A Statistical Framework for Image Category Search from a Mental Picture

Marin Ferecatu and Donald Geman, *Senior Member*, *IEEE*

Abstract—Starting from a member of an image database designated the "query image," traditional image retrieval techniques, for example, search by visual similarity, allow one to locate additional instances of a target category residing in the database. However, in many cases, the query image or, more generally, the target category, resides only in the mind of the user as a set of subjective visual patterns, psychological impressions, or "mental pictures." Consequently, since image databases available today are often unstructured and lack reliable semantic annotations, it is often not obvious how to initiate a search session; this is the "page zero problem." We propose a new statistical framework based on relevance feedback to locate an instance of a semantic category in an unstructured image database with no semantic annotation. A search session is initiated from a random sample of images. At each retrieval round, the user is asked to select one image from among a set of displayed images—the one that is closest in his opinion to the target class. The matching is then "mental." Performance is measured by the number of iterations necessary to display an image which satisfies the user, at which point standard techniques can be employed to display other instances. Our core contribution is a Bayesian formulation which scales to large databases. The two key components are a response model which accounts for the user's subjective perception of similarity and a display algorithm which seeks to maximize the flow of information. Experiments with real users and two databases of 20,000 and 60,000 images demonstrate the efficiency of the search process.

Index Terms—Image retrieval, relevance feedback, page zero problem, mental matching, Bayesian system, statistical learning.

1 Introduction

Our scenario is this: A person has an image concept or category "in mind." This category is essentially semantic and might be represented by various "mental pictures," by an actual object or photograph in hand or merely by subjective impressions. The person wishes to view images in a large database which match this concept. For example, the person is thinking about "old bridges" and has access to the Alinari database; see Fig. 4. The database is not semantically annotated, for example, by a rich variety of keywords. Even if it were, as pointed out in [1], there are situations in which the person may have difficulty in expressing his concept in words; he will know it when he sees it. Moreover, it may be more efficient to look over displayed images "...and make unconscious 'matches' with the one drawn by imagination..." than to rely on text or keywords that may not capture the concept. Our objective is to design a system to accommodate this "user," more specifically to get started by finding a first exemplar. This is the "page zero problem."

Scenarios like this, and the increasing demands of managing the large quantity of existing multimedia documents, have generated a growing interest in content-based retrieval techniques, both from academia and from industry [1], [2], [3], [4]. Query-by-Example (or QBE) is successfully used in many retrieval systems for ranking the elements in a database according to their similarity to a "query image" [1]. This does not solve the page-zero problem since the query image must be available. However, once an exemplar is found, QBE does allow the system to display other images that might even better match the user's concept.

Since a user's concept of similarity is largely semantic, the efficiency of the search process, whether for QBE or mental matching, is adversely affected by the infamous "semantic gap"—the discrepancy between the low-level representations of images and the high-level descriptions meaningful to users [1], [3]. Indeed, in many cases, images that present similar low-level descriptors may have very different semantic content. A partial solution is provided by relevance feedback (or RF): Divide the search session into several rounds and solicit information from the user at each step, for example by asking the user to declare which displayed images are "relevant" and which are "nonrelevant" with respect to the desired target category [5]. The system iteratively refines a model of the user's target category and uses this model to filter hopefully relevant images from the database.

Due to the page-zero problem, most proposed systems either assume that a starting image has already been identified by the user, or that the query is seeded by keywords. Some simply display randomly sampled pages from the database until the user identifies a suitable starting point. This rapidly becomes impractical for large databases. More direct solutions have been explored as well, such as database categorization [6] and query construction [7]. Other methods, initially directed toward other objectives, such as mental matching for target search [8], [9] and

- *M. Ferecatu is with the TSI Department, Institut Telecom, Telecom Paristech, 46, rue Barrault, 75634 Paris, France. E-mail: marin.ferecatu@telecom-paristech.fr.*
- *D. Geman is with the Department of Applied Mathematics and Statistics, The Johns Hopkins University, Clark Hall 302A, 3400 N. Charles Street, Baltimore, MD 21218. E-mail: geman@jhu.edu.*

Manuscript received 11 June 2008; revised 9 Oct. 2008; accepted 10 Oct. 2008; published online 17 Oct. 2008.
Recommended for acceptance by K. Murphy.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-06-0342.
Digital Object Identifier no. 10.1109/TPAMI.2008.259.

0162-8828/09/$25.00 © 2009 IEEE Published by the IEEE Computer Society

automatic semantic annotation [10], [11], could be adapted to the initialization problem. These connections will be amplified in Section 2.

Our main contribution is a new, iterative approach for discovering an instance from a semantic image category residing in the mind of the user. The search is terminated upon displaying one of these images and performance is measured by the expected number of iterations necessary to achieve this. No semantic annotation is assumed. Also, unlike previous approaches to mental matching, ours extends from target to category search and from small, structured databases to large, unstructured databases. ("Unstructured" means that the images in the database are not labeled by semantic categories.) It could serve either as a stand-alone module in a retrieval system or as a method for initializing another session, such as QBE, to obtain additional examples.

The core of our framework is a new statistical model for relevance feedback by mental matching. A binary random variable is assigned to every image in the database; the value is one if that image belongs to the target class and is zero if it does not. Taken together, these variables determine the category. The relevance feedback session starts with a random screen. At each iteration, the user is asked to choose from among the displayed images the one that is closest to his target category using whatever criteria he desires. The interface used in our experiments is shown in Fig. 12. Obviously, the target class is not displayed during a search session, ensuring that matching is entirely "mental." Even if exemplars were on display, the decisions are inevitably subjective; indeed, the challenge is to design an "answer model" which accounts for the nature of human decision making, hopefully capturing the gap between the user's "metric" and the one used by the system. Formally, the answer model is the probability distribution for the user's response conditional on the membership status of any given image.

The system maintains a separate, iteration-dependent posterior distribution for each image. Probabilities are updated based on the evidence gathered from the search, i.e., the responses of the user. The evolution of this distribution is depicted in Figs. 15 and 16 for two search sessions, one relatively efficient and one relatively inefficient. Theoretically, the optimal new display would minimize the conditional entropy on the whole family of membership variables conditional on the search history and the new response. As this is computationally intractable, we use an extension of the heuristic proposed in [9], which is shown to work very well in practice. Moreover, in order to overcome certain problems introduced by the redundancy among images with very similar low-level descriptors, we use an unsupervised categorization of the database into small clusters that are visually highly coherent. The efficiency of the search is illustrated by experiments with real users on two databases of sizes 20,000 (Alinari) and 60,000 (Corel). In both cases, fewer than five iterations are sufficient to locate an instance from a category of order 100 in 50 percent of the searches and fewer than 10 iterations in about 80 percent.

A preliminary version of this system appeared in [12]; the one presented here is considerably more mature in both practical and theoretical terms (see Section 2).

The paper is organized as follows: Related and motivating work is discussed in Section 2. In Section 3, we formulate the statistical framework for interactive search—a Bayesian relevance feedback model consisting of an update model (Section 3.1), answer model (Section 3.2), and display algorithm (Section 3.3). The low-level image descriptors and the clustering algorithm are described in Section 4, followed by an analysis of the "enabling assumption" in Section 5. The parameters are estimated in Section 6 and the whole system is evaluated in Section 7. Finally, we conclude in Section 8 with a discussion of our findings and some speculative remarks.

2 Related Work

Relevance feedback has matured into an effective method for dealing with the semantic gap in image category search. Assuming a starting image, feedback strategies exploit high-level information provided by the user in order to discover other images which represent a target class. Recent advances based on kernel methods [13], [14] avoid restrictive assumptions about the data (e.g., that classes are elliptically shaped in feature space), are flexible, and allow for efficient learning and searching even for large databases [15], [16]. Other successful approaches include active learning [15], [17], manifold learning [18], [19], graph Laplacians [20], and utilizing enhancements of the training set, for example, user logs [21] and query expansions from unlabeled images [22]; see Zhou and Huang [5] for a review and [1], [3] for connections with other machine learning and multimedia retrieval methods. The shared aspects with our work are the feedback loop and incremental learning. The key difference is that we seek an initial element of the target class starting from a random display (the page zero problem).

Perhaps the most straightforward solution to the page zero problem, at least for *target search* (singleton categories), is to ask the user to *create* the starting image: This is called "query-by-sketch" and was a part of the first image retrieval systems, for example, QBIC [23]. Similarity is then based on shape matching and evidently the results depend on the ability of the user to draw the desired query target. Recent research has focused on elastic matching of images [24], color [25], or matching the sketch to an automatically determined relevant subset of regions [26]. Following the same idea, Fauqueur et al. [7] fabricate a query example by composing image patches (regions), utilizing a visual thesaurus composed of many region categories ("sky," "building," "grass," etc.) and logical connectors.

With the page zero problem in mind, Lesaux et al. [6] create a summary of the image database from unsupervised categorization followed by a user-guided refinement of the resulting clusters. Cluster prototypes then provide a summary of the database that can be consulted to find a suitable query point.

