CBIR IN DISTRIBUTED DATABASES USING A MULTI-AGENT SYSTEM

David PICARD, Matthieu CORD, Arnaud REVEL

ETIS, ENSEA/UCP/CNRS UMR 8051
6, avenue du Ponceau, 95014 Cergy Cedex France
picard@ensea.fr, cord@ensea.fr, revel@ensea.fr

ABSTRACT

Information retrieval techniques have to face both the growing amount of data to be processed and the "natural" distribution of these data over the network. Hence, we introduce in this paper a new architecture for image retrieval in distributed image databases, based on multi-agent systems. Our system, inspired by "ant-agents", uses labels provided by the user for learning both the searched category of images and the path to the most relevant databases. We then show how effective can be our architecture on a generalist image database network.

***Index Terms*—** Image recognition, Information retrieval, Image databases, Distributed database searching, Cooperative systems

1. INTRODUCTION

The number of collections of images on the Internet, at home... grows more and more since the proliferation of digital equipments. In order to manage these large collections, powerful system assistants are required. Traditional techniques in content-based image retrieval (CBIR) are limited by the semantic gap between the low-level representations of images based on color, texture and shape analysis, and the semantic subsets of the database the users are looking for [1]. The increasing database sizes and the diversity of search types contribute to amplify the semantic gap.

Interactive learning approaches have been introduced in CBIR context to improve the effectiveness of visual information retrieval tasks [2, 3]. The largest improvement is definitely obtained by using active learning strategies optimizing the selection of images to present to the user [4]. Recently, we have introduced an active learning strategy [5] to carry out an efficient relevance feedback working as well with SVM as other classification methods.

In this article, the problem of interactive retrieval into distributed databases context is considered. Although this context is close to real applications (search engines[1] or the peer-to-peer networks[2]), it has been the focus of very few researches ([6]).

[1] http://www.google.com
[2] http://www.bittorrent.com/introduction.html

To carry out the search, people usually consider centralized systems: images on remote databases are collected and downloaded on a central database on which classical CBIR techniques are applied.

We propose in this article a different approach no more centralized. It is based on Multi-Agent Systems (MAS) offering interesting properties in comparison with centralized systems [7]:

- since the agents are distributed, several sub-tasks can be processed in parallel, thus sparing both CPU and bandwidth
- if a machine is down, machines still up can keep on processing the search task

We introduce a new Multi-Agent System architecture dedicated to image retrieval on distributed databases. We recently developed a MAS system based on ANT algorithms [8] for text retrieval on the web [9, 10]. That allowed us to validate the global properties of our MAS-based search system on distributed machines. We propose here a MAS architecture integrating a new formulation for image retrieval agent and selection. In particular, we show that the classical active learning strategies [4] cannot be efficiently used without adaptation to the database-distributed search context. We then introduce an active learning strategy dedicated to that context. Afterwards, we present results comparing our active strategy to classical strategies, and an evaluation of the global effectiveness of our architecture.

2. ARCHITECTURE OF OUR SYSTEM

2.1. The context

We consider a network of computers where some of the machines host image databases and form the locations where images can be retrieved from (See figure 1). On each of these machines (Image Servers - IS), a local static agent (Local Image-Database Agent - LIDA) keeps up-to-date an index of the local database.

Practically, we want the user to be able to easily and efficiently search images among those databases. For that purpose, he starts the search by giving a query image to the

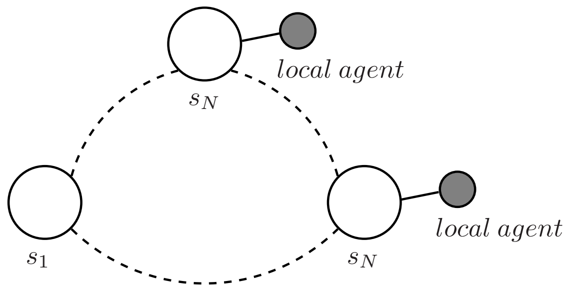

Fig. 1. Mobile agents explore the network, starting from s_1, while local agents on s_N keep an index of the local image database up-to-date.

human-machine interface program running on his computer (s_1). The system then distributes the queries to the machines hosting the image databases and propose to the user a set of images.

