OPTIMIZATION ON ACTIVE LEARNING STRATEGY FOR OBJECT CATEGORY RETRIEVAL

David GORISSE[1], *Matthieu CORD*[2], *Frederic PRECIOSO*[1]

[1] ETIS, CNRS/ENSEA/UCP, Univ Cergy-Pontoise, France, gorisse, precioso@ensea.fr
[2] LIP6, UPMC-P6, Paris, France, matthieu.cord@lip6.fr

ABSTRACT

Active learning is a machine learning technique which has attracted a lot of research interest in the content-based image retrieval (CBIR) in recent years. To be effective, an active learning system must be fast and efficient using as few (relevance) feedback iterations as possible. Scalability is the major problem for such an on-line learning method, since the complexity of such methods on a database of size n is in the best case $O(n * log(n))$. In this article we propose a strategy to overcome that limitation. Our technique exploits ultra fast retrieval methods like Locality Sensitive Hashing (LSH), recently applied for unsupervised image retrieval. Combined with active selection, our method is able to achieve very fast active learning task in very large database. Experiments on VOC2006 database are reported, results are obtained four times faster while preserving the accuracy.

Index Terms— active learning, image retrieval, relevance feedback, support vector machines, locality sensitive hashing

1. INTRODUCTION

Active learning is an extension of semi-supervised learning which not only exploits non-annotated data but also proposes to the user, considered as an expert, a choice of images to be annotated. In active classification, the images to be annotated by the user are chosen such that the classification error on the database is the lowest [1]. Active learning is particularly relevant in image interactive retrieval since only few annotations can be required from the user to define the training set. Thus the training set is small, the annotations must then provide the best classification. This specific process, compared to simple classification methods, is called *selection problem*.

However, this process of image selection has, at best, a linear computational complexity in the number of images in the database for each feedback iteration. When the database becomes very large, this scheme becomes untractable.

In the context of copy detection [2] or in the context of image retrieval [3], methods have recently been proposed to overcome this problem of scalability. In this paper, the idea is to propose a new approach for fast selection to exploit active learning techniques on image retrieval in very large databases.

2. COMPUTATIONAL COMPLEXITY OF ACTIVE LEARNING IN CBIR

In CBIR classification framework, retrieving classes of images is usually considered as a two-class problem: the relevant class, the set of images corresponding to the user query concept, and the irrelevant class composed by the remaining database.

Let $\{\mathbf{x}_i\}_{1,n}$ be the n image indexes of the database. A training set is expressed from any user label retrieval session as $\mathcal{A} = \{(\mathbf{x}_i, y_i)_{i=1,n} \mid y_i \neq 0\}$, where $y_i = 1$ if the image $\mathbf{x}_i$ is labeled as relevant, $y_i = -1$ if the image $\mathbf{x}_i$ is labeled as irrelevant and $y_i = 0$ otherwise. The classifier is then trained using these labels, and a relevance function $f_{\mathcal{A}}(\mathbf{x}_i)$ is determined in order to be able to rank the data. The set of unlabelled images is denoted by $\mathcal{U}$. In this paper, image descriptors are given by a compact and efficient representation of visual features: adapted histograms of colors and textures.

We consider, in this paper, active learning classification which aims at minimising classification error over the whole set $\mathcal{B}$ of images from the database ($\mathcal{B} = \mathcal{A} + \mathcal{U}$) by considering the user as an expert and requiring to iteratively annotate carefully chosen images.

This expert can be represented by a function $s : \mathcal{B} \rightarrow \{-1, 1\}$, which assigns a label to an image of the database. In active classification, the images iteratively presented to the user to be annotated are chosen such that the classification error on the database is the lowest. This iterative annotating process is called *relevance feedback*. In the following, the index t corresponds to the t^{th} iteration of labelling process.

