A Comparison of Active Classification Methods for Content-Based Image Retrieval

Philippe-Henri Gosselin, Matthieu Cord

A Comparison of Active Classification Methods for Content-Based Image Retrieval

Philippe H. Gosselin
ETIS CNRS UMR-8051
University of Cergy-Pontoise
ENSEA
6, av. du Ponceau
95014 Cergy-Pontoise, France
gosselin@ensea.fr

Matthieu Cord
ETIS CNRS UMR-8051
University of Cergy-Pontoise
ENSEA
6, av. du Ponceau
95014 Cergy-Pontoise, France
cord@ensea.fr

ABSTRACT

This paper deals with content-based image indexing and category retrieval in general databases. Statistical learning approaches have been recently introduced in CBIR. Labelled images are considered as training data in learning strategy based on classification process. We introduce an active learning strategy to select the most difficult images to classify with only few training data. Experimentations are carried out on the COREL database. We compare seven classification strategies to evaluate the active learning contribution in CBIR.

1. INTRODUCTION

Content-Based Image Retrieval (CBIR) has attracted a lot of research interest in recent years. This paper addresses the problem of category search, which aims at retrieving all images belonging to a given category from an image database.

Traditional techniques in CBIR are limited by the semantic gap, which separates the low-level information extracted from images and the semantic user request [20, 9]: the user is looking for one image or an image set with semantics, for instance a type of landscape, whereas current processing deals with color or texture features. The increasing database sizes and the diversity of search types contribute to increase the semantic gap. Various strategies have been used to reduce the semantic gap.

Some off-line methods focus on the feature extraction or on the similarity function definition. In computer vision community, some works deal with local descriptor extraction [1, 24] and are concerned with creating indexes invariant to geometric transformations and robust to illumination changes. An image description may be built from local rotational invariant features and spatial constraints [22]. These models try to efficiently catch the visual structures of object categories. Thanks to psycho-visual experiments, Mojsilovic and Rogowitz [16] propose to identify image features and similarity functions which are directly connected to semantic categories. Experiments have also been carried out with user interaction to integrate a user model in a Bayesian similarity function [8]. The aim is to define a similarity between images as close as possible to the human similarity interpretation.

Other strategies focus on the on-line retrieval step to reduce the semantic gap. Interactive systems ask the user to conduct search within the database. Starting with a coarse query, the interactive process allows the user to refine that query as much as necessary. Most of the times, user provides binary annotations indicating whether or not the image belongs to the desired category. The system integrates these annotations through relevance feedback. Interactive retrieval techniques are mainly of two types: statistical and geometrical [26, 18]. The geometrical methods refer to search-by-similarity systems [13, 19]. The objective of the statistical methods is to update a relevance function [2, 8] or a binary classification of images using the user annotations. Recently, statistical learning approaches have been introduced in CBIR context and have been very successful [26, 4]. Discrimination methods (from statistical learning) may significantly improve the effectiveness of visual information retrieval tasks. This approach treats the relevance feedback problem as a supervised learning problem. A binary classifier is learn by using all relevant and irrelevant labelled images as input training data [5].

CBIR has a very specific classification context. There are very few training data during the retrieval process, the input space dimension is usually very high, unlabeled data are available, *etc*. Thus classical learning schemes have to be adapted. We analyze these specificities and propose some classification methods for comparison. We defend that active learning [7] may be helpful to carry out an efficient relevance feedback strategy. Active learning strategies offer a natural framework for interactive image retrieval and very efficient strategies, based on a SVM classification, have been proposed [23]. We introduce in this article an alternative to Tong's method, working as well with SVM classification as other classification methods. Our method, RETIN AL (RETIN Active Learning) is a new version of a previous search-by-similarity system, RETIN, working with both query and similarity updating [11]. Intensive experimentations are carried out on the COREL database. We compare seven classification strategies to evaluate the active learning contribution in CBIR.

2. CLASSIFICATION METHODS

The estimation of the searched category can be seen as a binary classification problem between relevant a class (1) and an irrelevant class (-1). In this section, three generative and discriminative learning methods, Bayes, kNN and SVM, are presented. They have

CVDB '04 Paris, France

been selected for their known classification performances in pattern recognition and CBIR context. To deal with non linearity of input data, all methods are *kernelized*. We denote Kernel functions by $k(.,.)$.