Active learning aims at optimizing the interaction between the user and the system. The user is able to label any document in the database but does not want to label a lot of documents. Active strategies attempt to significantly improve the results with a small set of labelled documents.

We guess that, as the categories searched by the user could be concentrated on specific local databases, the quality of the answers could be improved by facilitating answers coming from these machines. Besides, it should also improve the quality of the active learning since the system classifying the images should be restricted to only a subset of the concatenation of all the image databases available on the network.

In the next section, we develop the principle of the MAS we use to select the "best" databases according to the category searched by the user. We then develop the active learning strategy we have used.

2.2. The ant system

Practically, once the HMI has received the query image, it launches N mobile agents that move over the network in order to find the local agents maintaining image databases. These moving agents obey the following algorithm based on the reinforcement of "pheromone-like" markers (See **Algo 1**).

Algo 1: Mobile-agent behavior

INITIALIZATION: $s_1 \rightarrow s_c$
Do
  If $s_c = s_N$ `// A local agent is found!`
    Go back to the HMI
  Else `// Look forward`
    Go to site s_j with probability $\dfrac{Ph_j(t)}{\sum_{l=1}^{k} Ph_l(t)}$
    with $k = |succ(s_c)|$
  End If
While s_N not found or $prof < \theta$

With s_j being the computers in the network, s_c the current site explored by the agent and $succ(s_c)$ the sites directly following s_c. $Ph_k(t)$ is the pheromone level on the site s_k (t is discrete and evolves only when the Ph are updated). θ is a threshold corresponding to a maximum exploration depth.

When a mobile agent finds a local image-database agent, it sends a request to this agent with the description of the searched images. Thanks to a specific active learning strategy (See Section 2.3), the local agent sends back a series of image in accordance with the description given by the mobile agent. Once the mobile agent gets the answers, it comes back to the user's computer and proposes them to the HMI. Actually, the answers of all the mobile agents are collected into a pool until a timeout is reached. Then, N_I images are selected from the pool and proposed to the user for labelling. For each label, a reinforcement signal is sent back to all the computer on the route to the corresponding image database in order to update the pheromone level on the pathway taken by the mobile agent who brought this image. Positive labelling increases the pheromone level, whereas negative labeling decreases it, according to the following rules:

$$\begin{aligned} Increase: &\quad Ph_k(t+1) = Ph_k(t) + \beta \\ Decrease: &\quad Ph_k(t+1) = (1-\alpha)\cdot Ph_k(t) \end{aligned}$$

Such reinforcement rules are known to optimize the path to the sites containing relevant information ([11, 8, 10]).

As soon as images are given, the mobile agents continue their exploration of the network, searching for new examples to be learned. Once a "sufficient" number of images has been learned, mobile agents are launched a very last time in order to return the most relevant images. Results are then ranked and presented to the user.

2.3. Our active learning system

In this paper, we focus on the active learning scheme where a pool of unlabeled examples is available. We suppose that we have a set $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ of image descriptions, a set of labels $\mathbf{y} = (y_1, \ldots, y_N)$ (1 relevant, -1 irrelevant, 0 unknown), a relevance function $f_{\mathbf{y}} : \mathbf{X} \rightarrow [-1, 1]$ trained with $\mathbf{y}$, and a teacher $\tau : \mathbf{X} \rightarrow \{-1, 1\}$ that labels documents as -1 or 1 (the same labels as in 2.2). The aim of an active learning within this context is to choose the unlabeled document $\mathbf{x}$ that will enhance the most the relevance function trained with the label $\tau(\mathbf{x})$ added to the previous labelling $\mathbf{y}$.

Uncertainty-based sampling is the active strategy the most used in image retrieval. This strategy aims at selecting unlabeled image that the learner of the relevance function is most uncertain about. The first solution is to compute a probabilistic output for each image, and select the unlabeled images with the probabilities closest to 0.5 [12]. Similar strategies have been also proposed with SVM classifier [13], with a theoretical justification [14], or with nearest neighbor classifier [15].

In all cases, a relevance function may be computed. This function can be a distribution, a fellowship to a class (distance to the hyperplane for SVM), or a utility function. Thus, with some adaptation of each approach, a relevance function $f_{\mathbf{y}}$ is trained, where the most uncertain documents have an output close to 0.

