "Content-based retrieval from image databases: current solutions and future directions," in *International Conference in Image Processing (ICIP'01)*, Thessaloniki, Greece, October 2001, vol. 3, pp. 6–9.

[3] Simon Tong, *Active Learning: Theory and Applications*, Ph.D. thesis, Stanford University, 2001.

[4] L. Wang, "Image retrieval with svm active learning embedding euclidian search," in *IEEE International Conference on Image Processing*, Barcelona, September 2003.

[5] K. Veropoulos, "Controlling the sensivity of support vector machines," in *International Joint Conference on Artificial Intelligence (IJCAI99)*, Stockholm, Sweden, 1999.

[6] B. Scholkopf, "Estimating the support of high-dimensional distribution," Tech. Rep., Microsoft Reseach, 1999.

[7] T. Hastie and R. Tibshirani, "Discriminant analysis by gaussian mixtures," *Journal of the Royal Statistical Society B*, vol. 58, pp. 155–176, 1996.

[8] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell, "Classification from labeled and unlabeled documents using em," *Machine learning*, vol. 39, no. 2/3, pp. 135–167, 2000.

[9] N. Najjar, J.P. Cocquerez, and C. Ambroise, "Feature selection for semi supervised learning applied to image retrieval," in *IEEE ICIP*, Barcelena, Spain, Sept. 2003.

[10] A. Dempster, N. Laird, and D. Rubin, "Maximum likehood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, vol. 39, pp. 1–38, 1977.

[11] G. Patanè and M. Russo, "The enhanced LBG algorithm," *IEEE Transactions on Neural Networks*, vol. 14, no. 9, pp. 1219–1237, November 2001.

[12] J. Fournier, *Content based image indexing and interactive retrieval*, Ph.D. thesis, UCP, Paris, France, Oct. 2002, Written in French.

[13] Thorsten Joachims, "Transductive inference for text classification using support vector machines," in *Proc. 16th International Conference on Machine Learning*. 1999, pp. 200–209, Morgan Kaufmann, San Francisco, CA.