For databases which are semantically annotated, a visual search session can, of course, be seeded by keywords provided by the user. Understandably, then, automatic image annotation has generated a lot of interest lately, even if the methods remain far from reliable and the state-of-the-art unsatisfactory. For example, Li and Wang [11] represent semantic concepts by feature-based probability distributions, allowing for models to be updated as the database grows without massive retraining. Carneiro et al. [10] model images as bags of localized feature vectors, estimating a mixture density for each image; the mixtures associated with images with shared annotations are pooled into a density estimate for the corresponding semantic class. Once images are associated with semantic concepts, by whatever method, new queries can be seeded by using natural language or keywords. Even if the annotations are not completely reliable, the user may still find a suitable starting point among the retrieved results.

In the area of category search, but assuming a starting point, Caenen and Pauwels [27] assign to each image in the database a probability that reflects its relevance to the user's intentions. The system is based on a quadratic logistic regression model used to select the next sample of images that will be presented to the user for individual annotation. There is no mental matching. The shared feature with our work is the image-specific distribution and a statistical framework.

A number of probabilistic frameworks for content-based image retrieval have been proposed in the last few years. Vasconcelos and Lippman [28] minimize the probability of retrieval error by combining feature selection and similarity measures into a Bayesian formulation. See also [29], where the same authors formulate the problem of retrieving images using Bayesian inference; the algorithm relies on belief propagation to account for both positive and negative examples of the user's preferences. Su et al. [30] suppose that the elements of the target class are generated by an underlying Gaussian density and use a Bayesian-classifier reranking of the images after each feedback step. All these results, while not directed toward solving the page zero problem, do establish that probabilistic frameworks, albeit computationally intensive, can provide state-of-the-art results in standard scenarios.

Mental matching seems to have first appeared in the seminal work of Cox et al. [8] on iterative search for a specific image in the database (target search). At every round, the user is asked to choose which of two images displayed by the search engine is "closest" to the target image residing in his mind. The formulation in [8] does not extend to *category search* because the mechanism gathering information ceases to be computationally feasible. Indeed, one cannot maintain a probability distribution on arbitrary *subsets* of images, even for small databases. Also, the answer model does not accommodate more complex user behavior associated with displaying multiple images, which is necessary to achieve reasonable search times with large databases.

Fang and Geman [9] and Ferecatu and Geman [12] extended the Bayesian framework introduced in [8]. In the context of target search, an efficient, entropy-based display algorithm was proposed in [9] and applied to mental face retrieval. We shall adapt their display mechanism to our purposes in Section 3.3. Also, unlike in [8], the answer model is explicitly designed to capture human decision making (through learned parameters). Still, the approach in [9] does not scale to large generic and heterogeneous databases, both computationally and in terms of number of feedback rounds necessary to reach the target. Indeed, the user's notion of similarity is more complex for generic images than for faces and his choices are less likely to be coherent with the feature-based metric employed by the system. Nor does the method in [9] extend to category search and unstructured databases.

The direct precursor of this work is [12], which adapts [9] to category search. First, we extend that system to handle larger databases (60,000 images), more complex semantic classes (art images, architecture, and history), and user-terminated search. Second, we provide theoretical explanations for the main algorithms; in particular, we show that the Voronoi-based display algorithm minimizes the conditional entropy of an ideal user who responds to queries based on a "reference" image in his semantic class. We also provide a quantitative analysis of the enabling assumptions by measuring the degree of semantic unity within sets of images which are close in the system metric and the degree of coherence between the answer statistics of the user and the system. Finally, we provide a behavioral interpretation for the two parameters of the answer model which leads to a highly efficient and statistically rigorous model estimation scheme in the context of two psychovisual experiments.

3 Statistical Framework

Suppose Ω denotes a database of N images, labeled $\{1, 2, \ldots, N\}$ for simplicity. The objective is to identify an image that matches the semantic and visual impressions in the mind of the user. Let $S \subset \Omega$ denote that subset of the database, i.e., the ones the user would deem as belonging to his category or *target class*. Naturally, the subset S is unknown to the system and regarded as a random set. We assume that if a member of S is displayed, the user will recognize it as an instance of the target class, terminating the search. At that point, other members of S could be retrieved by standard query-by-visual-example.

A relevance feedback session is composed of several rounds (or iterations) during each of which a different set $D \subset \Omega$ of m images is displayed. If $D \cap S \neq \emptyset$, the user identifies an element of his category; otherwise, the user chooses the image in D which he deems to be "closest" to S. Naturally, this concept of similarity will only partially cohere with the one employed by the system, which is based on low-level image features (see Section 4).

The most straightforward generalization of the Bayesian framework for target search [8], [9] would be centered on a probability distribution for S and an answer model conditional on S. This distribution would then be updated after each iteration and would drive the display algorithm. Needless to say, this is computationally impossible because, in practice, S is of order 10 to 10^2 and N is of order 10^4 to 10^5. Hence, the number of possible subsets S is far too large to support the maintenance of a probability distribution.

Instead, we associate a binary random variable Y_k with each image $k \in \Omega$: $Y_k = 1$ if $k \in S$ and $Y_k = 0$ if $k \notin S$. Of course, $S = \{k \in \Omega : Y_k = 1\}$, so S and $\{Y_k\}$ carry the same information. We maintain N parallel Bayesian systems, one for each image. Consequently, there is a response model for each k separately and, after each feedback iteration and for each k, we update the posterior distribution on Y_k given the search history. More specifically, if B_t denotes the responses of the user to the first t displays (see Section 3.1), then the distribution of Y_k given B_t is represented by the single parameter $p_t(k) = P(Y_k = 1|B_t)$. Since we assume no prior knowledge about S, we take the starting distributions $p_0(k) = 0.5$. Notice that summing the $p_t(k)$ over all k in Ω gives the expected size of S after t rounds:

$$E\big(|S|\big|B_t\big) = E\left(\sum_{k\in\Omega} \mathbf{1}_S(k)\big|B_t\right) = \sum_{k\in\Omega} p_t(k), \tag{1}$$

where $\mathbf{1}_S(\cdot)$ is the indicator function of the set S. In particular, p_t is *not* a distribution over Ω.

Our framework has three key components:

- *Update Model*: Computes $p_{t+1}(k)$ in terms of $p_t(k)$ and the user's answer at step t;
- *Answer Model*: Specifies, for each $k \in \Omega$, the probability that the user chooses an image $i \in D$ given $Y_k = 1$ (for the positive model) and given $Y_k = 0$ (for the negative model);
- *Display Model*: Determines which images to display at step t based on $\{p_t(k)\}$ and the search history.

3.1 Update Model

Let X_{D_t} denote the user's response to display D_t at time t. Notice that both D_t and X_{D_t} are random variables (see Section 3.2); indeed, X_{D_t} remains random even for D_t fixed. The first display (D_1) is randomly sampled from Ω, but the actual one chosen is important because it is involved in the determination of the posterior distributions $p_t(k)$ at subsequent times. In our scheme, D_{t+1} is determined by D_1 and the answers $X_{D_s}, s = 1, \ldots, t$, up to iteration t. It follows that the search history up to iteration t is then

$$B_t = \{D_1, X_{D_1} = i_1, \ldots, X_{D_t} = i_t\}, \tag{2}$$

where i_s is the image selected by the user in response to D_s. For simplicity, we will suppress D_1, it being understood that all conditional probabilities are so conditioned.

The basic statistical assumption we need is that

$$P(X_{D_{t+1}} = i|Y_k = 1, B_t) = P(X_D = i|Y_k = 1, D_{t+1} = D).$$

That is, given $Y_k = 1$, the distribution of the answer at time $t+1$ only depends on the history B_t as represented by the display D_{t+1}, which, as stated above, is determined by B_t. Put differently, the display is a "sufficient statistic." We also assume that

$$P(X_D = i|Y_k = 1, D_{t+1} = D) = p_+(i|k, D)$$

the "positive answer model." That is, the answer probabilities are time-independent. Similarly, for conditioning on $Y_k = 0$ and the negative answer model,

$$P(X_D = i|Y_k = 0, D_{t+1} = D) = p_-(i|k, D).$$

Notice that we are not assuming (as in some earlier work) that the answers X_{D_s}, $s = 1, 2, \ldots$, are conditionally independent given S (and D_1). This assumption is unreasonable. For example, since D_2 is determined by i_1 (and D_1), the joint distribution $P(X_{D_1} = i_1, X_{D_2} = i_2|S)$ becomes $P(X_{D_1} = i_1, X_{D_2(i_1)} = i_2|S)$, which factors into $P(X_{D_1} = i_1|S)P(X_{D_2(i_1)} = i_2|S)$. But the second factor is not the same as $P(X_{D_2} = i_2|S)$. Indeed, we do not expect that the second answer follows the same distribution knowing the display as not knowing the display.