Notations: Let $(\mathbf{x}_i)_{i\in[1,N]}$, $\mathbf{x}_i \in \mathbb{R}^p$ be the feature vectors representing labelled images, and $(y_i)_{i\in[1,N]}$, $y_i \in \{-1,1\}$ be their respective annotations (1 = relevant, -1 = irrelevant). We denote the relevance function, which returns the fellowship to the relevant class for any feature vector $\mathbf{x}$, by $f(\mathbf{x})$.

2.1 Bayes Classifiers

Bayes classifiers is used in text retrieval systems. Since ten years, CBIR community is transposing them to image retrieval [25, 26].

Bayes binary classifiers use the class-conditional likelihood associated with class c $P(\mathbf{x}|c)$ to compute the mapping function $g(\mathbf{x})$ of an input vector $\mathbf{x}$:

$$g(\mathbf{x}) = \underset{c\in\{-1,1\}}{\text{argmax}}\ P(\mathbf{x}|c)P(c) \tag{1}$$

Because we have no prior assumption on the size of a class, we assume that $P(1) = P(-1) = \frac{1}{2}$. Once $g(\mathbf{x})$ is computed, the relevance function $f(\mathbf{x})$ may be expressed as follows:

$$f(\mathbf{x}) = P(\mathbf{x}|c = g(\mathbf{x})) \tag{2}$$

To estimate the probability density function, we use a *kernelized* version of Parzen windows:

$$P(\mathbf{x}|c) = \frac{1}{|\{i|y_i = c\}|} \sum_{i\in\{i|y_i=c\}} K(\mathbf{x},\mathbf{x}_i) \tag{3}$$

where $K(.,.)$ is a kernel function.

2.2 k-Nearest Neighbors

This classification method has been used successfully in image processing and pattern recognition. For instance, in competition with neural networks, linear discriminant analysis (and others), k-Nearest Neighbors performed best results on pixel classification tasks (STATLOG project [15]).

k-Nearest Neighbors classifiers attempt to directly estimate $f(\mathbf{x})$ using only the k nearest neighbors of $\mathbf{x}$: $f(\mathbf{x}) = Ave(y_i|\mathbf{x}_i \in nn_k(\mathbf{x}))$ where Ave denotes the average and $nn_k(\mathbf{x})$ the set of the k points nearest to $\mathbf{x}$ in squared distance. We use a *kernelized* version of these classifiers to better deal with non-linearity [12]:

$$f(\mathbf{x}) = \frac{\sum_{i\in nn_k(\mathbf{x})} y_i K(\mathbf{x},\mathbf{x}_i)}{\sum_{i\in nn_k(\mathbf{x})} K(\mathbf{x},\mathbf{x}_i)} \tag{4}$$

2.3 Support Vector Machines

Support Vector Machines have shown their capacities in pattern recognition, and today know an increasing interest in CBIR [23, 5, 6, 4].

The aim of SVM classification method is to find the best hyperplane separating relevant and irrelevant vectors maximizing the size of the margin (between both classes). Initial method assumes that relevant and irrelevant vectors are linearly separable. To overcome this problem, kernels $k(.,.)$ have been introduced. It allows to deal with non-linear spaces. Moreover, a soft margin may be used, in order to tolerate noisy configuration. It consists in a very simple adaptation by introducing a bound C in the initial equations [27]. The resulting optimization problem may be expressed as follows:

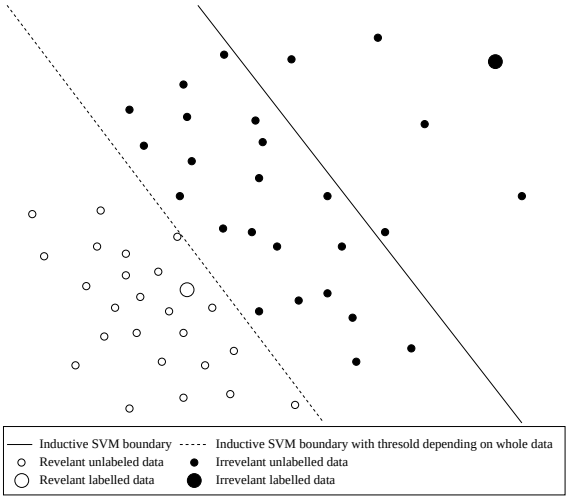

Figure 1: Little data artifact with SVM boundary.

$$\boldsymbol{\alpha}^\star = \underset{\boldsymbol{\alpha}}{\text{argmax}} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i\alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j)$$
$$\text{with} \begin{cases} \sum_{i=1}^{N} \alpha_i y_i = 0 \\ \forall i \in [1,N] \quad 0 \le \alpha_i \le C \end{cases} \tag{5}$$

Thanks to the optimal $\boldsymbol{\alpha}^\star$ value, the distance between a vector $\mathbf{x}$ and the separating hyperplane is used to evaluate how relevant is $\mathbf{x}$:

$$f(\mathbf{x}) = \sum_{i=1}^{N} y_i \alpha_i^\star K(\mathbf{x},\mathbf{x}_i) + b \tag{6}$$

where b is computed using the KKT Conditions [3].