In the case where only one image $\mathbf{x}_i$ has to be selected, this turns to the minimisation of classification error on $\mathcal{B}$ over all the functions $f_{\mathcal{A}_t}(s(\mathbf{x}_i))$ of classification on the previous training set $\mathcal{A}_t$, at iteration t of relevance feedback loop, augmented with the annotation $s(\mathbf{x}_i)$ of the image $\mathbf{x}_i$:

$$i^\star = \arg\min_{i \in U} R_{test}(f_{\mathcal{A}_t}(s(\mathbf{x}_i))) \tag{1}$$

with $R_{test}(f_{\mathcal{A}})$ a risk function, which can have different definitions depending on the approximation introduced in its evaluation.

For instance, Roy & Mc Callum [4] propose a technique to determine the data $\mathbf{x}_i$ which, once added to the training set $\mathcal{A}$ with the user annotation $s(\mathbf{x}_i)$, minimizes the error of generalization. This problem cannot be directly solved, since the user annotation $s(\mathbf{x}_i)$ of each $\mathbf{x}_i$ image is unknown. Roy & Mc Callum [4] thus propose to approximate the risk function $R_{test}(f_{\mathcal{A}})$ for both possible annotation, positive and negative. The authors propose to approximate $P(y|\mathbf{x}_i)$, the probability that the $\mathbf{x}_i$ image is labelled y, using an *a posteriori* probability $P_{\mathcal{A}+y'e_i}(y|\mathbf{x}_i)$ with y' the annotation of $\mathbf{x}_i$. The labels $s(\mathbf{x}_i)$, being unknown on $\mathcal{U}$, are estimated by training 2 classifiers for both possible label on each unlabelled data $\mathbf{x}_i$.

Such a classification method implies a computational complexity of $O(|\mathcal{U}|^3)$.

Tong et al.[5] proposed a selection method, SVM_{active}, in the case of one image selection, which is fast, with strong mathematical foundations and which is a classic reference in many papers to compare image retrieval approaches. Their approach is based on the minimization of the set of separating hyperplanes.

A relevance function $f_{\mathcal{A}}$, adapted from the membership to a class (distance to the hyperplane for SVM), is trained. Using this relevance function, uncertain data $\mathbf{x}$ will be close to 0: $f_{\mathcal{A}}(\mathbf{x}) \sim 0$.

The solution to the minimization problem in eq. 1 is:

$$i^{\star} = \arg\min_{i \in \mathcal{U}} (|f_{\mathcal{A}}(\mathbf{x}_i)|) \tag{2}$$

The efficiency of this method depends on the accuracy of the relevance function estimation close to the boundary between relevant and irrelevant classes. Tong et al. strategy [5] requires to compute the relevance function on the whole database and to sort the relevance scores obtained. The complexity of this stage is in $O(|\mathcal{U}| * log(|\mathcal{U}|))$, with $\mathcal{B}$ the whole database, which makes the interactive search impossible when the size of the database becomes too large.

Many methods have been proposed to pre-select data to be annotated, but the best schemes remain linear in the size of the database.

3. APPROXIMATION SCHEME

In this paper, we propose a method to pre-select data to be presented to the user which is sub-linear in the size of the database. Following the same idea as Tong et al. [5], our system is going to present the n most uncertain images regarding the classification function. However, we propose to not consider all the images from the whole image set $\mathcal{B}$ but provide an optimised selection of images to be annotated in order to approximate the relevance function $f_{\mathcal{A}}$.

3.1. Selection strategy

To decrease the complexity, we propose to carefully select a relevant subset of images $\mathcal{S}$ and to only look for the most uncertain images in this subset. This data subset must be as small as possible to decrease as much as possible the computational complexity and must contain the most uncertain images, i.e. images at the boundary between relevant and irrelevant images. At the beginning of the interactive search, the relevance function is not accurate, i.e. we have few knowledge of the boundary. As a consequence, we do not know if the most uncertain images at first iterations are really at the boundary of real classification. However, we know that the size of relevant images is smallest than the size of irrelevant images. It follows that a positively annotated image is most likely to be at the real boundary than a negatively annotated image. For this reason, our strategy to build $\mathcal{S}$ is to add the nearest neighbors of each positively annotated image $(\mathbf{x}_i, +1) \in \mathcal{A}$. This modified optimization scheme is interesting if the computation of $\mathcal{S}$ is fast. Instead of doing a linear scan for the k-NN search, we use an efficient indexing scheme based on LSH, which will be detailed below.