Updating each $p_t(k)$ depends on *both* the positive and negative response models. From (2), we have $B_{t+1} = B_t \cap \{X_{D_{t+1}} = i\}$ and, since $X_{D_{t+1}}$ is independent of B_t given Y_k and D_{t+1}, we have

$$\begin{aligned} p_{t+1}(k) &= P(Y_k = 1|B_{t+1}) \\ &= P(X_D = i|Y_k = 1, D_{t+1} = D)p_t(k)/C_{t+1} \\ &= p_+(i|k, D)p_t(k)/C_{t+1}, \end{aligned}$$

where the normalizing constant is:

$$C_{t+1} = p_+(i|k, D)p_t(k) + p_-(i|k, D)(1 - p_t(k)).$$

3.2 Answer Model

Let $D = D_t = \{i_1, \ldots, i_m\} \subset \Omega$ be the set of images displayed at iteration t. We can assume that no element of S appears in D since otherwise the search terminates. Consequently, the response X_D assumes values in D itself: $X_D = i$ signifies that image i is the closest image to S *in the opinion of the user*.

Let d denote the metric in the features space; the image descriptors are discussed in Section 4. Our answer models are of the form:

$$p_+(i|k, D) = \frac{\phi_+(d(i,k))}{\sum_{j\in D} \phi_+(d(j,k))}, \tag{3}$$

$$p_-(i|k, D) = \frac{\phi_-(d(i,k))}{\sum_{j\in D} \phi_-(d(j,k))}. \tag{4}$$

The design issue is complex as it involves human psychology and decision making. Naturally, the efficiency of the model will also be affected by the extent to which the system metric captures semantic similarity.

The overall design of the functions ϕ_+ and ϕ_- is motivated by the intuitive expectation that, *generally speaking*, the perceived similarity between two images will be roughly inversely proportional to their distance apart in the metric d. Therefore, we take $\phi_+(d)$ to be monotonically decreasing in d and $\phi_-(d)$ to be monotonically increasing in d. As a result, if $k \in S$, the closer the image $i \in D$ is to k in the stored metric, the more likely the user is to choose it in the positive model. That is, if $i, j \in D$ and $d(i,k) < d(j,k)$, then we expect $p_+(i|k, D) > p_+(j|k, D)$. Similarly for the negative model with the inequality on probabilities reversed since we are assuming $k \notin S$.

We adopt parametric forms for ϕ_+ and ϕ_- (see Fig. 1) and learn the parameters from real data collected from users (see Section 6). Parameter estimation is based on characterizing the parameters in psychovisual terms. The parameter θ_1 can be viewed as a "saturation" threshold: For the positive model (respectively, negative model), an image θ_1 units away from a target is no more likely (respectively, less likely) to be chosen than one still farther away. The

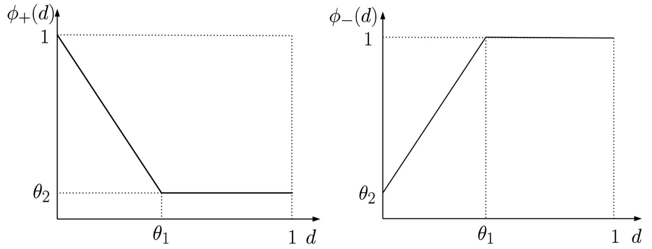


Fig. 1. Parametric forms for ϕ_+ and ϕ_-.

parameter θ_2 controls the degree of coherence between the subjective decisions and the system metric. Take, for example, the positive model and suppose one displayed image i is very close to k and all the other $m-1$ images are farther than θ_1 units from k. In other words, there is one overwhelmingly best choice in terms of d. Then, according to (3),

$$p_+(i|k,D) \cong (1+(m-1)\theta_2)^{-1}. \tag{5}$$

Small values of θ_2 would then embed high coherence. We shall return to this issue in Section 6.

3.3 Display Model

Perhaps the simplest procedure for choosing D_{t+1} would be to select the m images most likely to belong to S, as measured by their masses under $p_t(k)$. Unfortunately, this elementary strategy is far less effective (in terms of average search time) than others due to the fact that it does not adequately "sample" the database. For one thing, it does not take into account visual similarity; for instance, two very similar images, both with high masses, are probably either both in S or both not in S. In addition, from an information-theoretic viewpoint (and confirming what we already suspect), this is not an efficient way to gather information. Instead, we borrow the line of reasoning in [9], but adapted to category search, and seek a more powerful strategy. First, we establish an interpretation of the normalized $p_t(k)$ distribution and then use it to derive a more appropriate sampling of D from Ω.

Imagine an "ideal user" who picks at random an image i from his category S and whose selections from D are based entirely on this image alone and the actual system metric. Specifically, presented with D, this ideal user chooses the image $j \in D$ that is closest to i using d. We will compute the optimal display for learning the reference image of this user.

Since S is random, and since i is picked randomly from S, the reference image is a random variable Z with values in Ω. For every image $k \in \Omega$, at stage t, we first calculate $P(Z=k|B_t)$. Since $Z=k$ implies $k \in S$:

$$\begin{aligned} P(Z=k|B_t) &= P(Z=k, k\in S|B_t) \\ &= P(Z=k|k\in S, B_t)P(k\in S|B_t) \\ &= E\big(1/|S|\big|k\in S, B_t\big)p_t(k). \end{aligned}$$

Now, make the assumption that $E\big(1/|S|\big|k\in S, B_t\big) \equiv C_t$ independently of k. Summing the above equation over k yields

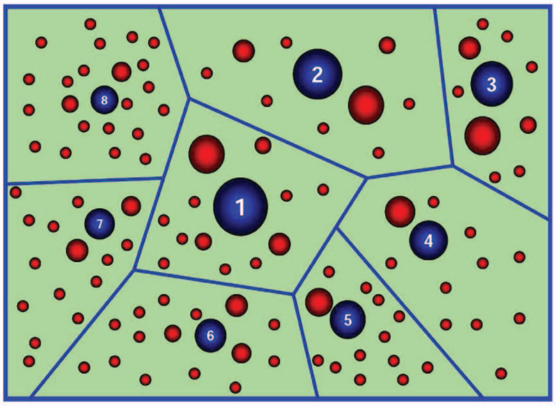


Fig. 2. A Voronoi partition of the image database with eight cells. The size of the images is proportional to their mass. If all the cells have equal mass, the centers represent an optimal display D for an ideal user.

$$1 = \sum_k P(Z=k|B_t) = C_t \sum_k p_t(k),$$

from which it follows that *the normalized probabilities* $p_t(k)/\sum_j p_t(j)$ *represent the distribution of* Z *at time* t, i.e., the normalized $p_t(k)$ is the probability that image k is a given element randomly extracted from S. From (1), we also know that the normalizing constant represents $1/E(|S|)$.

Returning to the choice of the display D, for this ideal user, we attempt to minimize the uncertainty about Z given the search history and the new evidence provided by $X_{D_{t+1}}$:

$$D_{t+1} = \arg\min_{D\subset\Omega} H(Z|B_t, X_D). \tag{6}$$

This combinatorial optimization problem is evidently intractable because it involves looping over all subsets of Ω. But, an equivalent reformulation leads to a practical algorithm.

3.3.1 Reformulation of (6)

Using elementary properties of conditional entropy,

$$D_{t+1} = \arg\min_{D\subset\Omega}(H(X_D|Z,B_t) - H(X_D|B_t)). \tag{7}$$

However, the response X_D of this ideal user is a function of Z and, hence, $H(X_D|Z,B_t)=0$. As a result, the optimal display is the one for which $H(X_D|B_t)$ is maximized. Since entropy is maximized at the uniform distribution, we seek m images, again call them $\{i_1,\dots,i_m\}$, such that $P(X_D=i_l|B_t)\approx\frac{1}{m}$. *In summary, in order to solve (6) for our ideal user, we want the Voronoi partition based on D and on the metric d to have cells of equal mass under the normalized* $p_t(k)$ *distribution over* Ω. This situation is depicted in Fig. 2 for the case $m=|D|=8$; the disks represent images in the database and the size of the disks is proportional to their mass under $p_t(k)$. The centers are the images in the optimal D. All of the images in each cell are closer to the center of the cell than to any other center; hence, knowing only the search history B_t, the answer of our ideal user is uniformly distributed.

A natural, sequential procedure for constructing a display D which yields approximately equally likely answers relative to a distribution over Ω was described in [9]. Hence, we normalize the distribution $p_t(k)$ over Ω and use the algorithm described there in order to compute the images $i_1, ..., i_m$ sequentially. We refer the reader to [9] for the details. Roughly speaking, the center of the first cell is the image with the highest mass; the cell is initially constructed by adding images according to their distance to the center until mass $1/m$ is reached. Then, the next largest mass seeds the next cell, and so forth. There is also a feedback loop which adjusts the cells after each iteration.