In our architecture, we choose a SVM classifier as relevance function. This classifier is updated with the images brought back by the mobile agents. When departing, mobiles agents get a copy of the latest classifier (*ie*, trained on more examples). The exploration made by mobile agents and the learning procedure of the classifier are entirely asynchronous.

On the IS, the LIDA (see 2.1) computes the relevance of each image in the local database D, and selects a answer set of I images given an active strategy. As many agents arrive with the same relevance function, the active strategy should not answer in a deterministic way, otherwise all these mobile agents will get the same set, and thus act as one single agent.

We adapted the SVM_{active} strategy of Tong [4], in order to take advantage of the MAS structure. All images $\mathbf{x}$ in the database D are ranked ($r(\mathbf{x})$) given their distance to the classifier's hyperplane. We associate a Gaussian probability $P(\mathbf{x})$ to each image based on this ranking :

$$P(\mathbf{x}) \propto e^{-\frac{1}{2}(\frac{r(\mathbf{x})}{\sigma})^2} \tag{1}$$

To obtain the discrete probability, $P(\mathbf{x})$ is normalized over the local database D :

$$P(\mathbf{x}) = \frac{e^{-\frac{1}{2}(\frac{r(\mathbf{x})}{\sigma})^2}}{\sum_{\mathbf{x}_i \in D} e^{-\frac{1}{2}\left(\frac{r(\mathbf{x}_i)}{\sigma}\right)^2}} \tag{2}$$

I images are randomly drawn thanks to the propability $P(\mathbf{x})$ over the $\mathbf{x}_i \in D$. We compute σ so that the set of the I nearest images from the hyperplan have a probability of p :

$$p = \sum_{r(\mathbf{x}_i)=1}^{I} P(\mathbf{x}_i) \tag{3}$$

p is the only parameter we tune, and represents the exploration in the local database D. This allows any image in the database to be selected, even if the images are far from the hyperplan. As I images are selected, if $p = 1$, the returned images are exactly the I nearest images, which is the SVM_{active} strategy.

3. EXPERIMENTS

Our test database is an excerpt of the *Corel* database which contains 6,000 images categorized in 50 concepts of 50 to 300 images. The signatures used for the images consists of 50 features : 25 colors and 25 textures, based on previous work ([16]).

In order to validate our new active learning strategy, and to tune the parameter p, we first consider experiments without the MAS context. For each image in a concept, we made a learning session which consisted of ten feedback rounds. At each round, the active strategy selected ten images and updated the classifier consequently. The experiment was run for several values of p and we compute the Mean Average Precision (MAP) to evaluate the learning quality. Results for the categories *dogs* and *mountains* are shown in Fig. 3:

p	0.3	0.4	0.5	0.6
$dogs$	0.311	0.321	0.342	0.348
$mountains$	0.367	0.361	0.354	0.351
p	0.7	0.8	0.9	1
$dogs$	0.335	0.343	0.335	0.344
$mountains$	0.354	0.351	0.348	0.345

Fig. 2. MAP for different categories given the parameter p. $p = 1$ stands for SVM_{active}.

The parameter p seems to have little influence beyond 0.5, as the MAP is quite similar to SVM_{active} ($p = 1$). We determined $p = 0.6$ as a good compromise between exploration of the feature space (low value of p) and determinism within the returned training set (high value of p).

In order to see the influence of the repartition of the researched category on our MAS system, we split the database into two smaller databases hosted on two different machines (A and B). This repartition c varies from 50% (the category is equally distributed on A and B) to 100% (the category is entirely on B). Eight mobile-agents were launched, each of them carrying back two images. The classifier's update and the network exploration are asynchronous, as described in Section 2. The graphic on Fig. 3 shows experiments without the pheromone update (*i.e.*, mobile-agents randomly explore the network) and with the pheromones update. A processing of the whole database is pending.

The learning scheme without the pheromones performs a little better than the centralized implementation for a strong dilution of the class among the databases (ie, equally distributed over A and B), showing that the active learning strategy performs better on smaller databases. For high concentrations of the class (ie, all images on A), the performances decreases. The learning scheme with the pheromones is as good as other implementations for strong dilution of the searched class : as relevant information can be retrieved both from A and B, no path can be efficiently learned. For high concentration of the class on a specific database, this implementation outperforms others. This shows how the path-learning strategy combined to the active learning strategy really improve the results.