The subset $\mathcal{S}$ is define as $\bigcup_i LSH(\mathbf{x}_i, +1) \bigcap \mathcal{U}$. Thus, the relevance function $f_{\mathcal{A}}$ is evaluated on the n most uncertain data in $\mathcal{S}$, i.e. the n images with a relevance value closest to 0. As we will show in the result this approach allows us to reduce highly the computational time of retrieval compared to Tong et al. approach.

3.2. LSH indexing

We shortly report in this section the basic LSH principle to explain how we use it in our context.

LSH solves the $(R, 1+\epsilon)$-NN problem: finding, on a given space, vectors that are at a distance of $(1+\epsilon)R$ to a query vector. The fundamental principle relies on the construction of a hash table using a hash function instead of sorted data. The hash function associates a vector to a key and each key allows to access a bin of a hash table. Hash function has the property of associating vectors to the same key with higher probability when they are close to each other. To avoid boundary effects, many hash tables are generated.

Indyk and Motwani [6] solved this problem for the Hamming metric with a complexity of $O(n^{1/(1+\epsilon)})$ where n is the number of vectors of the database.

Datar et al. [7] proposed an extension of this method which solves this problem with the Euclidean metric and with similar time performances.

The hashing function works on tuples of random projections of the form: $h_{\mathbf{a},c}(\mathbf{b}) = \left\lfloor \frac{\mathbf{a}.\mathbf{b}+c}{w} \right\rfloor$ where $\mathbf{a}$ is a random vector whose each entry is chosen independently from a Gaussian distribution, c is a real number chosen uniformly in the range $[0, w]$ and w specifies a bin width (which is set to be constant for all projections).

Each projection splitting the space by a random set of parallel hyperplanes; the value of the hash function indicates in which slice of the space, beteween two hyperplanes the vector has fallen.

The three parameters chosen for this algorithm are the radius R, the number of projections K and the number of hash tables L.

3.3. Our algorithm scheme

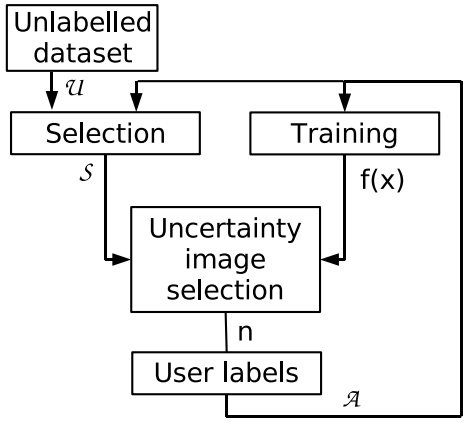

Fig. 1. scheme of a learning loop

A learning loop is summarized on the fig.1.

Training: A binary classifier is trained with the labels $\mathcal{A}$ the user has given. In this paper, we use a SVM with a Gaussian l_2 kernel to be consistent with the k-NN search. The result is a relevant function $f_{\mathcal{A}}$.

Selection: Annotated images $\mathcal{A}$ are also used to select among the unlabelled dataset $\mathcal{U}$ the relevant subset of images $\mathcal{S}$ as described in the section 3.1.

Uncertainty image selection: The system compute for each image of $\mathcal{S}$ a measure of uncertainty using Tong approach ($|f_{\mathcal{A}}(x)|, \forall x \in \mathcal{S}$). The n most uncertain images (data closest to 0) are shown to the user.

This learning loop is repeated a few times. At each iteration we can show to the user a preliminary result by ranking the set of selected images $\mathcal{S}$ using the relevant function $f_{\mathcal{A}}$. An example of preliminary result is given on the fig.2.