3.3.2 Acceleration by Clustering

Although fast, easy to implement, and highly effective for target search, the heuristic solution lacks efficiency for category search with large databases in which many images are visually very similar. In fact, many semantic categories can be very roughly decomposed into a union of clusters of highly similar images. For example, "red flowers" likely have very similar low-level descriptors. Applying the heuristic described here at the image level can then result in search sessions in which the probability mass gets highly concentrated on images in the complement of S at the beginning of the search session.

For this reason, and in order to lower the memory requirements of the algorithm, we reduce this redundancy by unsupervised clustering of the image database into small but highly coherent cells. Let $\mathcal{C} = \{C_l\}_{l=1}^{P}$ be a partition of Ω. For each cluster $C \in \mathcal{C}$, we compute the expected size of $C \cap S$ given the session history, namely, $\eta_t(C) = \sum_{k \in C} p_t(k)$, and then normalize these to a probability distribution $p_t(C)$ over $\mathcal{C}$. We then compute the next display screen D_{t+1} just as previously described, *but at the cluster level*, i.e., feeding the algorithm with the list of clusters $\mathcal{C}$ and the corresponding probabilities $\{p_t(C) : C \in \mathcal{C}\}$ in place of $\{p_t(k)\}$. The distance between two clusters is the average link distance:

$$d(C_l, C_p) = \frac{1}{|C_l||C_p|} \sum_{i \in C_l} \sum_{j \in C_p} d(i, j).$$

The output of the algorithm is then a list of clusters $\mathcal{D} \subset \mathcal{C}$. For each element $C \in \mathcal{D}$, we choose the image that has the highest posterior $p_t(k)$ for $k \in C$ to be displayed.

After the user has chosen an image in $i \in D$ (suppose $i \notin S$, otherwise the search session is over), the cluster C containing i is discarded from the list of clusters. Of course, if the cluster contains elements of S, these are also discarded. However, in our case, this is not an issue. Suppose the elements of C were independently sampled without replacement from a uniform distribution over Ω; then, the probability that an element of S belongs to C is $P(S \cap C \neq \emptyset) \approx |C| \cdot |S|/|\Omega|$, which is very small for large databases and small clusters. In our case (see Section 4.1), $|\Omega| = 60{,}000$, $|C| = 8$, and $|S| \approx 100$, thus $P(S \cap C \neq \emptyset) \approx 1.3 \times 10^{-2}$. This value is small enough for our purpose: The event appears once in an average number of 75 feedback iterations (the average search session length was less than 10 iterations in our experiments). Moreover, the elements of the clusters are not randomly sampled from Ω: They are highly coherent (close to each other in the description space). Since the element i chosen by the user is not in S, then the $P(S \cap C \neq \emptyset)$ should be even smaller compared to the baseline (random sampling).

We describe the clustering algorithm in Section 4.3. This procedure produced excellent results in practice and allowed the average relevance feedback session length to drop under 10 iterations on our test databases (see Section 7).

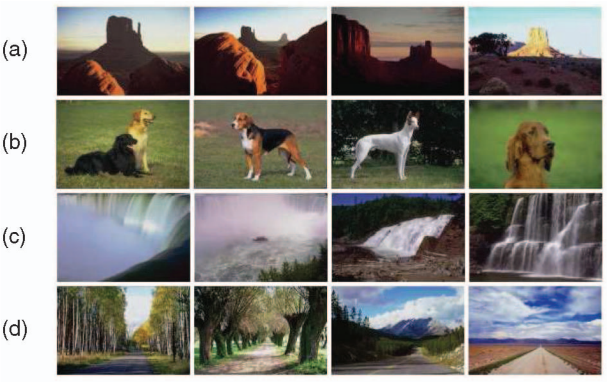


Fig. 3. Samples from four semantic ground truth classes from the Corel database: "Monument Valley," "Pedigree Dogs," "Waterfalls," and "Roads and Highways" (top to bottom).

4 Visual Content Description

4.1 Image Databases and Ground Truth

To test our framework, we use two image databases: the well-known Corel stock photodatabase and a database of art images kindly provided by Alinari.[1] Both databases are indexed by keywords which allowed us to select several ground truth classes to use in the tests, as described below.

The Corel database contains 60,000 natural images covering a broad range of semantic themes: agriculture, architecture, cities, closeups, cuisine, landscapes, museum, space, sports, textures, etc. For the tests, we selected 10 semantically coherent image classes as target categories: "Beverages," "Fruits," "Festive Food," "Models," "Monument Valley," "Office Interiors," "Pedigree Dogs," "Roads and Highways," "Space Scenes," and "Waterfalls" (see Fig. 3).

The Alinari database consists of 20,000 art images (paintings, sculpture, architecture, archeology, etc.), half of them being gray-level images and the other half color images. Although smaller, this database is more difficult than Corel; semantic concepts can be illustrated by different types of images. For example, images that match the concept "horse" include paintings, photos, statues and frescoes, all of them very different visually (see Fig. 4). We manually selected ten semantic target classes: "Portraits (paintings)", "Portal (architecture)", "Madonna and Child (paintings)", "Bridge (architecture)", "Tower (architecture)", "Bones (archeology)" "Horseback riding (mixed: paintings, sculpture, high-relief)", "Medieval Castle (architecture)", "Still Life (paintings)" and "Cupola (architecture)".

In choosing the target classes for our experiments, we tried to cover a reasonably large range of situations, including both difficult cases (cluttered, natural scenes) and more "standard" ones (objects on a relatively uniform background, human artifacts, etc.), while maintaining a feasible number of classes for experiments with real users. Also, we ensured that the interpretation is unambiguous, for example, images from one class would typically not be wrongfully attributed by users to another class. Approximately 100 images match the

[1] http://www.alinari.com.

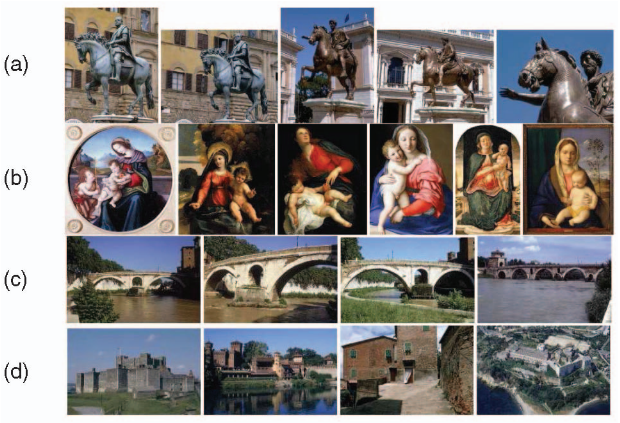


Fig. 4. Samples from four classes (Alinari database): "Horse and Rider," "Madonna and Child," "Old Bridges," and "Medieval Castle" (top to bottom).

concept for each target class, but, of course, the actual number is likely to be larger because we could not visually inspect all the images. For this reason, we let the user terminate the search session by identifying an element of the class.

A search session begins by showing the user a summary of the target class. However, during a search session, the user cannot consult the target class, ensuring that he will match displayed images with only mental pictures.

4.2 Image Descriptors

Finding good image descriptors that accurately describe the visual content of many different classes of images is a challenging task. Such descriptors are easier to compute for specialized databases (e.g., medical images, fingerprints, remotely sensed images), where prior knowledge can be used to devise dedicated mathematical models of the image content. For generic images, most representations balance different components of the image content, usually color, texture, and shape [1], [2], [4].

Local descriptors (e.g., points of interest or image regions) have been successfully used in several object detection tasks [31]. However, they are less adapted to detect semantic concepts that cannot be directly associated with individual rigid objects, such as emotional states and aesthetic impressions. Moreover, while such descriptors are largely stable and invariant to common geometric and photometric image transformations, they are resource intensive in terms of memory and computation and, consequently, not well adapted to large-scale image retrieval systems that require answers in real time.

Instead, we use a combination of global image descriptors, specifically color, texture, and shape, for the following reasons:

- *Small memory:* The descriptors for the Corel database (60,000 images) can be stored in the main memory of an ordinary PC.
- *High speed:* No special data structures are necessary and the distance function we use (L_1) is easy to compute.
- *Generality:* Our system is designed for unstructured, generic databases and with no restrictions on the target class. Whereas local descriptors may be fine-tuned to perform well for a given class of objects, there is (as yet) no universal detector that can be trained for any object from only a few examples (identified by relevance feedback). In contrast, global descriptors have been shown to perform well in this context, for example, with SVM-based relevance feedback (see [5] for a review).

In the rest of this section, we briefly describe the image descriptors we employ and dimensionality reduction. A more in-depth description can be found in [32].

4.2.1 Global Descriptors

Color histograms provide a description of the color content of an image, but ignore spatial information. We use weighted histograms [33], where the contribution of each pixel is proportional to its importance in the local context. As weighting functions, we use the Laplacian $\|\Delta(x,y)\|^2$ at the pixel (x,y) to emphasize corners and edges, and the probability of the color of the current pixel in a local window, with a small value signaling importance.