3. TRANSDUCTIVE METHODS

In CBIR, systems have to classify image databases with very few training data. Meanwhile, all unlabeled images are available. If data are structured, unlabeled data should be useful for classification.

When very few labels are available, inductive SVM classification may have unexpected results. Fig. 1 shows such a case. Using only labelled data, the computed boundary is misplaced (full line). Many irrelevant data are misclassified. Such a configuration may happen when learning samples do not represent accurately the structure of data.

LeSaux[21] proposes to adapt the SVM scheme using unlabelled data. Only one parameter (threshold b in Eq. 6) is modified for all the data. In the case of Fig. 1, this method provides a better classification (dotted line), but in the more complex case of Fig. 2, the boundary does not change.

Joachims proposes a method to deal with case of Fig. 2: Transductive SVM [14]. In this particular case, TSVM provides a good classification (dash dotted line). We adapt this approach, proposed in a text retrieval context. This method computes labels for unlabelled data such as hyperplane separates data with maximum margin. We used for experiments the SVM_{light} implementation

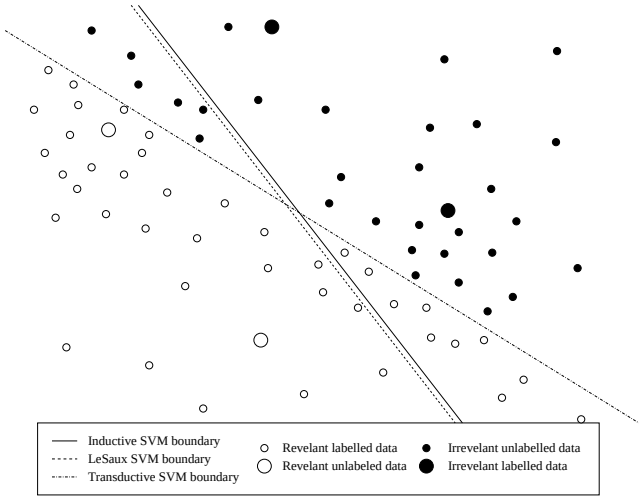

Figure 2: Transductive SVM.

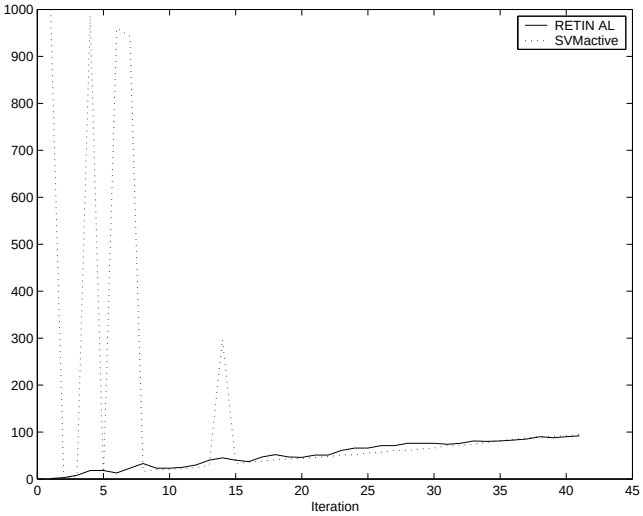

Figure 3: Values of s_j during feedback steps

proposed by Joachims. Unfortunately, no trend becomes apparent with the TSVM use. Actually, we noticed that the transductive approach sometimes improves results, sometimes not. It is very data-dependent, and, of course, time consuming [4].

4. ACTIVE LEARNING

Performances of inductive classification depend on the training data set. In interactive CBIR, all the images labelled during the retrieval session are added to the training set used for classification. As a result, the choice of this labelled images to add will change system performances. For instance, labelling an image which is very close to one already labelled will not change the current classification.

Usual strategies in statistical learning propose to choose elements with the less classification accuracy. Some researchers as Cohn [7], propose to train several classifiers with the same training data, and choose data where classifiers disagree at most. Other ones as Zhu [28], propose to minimize a cost function (the risk) to determine images with the less classification accuracy.