4. CLASSIFIER UPDATING

The problem of scalability appears again when the function $f_{\mathcal{A}(x)}$ needs to be computed to rank the database. As the user is only interested in the rank of the N most relevant images, called $TOPN$, we can decrease the complexity in computing the function $f_{\mathcal{A}}(x)$ only on a carrefully chosen subset $\mathcal{S}_u$. Reminding that the subset $\mathcal{S}$ contains images close to the positive annotations, i.e, images that have a *high probability* to be at the top of the rankink, we chose $\mathcal{S}_u = \mathcal{S}$.

5. EXPERIMENTS

Our experiments aims to prove that our active learning scheme on a selected relevant image subset is as efficient as Tong approach [8] while decreasing the computational complexity of image retrieval task.

5.1. Experimental setup

We perform evaluation of our method on the 10-class VOC2006 datasets [9] which contains 5,304 images. The goal of our system is to learn a category of images through a relevance feedback process. The SVM Active learner has no prior knowledge on the image categories. Each image is represented by a 192-dimension vector obtained by concatenating 3 histograms, one of 64 chrominances value from $CIEL^*a^*b^*$ and two of 64 textures from Gabor filters. We use Gaussian kernel function with l_2 distance. Each retrieval session is initialized with 15 relevant images and 5 irrelevant images. Then, at each iteration, 5 images chosen by the active learning system are labelled either positively or negatively. This process is repeated 25 times. At the end of the retrieval session, the training set is made of 145 labeled images. An illustration of the graphical interface of our system RETIN [10] is given on fig.2. Performances are evaluated with Mean

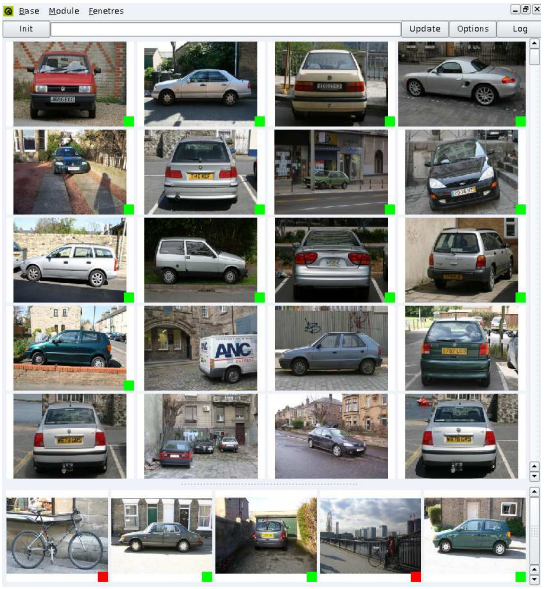

Fig. 2. Graphical interface of our system. Top part: retrieved images; Button part: Images selected by the active learner; Green square: image labelled positively; Red square: image labelled negatively

Average Precision of the $TOP500$, i.e., the sum of the Precision/Recall curve for the first 500 images retrieved. We chose as parameter of E^2LSH [11] a radius of $R = 16.0$ and $L = 30$ hash tables of $K = 20$ projections.

5.2. Results

Results are given on the fig.3. As we can see our method provides better results than Tong approach for the first iterations. Indeed, after initialisation, we obtain a MAP of 9.24% which is 6.5% better than Tong results. After some iterations, Tong approach gives better results, but our method remains competitive. In the worst case, iteration 21, we obtain a MAP of 16.38% with Tong approach while our method gives a MAP of 15.83%. For some categories, like the class of car, our system provides even better results regardless the number of iterations (fig. 4).