To describe the shape content of an image, we use a histogram based on the Hough transform, which captures the behavior along straight lines of varying directions. First, the direction of the gradient is found for every pixel. Then, a joint histogram is constructed for the angle of the gradient and the length of the projection of a reference point (the upper-left corner of the image) along the local tangent line going through the current pixel.

Finally, texture feature vectors are based on the Fourier transform—the distribution of spectral power density along different frequencies and along various angles [34].

4.2.2 Dimensionality Reduction

The joint feature vector has more than 600 dimensions, which can make relevance feedback impractical for large databases. We use linear Principal Component Analysis [35] and keep 95 percent of the variance of the data, corresponding to the highest eigenvalues of the covariance matrix. This procedure reduces the number of dimensions about fivefold, while remaining within a 5 percent overall loss on the precision-recall diagrams built using the ground truth classes presented above.

Of course, if the relevant image classes were known a priori, other methods, such as discriminant analysis, might be more appropriate. Also, we expected kernel PCA [14] to better focus on relevant nonlinear "dimensions"; this should indeed be the case when the manifold spanned by the images is low-dimensional and highly nonlinear. However, KPCA and linear PCA yielded similar precision-recall diagrams in our case and we decided to keep the linear PCA because it is easier to compute and does not require kernel parameter tuning.

4.3 Clustering

Recall that our algorithm for computing the optimal display is accelerated by clustering the database. Since the database is generic and since no prior information about semantic content is available, smaller clusters are expected to be more coherent than larger ones. Needless to say, the elements of even a small cluster may belong to different semantic

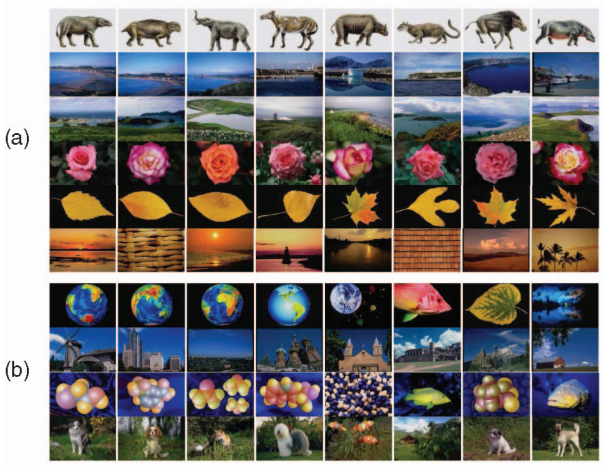


Fig. 5. Sample clusters of size 8: Some are very coherent semantically (top rows) whereas others are less so (bottom rows).

classes. However, this is not a problem since we maintain a list of probabilities $p_t(k)$ at the image level.

We tried several classical clustering algorithms, such as K-Means, Fuzzy K-Means [36], and Competitive Agglomeration [37]. However, the results were inadequate for our purposes because some quite large clusters (over 100 images) were generated with highly diverse visual and semantic structure.

To satisfy our requirements, we modified Quality Threshold clustering [38], which provides control over the size of the clusters and is independent of initialization. Briefly, given a desired cluster size R, the algorithm iteratively chooses new clusters from a list of candidates based on computing the R nearest neighbors to each unclustered image. The candidate with the smallest diameter (enveloping sphere) is chosen (see Algorithm 1). Running time is no issue since the computation is off-line. In Fig. 5, we show some example clusters of size $R = 8$. Most clusters are visually consistent in terms of our image descriptors based on color, texture, and shape. Semantic diversity is tolerated since we only use the clusters to simplify the display algorithm. Of course, the more homogeneous semantically the better, the ideal being that every semantic category be a perfect union of clusters.

```
Algorithm 1 Fixed size QT clustering
Require: $\Omega$: image database, $R$: cluster size
Ensure: $\mathcal{C}$: clusters set
  $\mathcal{C} \leftarrow \emptyset$
  while $\Omega \neq \emptyset$ do
    $\varepsilon_{\min} = \infty$
    for all $i \in \Omega$ do
      $A \leftarrow$ set of $R$-nearest elements to $i$
      $\varepsilon \leftarrow$ diameter$(A)$
      if $\varepsilon < \varepsilon_{\min}$ then
        $\varepsilon_{\min} = \varepsilon$
        $L = A$
      end if
    end for
    $\Omega = \Omega \setminus L$
    $\mathcal{C} = \mathcal{C} \cup \{L\}$
  end while
```

5 Enabling Assumption

The choice of image descriptors obviously has a direct impact on the overall efficiency of the system. Indeed, performance critically depends on the extent to which "closeness" in the system metric (the L_1 distance between two feature vectors) coheres with "closeness" in the objective sense of semantic identity as well as "closeness" in the subjective sense of the user. Having a significant degree of coherence is the central "enabling assumption."

As described in Section 4.2, we use global image descriptors, mainly because they have a small memory impact and scale well to large image repositories. In this section, we report two experiments for quantitatively measuring the extent to which our enabling assumptions are satisfied.

First, we measure the extent to which the system metric discriminates between a semantic class S and a random sample of size $|S|$ from the database. Second, we estimate the probability that the user chooses the lth closest image in the system metric.

5.1.1 Experiment One: Quantifying the Semantic Gap

Suppose we fix a semantic class S (chosen from the ground truth classes). Intuitively, we hope that elements of S will be much "closer" to each other in system metric compared with those in a set of size $|S|$ sampled randomly from Ω. For each element $k \in S$, define:

$$z_{kl}(S) = \begin{cases} 1 & \text{if } l\text{th closest image to } k \text{ in } \Omega \text{ is also in } S \\ 0 & \text{otherwise} \end{cases}$$

and let

$$z_l(S) = \frac{1}{|S|} \sum_{k \in S} z_{kl}.$$

Of course, $z_l(S)$ is the estimated probability that, for an image k chosen at random from S, the lth closest image to k is also in S.

The ideal case is when there is a perfect match between the system's metric and the semantic class S: For each element $k \in S$, all $|S| - 1$ nearest neighbors of k belong to S:

$$z_l(S) = \begin{cases} 1 & \text{for } l = 1, \dots, |S| - 1 \\ 0 & \text{for } l = |S|, \dots, |\Omega|. \end{cases}$$

The baseline hypothesis is that there is no connection between the system metric and the structure of S. That is, the behavior of this statistic is the same if S is randomly sampled from Ω, in which case $z_l(S) \approx |S|/|\Omega|$.

In Figs. 6 and 7 we present the results obtained for our two test databases (Corel and Alinari) and $l = 1, \dots, 8$ (the size of the display). For each semantic class S from the ground truth we compute $z_l(S)$ as described above and then we average over all classes. The baseline is $z_l \approx 0.0017$ for Corel ($|S| \approx 100$, $|\Omega| = 60{,}000$) and $z_l = 0.005$ for Alinari ($|S| \approx 100$, $|\Omega| = 20{,}000$). We see that the values of z_l, while far from unity, do provide between one and two orders of magnitude improvement over the baseline case. As we shall see in Section 7, this is enough to allow very reasonable search times for relevance feedback.

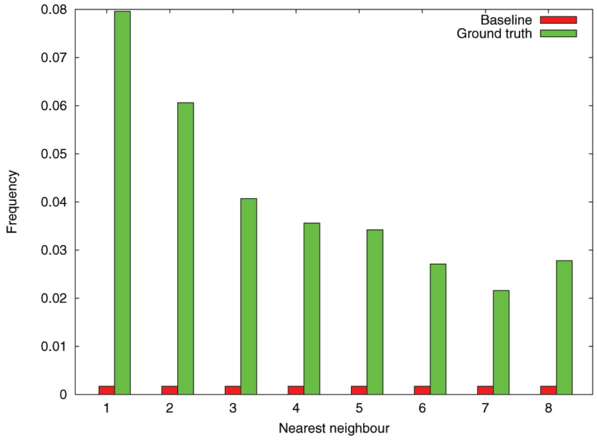


Fig. 6. Corel database: The probability that for an image k chosen at random from a ground truth class S, the lth closest image to k also belongs to S.

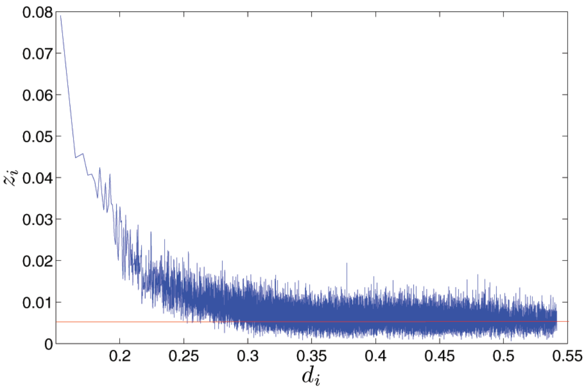


Fig. 8. Alinari database: plot of the pairs $\{d_l, z_l\}, l = 1, 2, ...$, where d_l is the average distance from an image k to to its lth nearest neighbor and z_l is the estimated probability that the lth nearest neighbor to an image k is in same semantic ground truth class. The horizontal line is the baseline case.