Here, we propose to use two active learning methods. Firstly, we present the well known Tong's SVM$_{active}$ [23], which uses the SVM boundary. Secondly we present our method, which is based on Tong's one, but reduces problems encounted by SVM$_{active}$ during the first iterations. Our method also deals with the sparseness of the training data.

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. 6, which may be seen as a distance to boundary).

4.1 Tong's SVM$_{active}$

The SVM$_{active}$ learning method tries to focus the user on images whose classification is difficult. It asks user to label m images closest to the SVM boundary ($m = 20$ in their experiments [23]). At the feedback iteration j, SVM$_{active}$ proposes to label m images from rank s_j to s_{j+m-1}:

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

In SVM$_{active}$ strategy, s_j is selected so that $\mathbf{x}_{(s_j),j}, ..., \mathbf{x}_{(s_{j+m-1}),j}$ are the closest images to the SVM boundary. The closer to the margin an image is, the less reliable its classification is.

4.2 RETIN Active Learning Strategy

SVM$_{active}$ strategy rests on a strong theoretic foundation and increases performances, but it works with an important assumption: a reliable estimation of the boundary between classes. In classification framework, the training data set approximatively represents 50% of the whole data set. In CBIR, the training set stays very small (even after interaction) in comparison to the database size. In such a context, a reliable estimation of the boundary is not obvious.

We introduce a method with the same principle than SVM$_{active}$ but without using the SVM boundary to find the value s. Indeed, we notice that, even if the boundary may change a lot during the first iterations, the ranking operation is quite stable. Actually, we just suppose that the best s (corresponding to the searched boundary) allows to present as many relevant images as irrelevant ones. Thus, if and only if the set of the selected images is well balanced (between relevant and irrelevant images), then s_j is relevant. We exploit this property to adapt s during the feedback steps.

At the jth feedback step, the user gives new annotations for images $\mathbf{x}_{(s_j),j}, ...\mathbf{x}_{(s_{j+m-1}),j}$. 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: $s_{j+1} = s_j + \lambda \times (r_{rel}(j) - r_{irr}(j))$ For now, we have used this relation with $\lambda = 2$ in all our experiments.

Figure 3 shows values of s_j for the 40th first feedback steps of a retrieval session on a test category (starting with one relevant image and one irrelevant image, 5 annotations per feedback). Both methods have the same behavior after 20 iterations, but the SVM$_{active}$ estimation is very unstable in the first iterations.

Once s_{j+1} is computed, the system should propose to the user the m images from $\mathbf{x}_{(s_{j+1}),j+1}$ to $\mathbf{x}_{(s_{j+1}+m-1),j+1}$. Actually, we also want to increase the sparseness of the training data. Indeed, nothing prevents the system to select for labelling an image close to another already selected. To overcome this problem, we consider exactly the same strategy but working no more on images

but on clusters of images: we compute m clusters of images from $\mathbf{x}_{(s_j),j}$ to $\mathbf{x}_{(s_{j+M-1}),j}$ (where $M = 10 \times m$ for instance), using an enhanced version of LBG algorithm [17]. Next, the system selects for labelling the most relevant image in each cluster. Thus, images close to each other in the feature space will not be selected together for labelling.

5. RETIN AL SYSTEM FRAMEWORK

RETIN AL (Active Learning) is a new version of the CBIR system developed in ETIS laboratory [11].

User interface is compound of two windows (Fig. 4). On top window, images in decreasing order of relevance are displayed, according the current classifier. On bottom window, images proposed for labelling are displayed, according to current active learner. The user is invited to follow advises in bottom window (best labelling according to current active learner), but he can choose to bypass these advises, and do some labelling of his own in the top window.

The user begins a new search with some images of his/her own. System updates the display in both windows. Next, user labels some pictures, and system updates the display, etc., until he/she is satisfied.

During experiment processes, the robot starts with some random pictures in the target category. During feedback, it acts as an user which always clicks in the bottom window.

6. EXPERIMENTS

6.1 Feature Distributions

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 [17]. We take the same class number for both spaces. Tests have shown that $c = 25$ classes is a good choice for all our feature spaces [10]. Image signature is composed of one vector representing the image color and texture distributions. The input size p is then 50 in our experiments.

6.2 Database and evaluation protocol

Tests are carried out on the generalist COREL photo database, which contains more than $50,000$ pictures. To get tractable computation for the statistical evaluation, we randomly selected 77 of the COREL folders, to obtain a database of $6,000$ images. To perform interesting evaluation, we built from this database 11 categories[1] (*cf.* Table 1) of different sizes and complexities. The size of these categories varies from 111 to 627 pictures, and the complexity varies from monomodal (low semantics) to highly multimodal (high semantics) classes, relatively to feature vectors. Some of the categories have common images (for instance, castles and mountains of Europe, birds in savannah). For any category search, there is no trivial way to perform a classification between relevant and irrelevant pictures.