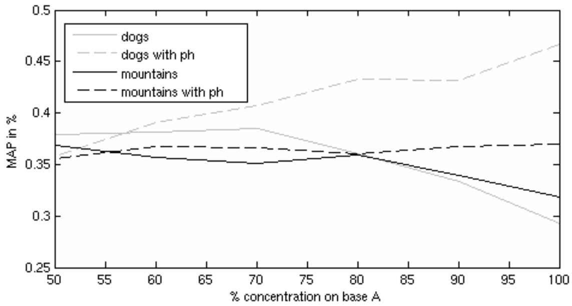

Fig. 3. MAP with (dashed line) and without (solid line) pheromones, depending on the concentration of the categories $dogs$, $mountains$ on base A.

4. CONCLUSION

In this paper, we presented a new architecture for image retrieval in distributed databases. Based on MAS, this system uses labels provided by the user for both training a classifier on the researched category, and reinforcing the path leading to the relevant information. This led us to adapt an efficient active learning strategy to this context. Experiments were made on a generalist database, split into two smaller databases, in order to see the influence of the localization of the researched category. We have shown that our system is as good as centralized solutions in all cases, and strongly increases the global efficiency of the retrieval in the case where the researched category is concentrated on a specific database.

5. REFERENCES

[1] S. Santini, A. Gupta, and R. Jain, “Emergent semantics through interaction in image databases,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 3, pp. 337–351, 2001.

[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] 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.

[4] S. Tong and E. Chang, “Support vector machine active learning for image retrieval,” in *ACM Multimedia*, 2001.

[5] P.H. Gosselin and M. Cord, “RETIN AL: An Active Learning Strategy for Image Category Retrieval,” in *IEEE International Conference on Image Processing (ICIP)*, Singapore, October 2004.

[6] Volker Roth, Ulrich Pinsdorf, and Jan Peters, “A distributed content-based search engine based on mobile code,” in *SAC ’05: Proceedings of the 2005 ACM symposium on Applied computing*, New York, NY, USA, 2005, pp. 66–73, ACM Press.

[7] E. Bonabeau, M. Dorigo, and G. Theraulaz, “The social insect paradigm for optimization and control,” *Nature*, vol. 406, pp. 39–42, 2000.

[8] M.Dorigo, V. Maniezzo, and A. Colorni, “The ant system: Optimization by a colony of cooperating agents.,” *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, vol. 1, no. 26, pp. 29–41, 1996.

[9] A. Revel, “Web-agents inspired by ethology: a population of “ant”-like agents to help finding user-oriented information.,” in *IEEE WIC’2003 : International Conference on Web Intelligence.*, Halifax, Canada, October 2003, IEEE, pp. 482–485, IEEE Computer Society.

[10] A. Revel, “From robots to web-agents: Building cognitive software agents for web-information retrieval by taking inspiration from experience in robotics,” in *ACM International conference on Web Intelligence*, Université Technologique de Compiègne, Compiègne, France, Septembre 2005, pp. 434–437.

[11] J.L. Deneubourg, S. Goss, N.Franks, A. Sendova-Franks, C. Detrain, and L. Chrétien, “The dynamics of collective sorting: Robot-like ants and ant-like robots,” in *From Animals to Animats: Proc. First Int. Conference on Simulation of Adaptive Behavior*, J-A. Meyer and S.W. Wilson, Eds., Paris, France, 1990, pp. 356–363.

[12] D. Lewis and J. Catlett, “Heterogenous uncertainty sampling for supervised learning,” in *International Confenrence on Machine Learning*, 1994.

[13] J. Park, “On-line learning by active sampling using orthogonal decision support vectors,” in *IEEE Neural Networks for Signal Processing*, 2000.

[14] Simon Tong and Daphne Koller, “Support vector machine active learning with applications to text classification.,” *Journal of Machine Learning Research*, vol. 2, pp. 45–66, 2001.

[15] Michael Lindenbaum, Shaul Markovitch, and Dmitry Rusakov, “Selective sampling for nearest neighbor classifiers,” *Machine Learning*, vol. 54, no. 2, pp. 125–152, 2004.

[16] 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*, vol. 4, no. 2/3, pp. 153–173, 2001.