Let d_l, $l = 1, 2, \dots, N-1$, be the expected value of the distance from k to the lth closest image to k, estimated over all images k in the union of the ground truth classes. Fig. 8 is a plot of z_l versus d_l. The shape of curve is explained by the fact that $\{d_l\}$ is an increasing sequence whereas $\{z_l\}$ is roughly decreasing. As l grows large, we would expect that $z_l \rightarrow |S|/|\Omega|$, which indeed happens. In fact, the limiting value is approximately reached as soon as d_l reaches the interval $[0.30, 0.35]$. In other words, for images this far or farther from the target class representative k, the system metric acts no better than random sampling. This value of d agrees very well with the value we estimate in Section 6 for the parameter θ_1 for the positive answer model ($\hat{\theta}_1 = 0.35$). Recall that this parameter represents a "saturation" threshold: The user is assumed to have no preference among displayed images lying more than θ_1 units away from $k \in S$ in system metric.

5.1.2 Experiment Two: Quantifying System-User Synchronization

Let $\{(i_n, D_n, S_n)\}_{n=1,\dots,M}$ be the outcomes of M feedback interactions collected from the users during various search sessions, where i_n denotes the user's response to display D_n for target class S_n. Let $p_l^{(u)}$ be the probability that the users select the lth closest image to the target relative to the system metric, $l = 1, \dots, |D|$, estimated from these data. The baseline is the "random user" who selects an image from D at random; for $|D| = 8$, the baseline probabilities are $1/8 = 0.125$.

We collected data points from 12 users, with $M = 1{,}616$ for the Corel database and $M = 1{,}124$ for the Alinari database, using $|D| = 8$. The values of $p_l^{(u)}, l = 1, \dots, 8$, are given in Figs. 9 and 10.

In neither case is the metric induced by the image descriptors highly consistent with mental matching by real users. For example, the probability the user selects the closest image to the target class is 0.27 for the Corel database and only roughly 0.19 for the Alinari database. Nevertheless, the departure from the uniform distribution is sufficiently large to convey enough information to yield very reasonable search times (see Section 7). Also, these results substantiate our impression that, although smaller, the Alinari database is more "difficult" in the sense that users' choices are visibly less coherent with the system metric.

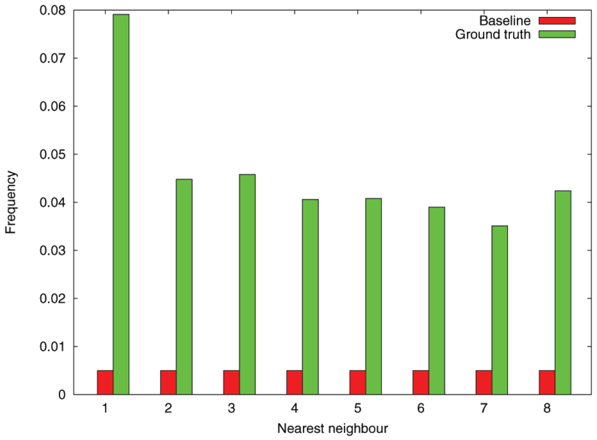


Fig. 7. Alinari database: The probability that for an image k chosen at random from a ground truth class S, the lth closest image to k also belongs to S.

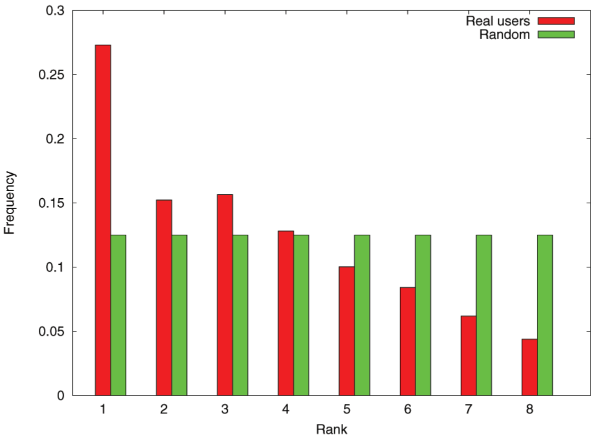


Fig. 9. Corel database: The estimated probability that a user selects the lth closest image to the target class among eight displayed images.

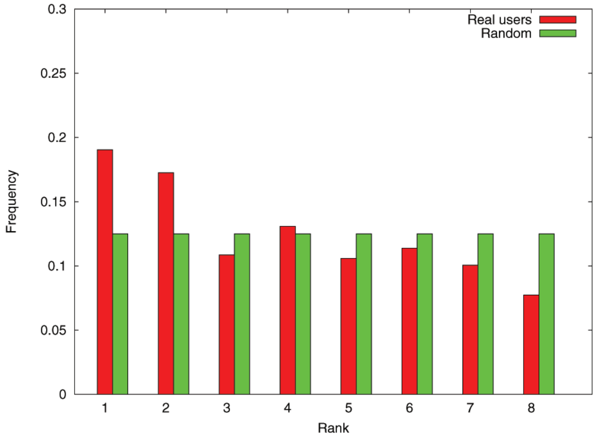


Fig. 10. Alinari database: The estimated probability that a user selects the lth closest image to the target class among eight displayed images.

6 Parameter Estimation

Recall that our answer models, $p_+(i|k,D) = P(X_D = i|Y_k = 1)$ and $p_-(i|k,D) = P(X_D = i|Y_k = 0)$, depend on the parametric functions ϕ_+ and ϕ_- (see Section 3.2). The parameters θ_1 and θ_2 are chosen to minimize the difference between how similarity is perceived by the users and by the system metric. The meaning of the parameters in psychovisual terms was explained in Section 3.2; basically, θ_1 is a "saturation" parameter and θ_2 controls the coherence between subjective decisions and the system metric.

6.1 Estimation of θ_1^+ (Positive Model)

Estimation of θ_1^+ is based on a statistical hypothesis test. Fix $\theta \in \{0.05, 0.1, \dots, 1\}$, the possible values of θ_1^+, a ground truth class S, a member $k \in S$, and select two images $i, j \notin S$ such that $d(i,k) \approx \theta$ and $d(j,k)$ is chosen uniformly in $[\theta, 1]$. Now, display a summary of the target class S to a user, as well as the two images i, j, and ask the user to choose which one, i or j, is closer to the target class S in his opinion. Consider the following two hypotheses:

- $\mathbf{H_0}$: *No Preference:* The two displayed images, i and j, are equally close to the target in user's opinion;
- $\mathbf{H_1}$: *Preference for the Closer:* The user has a preference for image i, the one closer to the target exemplar.

Basically, we want to choose the largest value of θ_1^+ for which the null hypothesis is rejected at the 0.05 significance level.

To determine p-values for the various values of θ, for each possible value of θ, we repeat the experiment 20 times for each of 12 users, each time with a different S and choice of i, j, k, yielding a sample of $n = 240$ user choices. Let $N(\theta)$ be the number of times that the users choose image i. Under H_0 (and assuming independent selections), $N(\theta)$ has a binomial distribution with parameters $n = 240$ and $p = 1/2$. Setting $p(\theta) = P(\text{Bin}(n, 1/2) \geq N(\theta))$ and appealing to the central limit theorem,

$$p(\theta) \approx 1 - \Phi\left(\frac{N(\theta) - \frac{n}{2}}{\sqrt{n}/2}\right),$$

TABLE 1
Estimation of θ_1^+, the Saturation Parameter

θ	0.20	0.25	0.30	0.35	0.40	0.45
$N(\theta)$	181	143	136	133	129	123
p	≈ 0	0.0015	0.0194	0.0466	0.1226	0.3493

We choose $\theta_1^+ = 0.35$ because this is the largest value for which a "no preference" hypothesis is rejected at significance level 0.05.

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. The results are presented in Table 1, which gives $\hat{\theta}_1^+ = 0.35$, the largest value with $p < 0.05$.

6.2 Estimation of θ_2^+ (Positive Model)

Recall that (5) gives the model probability, given $k \in S$, that a user chooses a displayed image i which is extremely close to $k \in S$ if all the other $m-1$ displayed images are at least θ_1^+ units from k. Since $P(X_D \neq i|Y_k = 1) = 1 - p_+(i|k,D)$, it follows that

$$\theta_2^+ \cong \frac{1}{m-1} \frac{1 - p_+(i|k,D)}{p_+(i|k,D)}. \tag{8}$$

Consequently, in order to estimate θ_2^+, we collect data as follows:

1. Randomly choose a target class S from the ground truth and an image $k \in S$.
2. Construct a display D for which there is an image $i \notin S$ with $d(i,k) \approx 0$ and the other $m-1$ images are at least θ_1^+ units away from k in the system's metric.
3. Display D and a summary of S and ask the user to select the image that in his opinion is closest to S.
4. Record user's decision: $X_D = i$ or $X_D \neq i$.
5. Repeat these steps p times for each user.