The CBIR system performances are measured using precision(P), recall(R) and statistics computed on P and R for each category. Let us note A the set of images belonging to the category, and B the set of images returned to the user, then: P $= \frac{|A \cap B|}{|B|}$ and R $= \frac{|A \cap B|}{|A|}$. Usually, the cardinality of B varies from 1 to database size, providing many points (P,R).

[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.

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 on an uniform background
savannah	408	animals in African savannah

Table 1: COREL categories for evaluation

We use the average precision P_a which represents the value of the P/R integral function. This metric is used in the TREC VIDEO conference[2], and gives a global evaluation of the system (over all the (P,R) values).

6.3 Comparative methods

We evaluate seven methods:

• Three systems using the presented classification methods (Bayes, kNN and SVM) with a "basic" active learning algorithm: system presents to user to m most relevant unlabelled images.

• Three systems using the presented classification method with the RETIN AL active learning algorithm.

• One system using SVM classification with Tong's active learning algorithm.

The kernel function used for SVM, kNN or Parzen estimation is a Gaussian kernel:

$$K(x, y) = exp^{-\frac{1}{2\sigma} d(x,y)^2} \quad (7)$$

Moreover, the distance in Gaussian kernel may be chosen according to the feature vector type. We use a χ^2 distance which is well suited for vectors representing distributions. As data is normalized, σ is tuned to 1.

When only one kind of labels is provided by user, binary classifications can not be computed. In this case, we use an estimation of the density of the labelled images to rank database, using a one-class SVM method [6].

6.4 Memory needs and computational complexity

The main memory need is the storage of feature vectors (np doubles) and kernel cache lines (nc doubles), where n is the number of images in database, p feature vector dimension, and c the number of lines to cache. Other requirements are negligible against n. In the following experiments, about 3 Mo are used by features vectors, and 10 Mo for kernel cache (as many cache lines as the maximum of labels). With a one million image database, a similar configuration should require 400 Mo for feature vectors, and 1.6 Go for kernel cache.

The main computational needs is the $O(n)$ computation of fellowship to the relevant class (function $f(.)$ in section 2) on the whole database. Other requirements are negligible against n. In the following experiments, with a 10 Mo kernel cache, all methods need at most 2-3 seconds to compute with a Pentium 3 GHz. With a