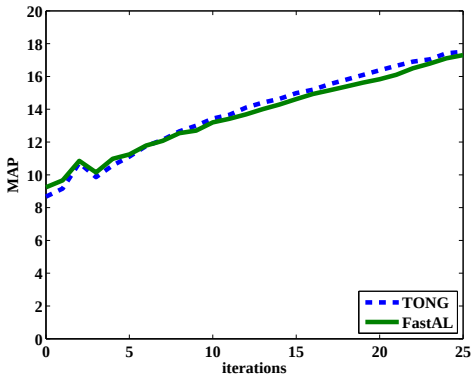

Fig. 3. Mean Average Precision(%) of the 10 classes

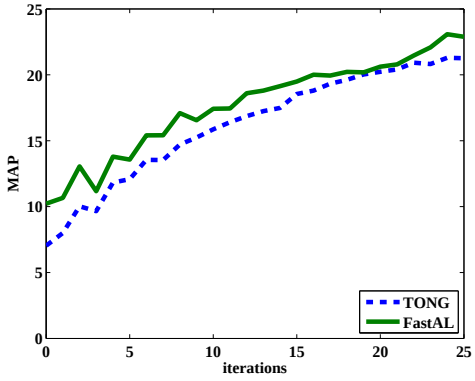

Fig. 4. Mean Average Precision(%) for an interactive search of the class CAR

As shown on fig.5, our method is about 4 times faster than Tong method. For a classification consisting of 25 iterations, our algorithm will take 0.72 second while using the Tong method will take 3.18 seconds. The classification is 4.4 times faster for a similar result.

6. CONCLUSION

Active learning has been proved to be particularly relevant in image interactive retrieval task since only few annotations can be required from the user. Thus the training set is small, the annotations must then provide the best classification. This selection process of image to be annotated is one of the key aspects for scalability of active learning methods. We have proposed a strategy to overcome this scalability problem with a preselection stage which quickly selects images to be annotated by the user. Experimental results on VOC2006 show that our algorithm achieves same accuracy than one of the reference methods, Tong approach, while dividing the computational complexity by four.

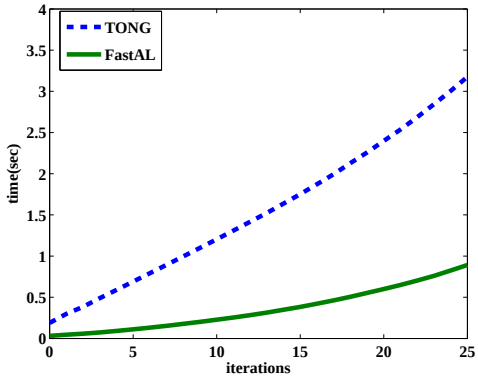

Fig. 5. average time of an interactive search function of the number of iteration

7. ACKNOWLEDGMENT

The authors are grateful to A. Andoni for providing the package E^2LSH and to P.-H. Gosselin for providing RETIN System.

8. REFERENCES

[1] Steven C.H. Hoi, Rong Jin, Jianke Zhu, and Michael R. Lyu, "Semi-supervised svm batch mode active learning for image retrieval," in *IEEE CVPR*, 2008.

[2] E. Valle, M. Cord, and S. Philipp-Foliguet, "High-dimensional descriptor indexing for large multimedia databases," *ACM*, 2008.

[3] D. Gorisse, M. Cord, F. Precioso, and S. Philipp-Foliguet, "Fast approximate kernel-based similarity search for image retrieval task," in *ICPR*. dec 2008, IEEE.

[4] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *ICML*, 2001, pp. 441–448.

[5] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *JMLR*, vol. 2, pp. 45–66, 2002.

[6] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," *ACM*, pp. 604–613, 1998.

[7] M. Datar, N. Immorlica, P. Indyk, and V.S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," *SCG*, pp. 253–262, 2004.

[8] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," *ACM*, pp. 107–118, 2001.

[9] M. Everingham, A. Zisserman, C. K. I. Williams, and L. Van Gool, "Pascal voc2006," .

[10] J. Gony, M. Cord, S. Philipp-Foliguet, P.H. Gosselin, and F. Precioso, "Retin: a smart interactive digital media retrieval system," in *ACM CIVR*, 2007, pp. 93–96.

[11] A. Andoni, "E2lsh," http://www.mit.edu/∼andoni/LSH/.