For each of 14 users and with $|D| = 8$, we performed the above experiment 40 times collecting 560 data choices. Of these, 385 corresponded to $X_D = i$ and 175 corresponded to $X_D \neq i$. From (8), we obtain $\hat{\theta}_2^+ = 0.065$. For the situation of a unique match (i.e., one displayed image very close to k and all others "far" away), the estimated probability of selecting the good match is therefore $1/(1 + 7 \cdot 0.065) \approx 0.69$.

The estimates for both parameters compare reasonably well with those in [12] based on a subset of 20,000 images from Corel using straightforward maximum likelihood. This is not altogether surprising for θ_2^+ in view of the invariance principle for maximum likelihood estimates. We prefer the method here; it provides more insight into the answer model because the parameters are estimated in the context of their psychological interpretation.

6.3 The Negative Model

We tried to estimate the parameters for the negative answer model in the same fashion as for the positive model. For example, θ_1^- would be estimated following Section 6.1, but selecting $k \notin S$ and H_1 to be the hypothesis "preference for the more distant image j." However, the data we obtained from the users produced estimates rather sensitive to S and k. Indeed, a user will likely correctly prefer an image that is close to an element of S to one farther away, but the "inverse" is not necessarily true. Indeed, an image "far"

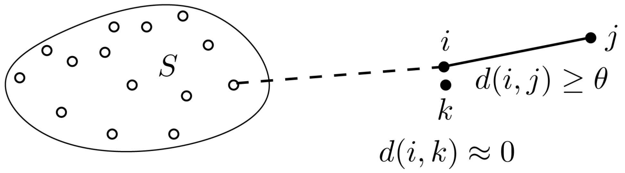


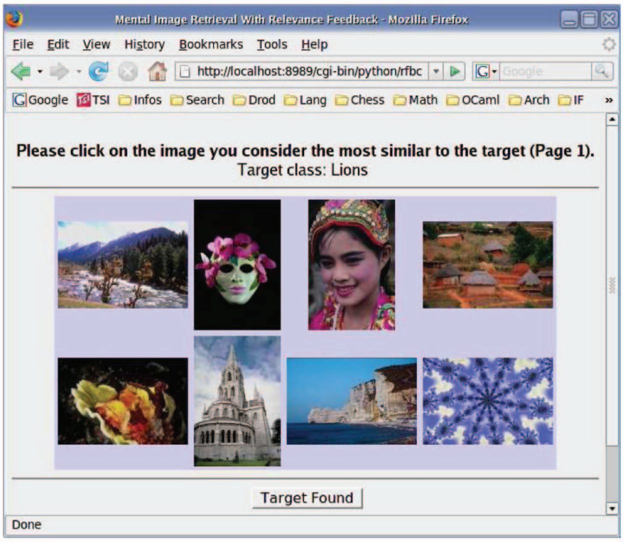
Fig. 11. Analysis of the negative answer model: Knowing only $k \notin S$, an image j "far" away from k is not necessarily preferable to a user than an image i close to k

from $k \notin S$ may be nowhere near S in either the system metric or the user's mind and no more suitable than an image near k. See Fig. 11.

Such observations suggested that perhaps the negative model had a limited impact on the performance of the system, certainly less than we initially thought. To confirm this, we performed several tests measuring the number of feedback iterations until the user identifies a target, comparing several values for θ_1^- and θ_2^- with the uniform negative model (i.e., $\theta_1^- = 0$ and $\theta_2^- = 1$). The results were very similar and we chose $p_-(i|k, D) \equiv \frac{1}{m}$.

7 Performance Evaluation

In order to estimate the distribution of the search time, we collected data from a group of 12 individuals not familiar with the system. For each individual and each ground truth class, the user was first presented with a visual summary of the class. Once the user considers that he has a good grasp of the target class, the feedback session starts and the user can no longer consult the class summary, assuring that matching and decision making are purely from memory. A relevance feedback session starts with a random display. A session ends when an element of the target class is identified by the user. Every (nonterminal) click provides a "data item" in the sense of a triple (S, D, i) corresponding to a target class, set of displayed images, and user's response. We set $m = |D| = 8$; displaying many fewer or many more images has adverse consequences with real users. The experimental interface is shown in Fig. 12.


Mental Image Retrieval With Relevance Feedback - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:8989/cgi-bin/python/rfbc
Google
Google TSI Infos Search Drod Lang Chess Math OCaml Arch IF
Target Found
Done

Please click on the image you consider the most similar to the target (Page 1).
Target class: Lions

Fig. 12. The interface used for experiments.

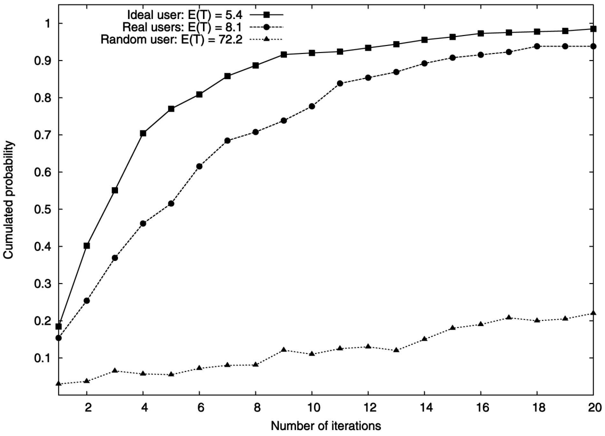


Fig. 13. Corel database: Distribution of the search time for real, ideal, and random users.

We measure the performance of the system by the number of iterations T required to locate an instance from the target class. We estimate $P(T \leq t)$, the cumulative distribution of T, from the data collected over M search sessions. Evidently, the faster $P(T \leq t)$ grows, the more efficiently the system is operating.

In addition to real users, we also present the results of two simulations under the same experimental settings (same ground truth classes, etc.) representing two extreme cases: the "ideal user" and the "random user." The ideal user always chooses the image closest to the target class in the system metric (using the average distance between an image and a set). Notice that this "ideal user" is not exactly the same one considered in Section 3.3, who matches to a randomly selected exemplar from his class rather than to the whole class. Matching to the entire class is clearly more efficient; indeed, this is the optimal performance we can hope to attain. The other extreme is a random response—the user selects one of the eight displayed images at random. The results are presented in Figs. 13 and 14. Obviously, the proposed model far out-performs a random response. More importantly, the absolute performance is

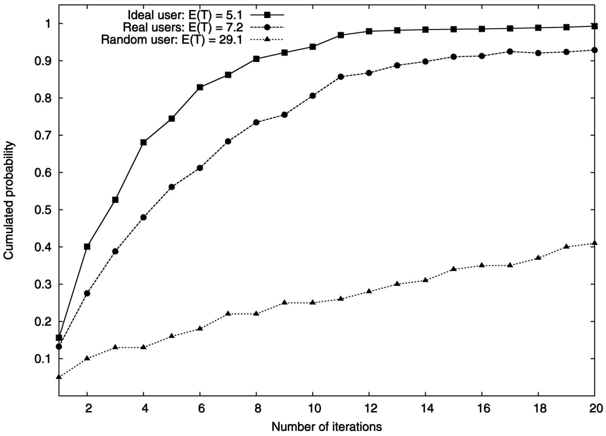


Fig. 14. Alinari database: Distribution of the search time for real, ideal, and random users.

quite reasonable, with a mean search time $E(T) \approx 8$ for the Corel database and $E(T) \approx 7$ for the Alinari database. In the latter case, an instance of the target is discovered in fewer than four iterations in approximately one-half of the searches and in fewer than 10 iterations in more than 80 percent of the searches. The results are similar for Corel. It should be emphasized that, after setting the two parameters for the positive model as described in the preceding section, *the system is fully determined*. In particular, there are no other parameters to tune.

Returning to the page zero problem, the baseline case is a random display of images, without replacement, until a member of the target class appears. Computing the average number of screens necessary is then relatively straightforward. Let N, L, m be the sizes of the database, target class and display, respectively. Imagine the N images are laid out in a row at random. Some of these slots are filled by the L images from the target class. The expected value of the position of the first such example is approximately $N/(L+1)$. Since random displays corresponds to exploring the images from left to right in groups of m, we obtain $E(T) \cong N/m(L+1)$.

In our experiments for the Corel and Alinari databases, $L = 100$, $m = 8$; hence the average is around 75 iterations for the Corel database and 25 iterations for the Alinari database. Accounting for the fact that we eliminate at each iteration the cluster containing the user's selection would lower this average, but not nearly by half since the clusters are so visually coherent, which works against rapid discovery. Indeed, a displayed cluster is not a random subset from the database and is not independent of the preceding display. In fact, for a cluster size of eight (Fig. 13), the mean search of the random user for the Corel database is $E(T) = 72.2$, barely smaller than the baseline mean of approximately 75 due to the coherence of the clusters.