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

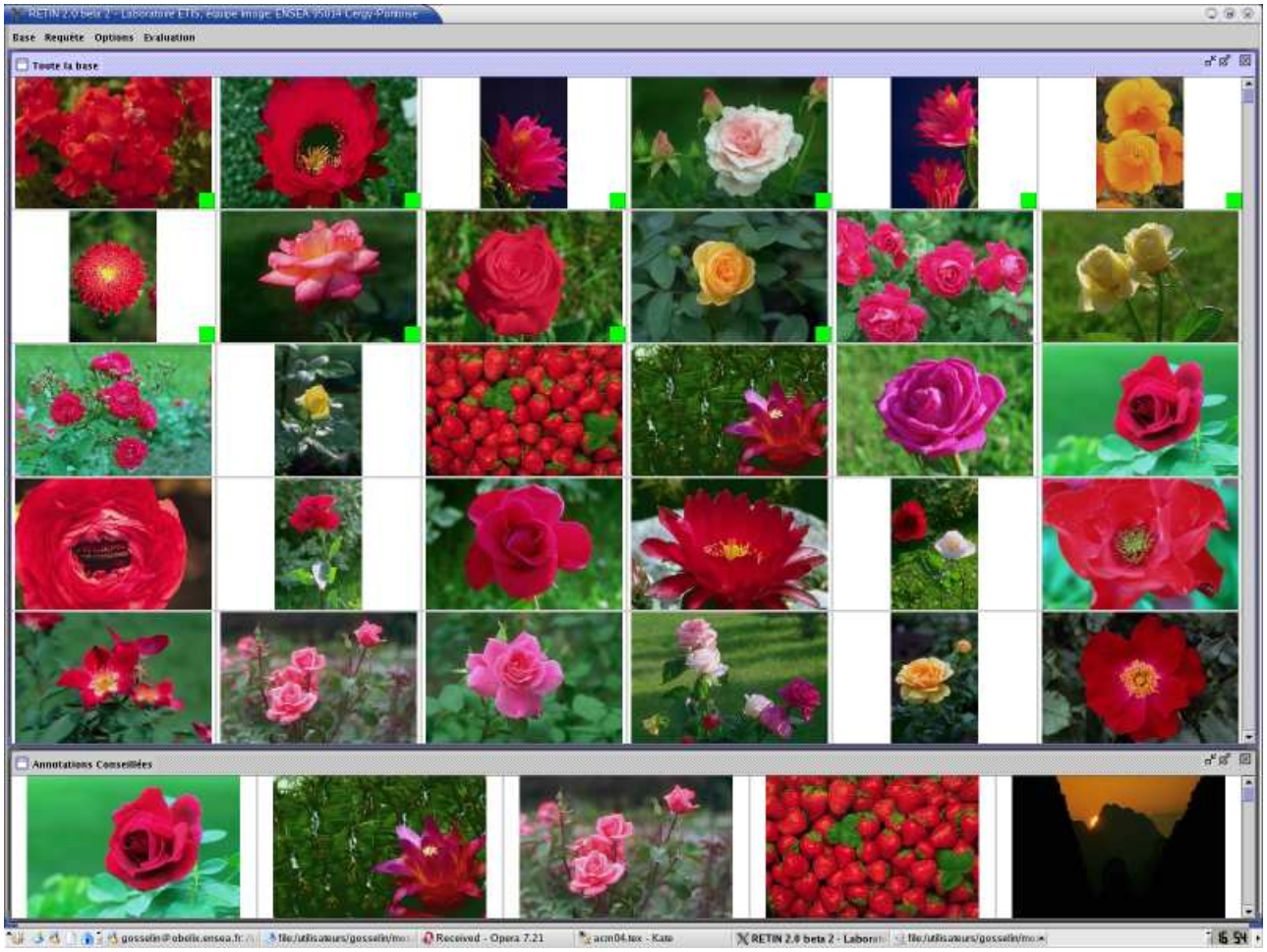

Figure 4: RETIN AL user interface

one million image database, a similar configuration should require about 10 minutes to compute.

6.5 Experiments

Experiments on COREL are very interesting because the database is quite large, with many kinds of categories. In this context, comparison between systems to retrieve large and complex sets of images is meaningful.

We experiment the seven active learners with 3 different contexts:

• 1 relevant image at the beginning of the retrieval process, $m = 5$ annotations per feedback, 40 feedback steps (Table 2);

• 1 relevant image at the beginning, $m = 20$ annotations per feedback, 10 feedback steps (Table 3);

• 11 relevant images and 10 irrelevant images at the beginning, $m = 20$ annotations per feedback, 9 feedback steps (Table 4).

In all cases, 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. Performances on mountains category are also presented in Figure 5 for the 10h first feedback steps.

First, one can notice that the system performances are category dependent. Best results on *birds* category remains very low in comparison to lowest performances on *doors* category. This can be explained by the capabilities of the low-level features to well represent the semantic categories. For instance, *birds* images have very few common colors and textures, while *doors* images have many common features (horizontal and vertical textures). Thus, absolute quality assessment is not relevant, only relative performances are meaningful.

As far as the number m of annotations per feedback is concerned, results with $m = 5$ (*cf.* Table 2) are somewhat better than those with $m = 20$ (*cf.* Table 3). It seems that one can get some improvements using less annotations per feedback, with the same number of training data at the end of the learning process.

Considering only classifiers, SVM is the most adapted to this learning context. The two others ones (Bayes and kNN), always provide lower results. The performance difference between SVM and the two others classifiers varies from 2 to 26 percents.

Focusing on active learners, SVM/RETIN gives the best performances for all categories, followed by SVM/Tong which share those performances for half of the categories. On some categories, none of the active learners improve performance. It seems that those categories are very badly represented by feature vectors, and active learners do not have enough information to act. The good point is that they do not reduce the performance in those cases, where some too optimistic active learners could. On other categories, performances rise, and sometimes up to 7 percents.

category	Bayes/Basic	kNN/Basic	SVM/Basic	Bayes/RETIN	kNN/RETIN	SVM/RETIN	SVM/Tong
birds	19	29	**31**	20	29	**31**	**31**
castles	15	17	**38**	17	18	**38**	**38**
caverns	72	75	77	73	75	**78**	75
dogs	22	28	**58**	21	32	**58**	**58**
doors	86	88	89	91	90	**93**	83
Europe	26	30	33	26	30	**35**	**35**
flowers	56	59	60	64	63	**67**	57
food	58	62	66	64	66	**71**	59
mountains	30	42	**54**	39	39	**54**	**54**
objects	60	69	75	67	67	**78**	76
savannah	56	58	62	60	59	**68**	56

Table 2: Performances: initialization with 1 relevant image, 5 annotations per feedback, 40 feedback steps.

category	Bayes/Basic	kNN/Basic	SVM/Basic	Bayes/RETIN	kNN/RETIN	SVM/RETIN	SVM/Tong
birds	16	27	**29**	17	27	**29**	**29**
castles	14	15	**36**	17	15	**36**	**36**
caverns	70	74	77	72	74	**78**	75
dogs	22	28	**58**	21	32	**58**	**58**
doors	86	88	89	91	90	**93**	83
Europe	26	30	33	26	30	**35**	**35**
flowers	56	59	60	64	63	**67**	57
food	58	62	66	64	66	**71**	59
mountains	30	42	**53**	38	39	**53**	**53**
objects	60	69	75	67	67	**78**	76
savannah	56	58	61	59	59	**67**	55

Table 3: Performances: initialization with 1 relevant image, 20 annotations per feedback, 10 feedback steps.

category	Bayes/Basic	kNN/Basic	SVM/Basic	Bayes/RETIN	kNN/RETIN	SVM/RETIN	SVM/Tong
birds	24	33	36	24	33	**38**	34
castles	15	25	40	16	27	**41**	**41**
caverns	75	76	80	76	76	**81**	78
dogs	39	50	**64**	42	50	**64**	**64**
doors	85	89	90	91	90	**93**	83
Europe	29	33	35	31	33	**36**	**36**
flowers	59	62	65	66	66	**69**	59
food	59	63	69	66	67	**72**	60
mountains	49	46	**55**	50	47	**55**	**55**
objects	71	74	**83**	75	74	**83**	**83**
savannah	56	60	63	61	61	**68**	54

Table 4: Performances: initialization with 11 relevant and 10 irrelevant images, 20 annotations per feedback, 9 feedback steps.

COREL evaluation: system performances estimated with the P_a metric ($\%$), at the end of the interactive learning process. In all experiments, final training data sets exactly contain 201 images.

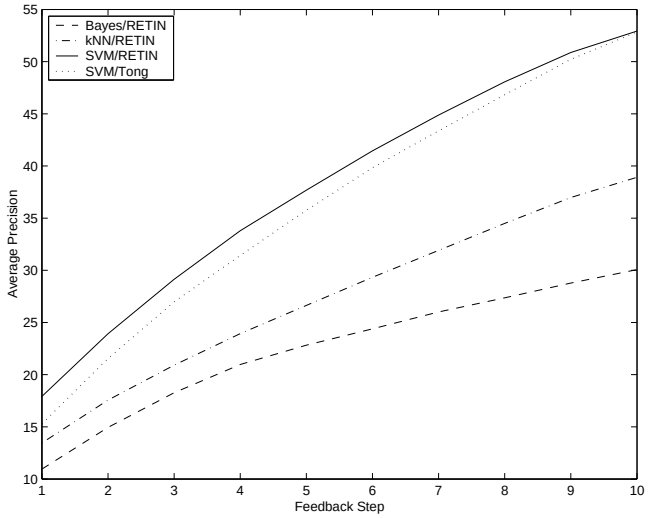

Figure 5: P_a curves according to feedback steps.

Starting with only one relevant image is difficult for the SVM/Tong strategy, because of the inaccuracy of SVM margin with very few training data (*cf.* Table 3). We run experiments with more labelled data at the beginning (*cf.* Table 4), but it is not enough to boost the SVM/Tong strategy.

Finally, whatever the experimental context is, the SVM/RETIN strategy has always *at least* the best performances.

7. CONCLUSION

In this article, an efficient active learning method (RETIN AL) for interactive content-based image retrieval is introduced. Our algorithm selects the most difficult images to classify, without using explicit boundary between relevant and irrelevant images.

We experimentally compare three well-known classification methods (Bayes, kNN and SVM) adapted to CBIR context, combined with active strategies. SVM gives the best results on COREL database. Experiments also show that active learning is improving performances of image retrieval process. RETIN AL strategy is more efficient than the SVM$_{active}$ strategy proposed by Tong *et al*. Unlike SVM$_{active}$, it deals with few training data, and moreover, it may work with any classification method.

Our currently works deal with the evaluation of the scalability of theses techniques in terms of modeling when very large databases are considered. The high accuracy of classifier does not necessary fit with very large databases, where most of the pictures are often out of a given request. We are convinced that, for category search in very large databases, efficient exploration process before classification process will become crucial.

The RETIN system is currently applied to the picture database of the Museum Research and Restoration Center of France (C2RMF), in addition to the European Research Open System (EROS). C2RMF is building a very large database of paintings from France museums, in order to propose an easy access high detailed images of artwork. Common uses of the final retrieval system will be classification (iconographers), semantics (artists, theologists) and dating (restorers).

8. REFERENCES

[1] L. Amsaleg and P. Gros. Content-based retrieval using local descriptors: problems and issues from a database perspective. *Pattern Analysis and Applications*, 4(2/3):108–124, 2001.

[2] G. Caenen, G. Frederix, A.A.M Kuijk, E.J. Pauwels, and B.A.M. Schouten. Show me what you mean! PARISS: A CBIR-interface that learns by example. In *International Conference on Visual Information Systems (Visual'2000)*, volume 1929, pages 257–258, 2000.

[3] C. Campbell. Algorithmic approaches to training support vector machines: A survey. In *ESANN*, pages 27–36. D-Facto Publications, 2000.

[4] E. Chang, B. T. Li, G. Wu, and K.S. Goh. Statistical learning for effective visual information retrieval. In *IEEE International Conference on Image Processing*, Barcelona, September 2003.

[5] O. Chapelle, P. Haffner, and V. Vapnik. Svms for histogram based image classification. *IEEE Transactions on Neural Networks*, 9, 1999.

[6] Y. Chen, X.S. Zhou, and T.S. Huang. One-class svm for learning in image retrieval. In *International Conference in Image Processing (ICIP'01)*, volume 1, pages 34–37, Thessaloniki, Greece, October 2001.

[7] D. Cohn. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

[8] I.J. Cox, M.L. Miller, T.P. Minka, T.V. Papathomas, and P.N. Yianilos. The bayesian image retrieval system, PicHunter: Theory, implementation and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1):20–37, 2000.

[9] J.P. Eakins. Towards intelligent image retrieval. *Pattern Recognition*, 35:3–14, 2002.

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

[11] J. Fournier, M. Cord, and S. Philipp-Foliguet. Retin: A content-based image indexing and retrieval system. *Pattern Analysis and Applications Journal, Special issue on image indexation*, 4(2/3):153–173, 2001.

[12] T. Hastie, R. Tibshirani, and J. Friedman. *The Element of Statistical Learning*. Springer, 2001.

[13] Y. Ishikawa, R. Subramanya, and C. Faloutsos. MindReader: Query databases through multiple examples. In *24th VLDB Conference*, pages 218–227, New York, 1998.

[14] T. Joachims. A statistical learning model of text classification with support vector machines. In *Proceedings of the Conference on Research and Development in Information Retrieval (SIGIR), ACM*, 2001.

[15] D. Michie, D. J. Spiegelhalter, and C. C. Taylor, editors. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, 1994.

[16] A. Mojsilovic and B. Rogowitz. Capturing image semantics with low-level descriptors. In *International Conference in Image Processing (ICIP'01)*, volume 1, pages 18–21, Thessaloniki, Greece, October 2001.

[17] G. Patanè and M. Russo. The enhanced LBG algorithm. *IEEE Transactions on Neural Networks*, 14(9):1219–1237, November 2001.

[18] R. Picard. A society of models for video and image libraries. *IBM Systems Journal*, 35(3/4):292–312, 1996.

[19] Y. Rui and T.S. Huang. Optimizing learning in image retrieval. In *Conf on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 236–243, Hilton Head, SC, June 2000.

[20] S. Santini, A. Gupta, and R. Jain. Emergent semantics through interaction in image databases. *IEEE Transactions on Knowledge and Data Engineering*, 13(3):337–351, 2001.

[21] B. Le Saux. *Classification non exclusive et personnalisation par apprentissage : Application à la navigation dans les bases d'images*. PhD thesis, INRIA, 2003.

[22] C. Schmid. Weakly supervised learning of visual models and its application to content-based retrieval. *International Journal of Computer Vision*, 56(1-2):7–16, January,February 2004.

[23] S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *ACM Multimedia*, 2001.

[24] T. Tuytelaars and L. Van Gool. Content-based image retrieval based on local affinely invariant regions. In *Third Int'l Conf. on Visual Information Systems, Visual99*, pages 493–500, 1999.

[25] N. Vasconcelos. *Bayesian models for visual information retrieval*. PhD thesis, Massachusetts Institute of Technology, 2000.

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

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

[28] X. Zhu, J. Lafferty, and Z. Ghahramani. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In *ICML 2003 workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, 2003.