The mean search time is almost the same for the two databases even though Corel is three times larger. The Alinari database seems to be more difficult in the sense of presenting a much higher visual diversity of images that match a given concept, which works against efficient search because the system uses global visual descriptors to update the model.

7.1 Peaking of the Posterior Distribution

To illustrate how the posterior distribution changes over time, we present the values of the largest 1,000 values of $p_t(k)$ for two search sessions: one relatively efficient (class "Space Scene" in Fig. 15) and one more difficult (class "Lion" in Fig. 16). The horizontal axis shows the image index and the vertical axis shows the normalized posterior $p_t(k)$. The thick dots mark images that belong to the target class. Note that images that are close on the x-axis are not necessarily close in system's metric.

For both search sessions, we see that after one iteration, no image from the target class belongs to the 1,000 with the highest posterior mass. However, at later rounds, the distribution becomes much more "stable" in the sense that an increasing number of elements of the target class have posterior probability in the top 1,000, which makes them more likely to be chosen by the display algorithm.

The class "Space Scenes" is obviously "easier" than the class "Lion." For "Space Scenes," after seven feedback iterations there are 65 images (out of 100) that belong to the top 1,000, whereas for the class "Lion" there are only 28 images (after 15 iterations). This is explained by the fact that images in the class "Space Scenes" usually show a luminous object on a black background. Even though this particular visual pattern is matched by other images (for example "fireworks" or "night scenes"), the visual descriptors we use discriminate this class from others much better than in the case of the "Lion" class, which contains images in which the background is very diverse and the subject ("lion") has a similar low-level representation to other animals (for example "tiger").

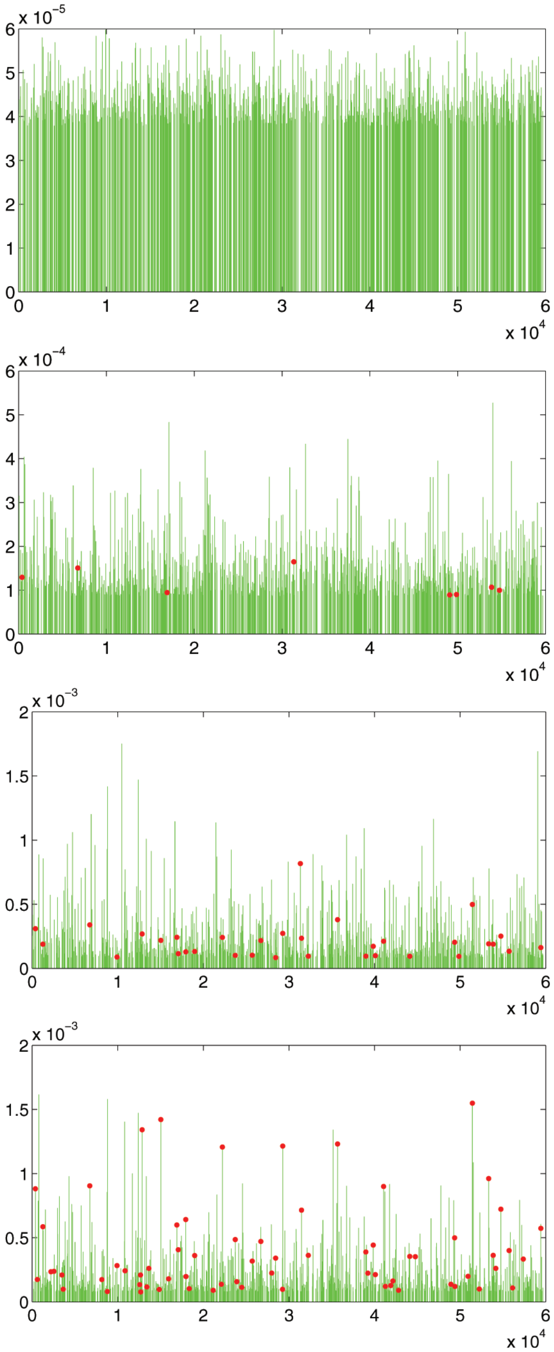


Fig. 15. Evolution of posterior $p_t(k)$ for the class "Space Scenes" (100 images). Dots represent the members of the class. Horizontal axis: The 1,000 images $k \in \Omega$ (Corel database) with the highest posterior mass. Vertical axis: $p_t(k)$ after $t = 1, 3, 5, 7$ (top to bottom). $T = 8$ for this search session.

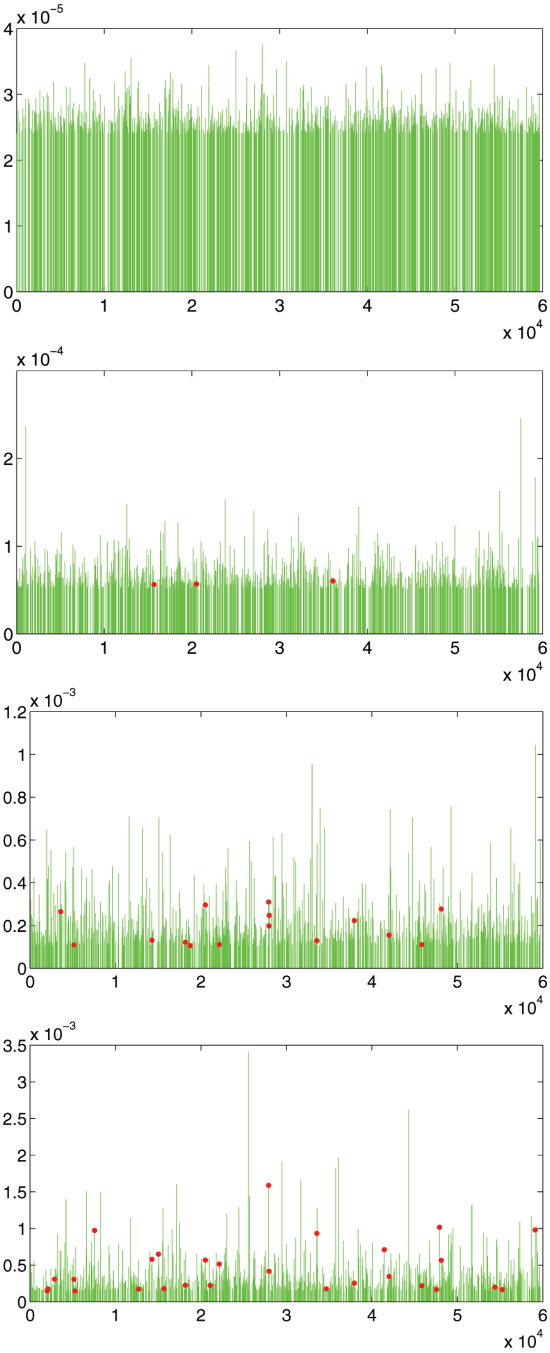


Fig. 16. Class "Lion" (100 images). Dots represent the members of the class. Horizontal axis: the 1,000 images $k \in \Omega$ (Corel database) with the highest posterior mass. Vertical axis: $p_t(k)$ after $t = 1, 5, 10, 15$ (top to bottom). $T = 16$ for this search session.

7.2 Discussion

In this section, we describe a number of issues we encountered during our experiments with users of the system.

7.2.1 Image Similarity

Image similarity has been a lively topic of debate and an active subject of research in recent years [1]. Whereas similarity can be more or less formalized for some specialized databases (e.g., fingerprints and faces), this is certainly not the case for generic databases, where similarity is strongly context-dependent and involves the expectations of the user. Even though global image descriptors are not suitable for matching specific objects, they scale well to large databases and have proven to be quite coherent with the ground truth we used for evaluating our system (see Section 5). However, they certainly fail for search concepts that are likely to be localized, in which case local descriptors would likely provide better results, albeit at the price of far more computation, and might allow the system to exploit information from the user about *parts* of the image which match his concept.

7.2.2 A "No Preference" Option

When none of the displayed images is semantically related to the target class, or shares evident color or texture motifs, users tend to spend noticeably more time making a selection. Evidently, deciding which one is most similar is somewhat arbitrary and even annoying. Some users have reported a preference for rejecting all the displayed images. This usually happened during a search session which got off to a bad start in the sense that either the posterior distribution remained very flat or actually concentrated on a part of the database which does not meet the target class.

We attempted to remedy this problem by introducing a new possible value for the user's response, $X_D = \mathrm{NP}$, standing for "no preference." We tried various modifications of the parametric forms of $p_+(i|k, D)$ and $p_-(i|k, D)$, adding a third parameter in order to account for probability of the answer NP.

None of these efforts were successful. Surprisingly, in fact, the mean number of iterations needed to reach the target class *increased*. We observed what appeared to be "overuse" of the NP option, especially when none of the displayed images was close to the target class. In the end, the flow of information was reduced: Declaring which is closest, even if none are very close, seems to convey more information than rejecting them all.

7.2.3 Mental Matching versus Visual Matching

Another surprise was that constantly displaying the target class only modestly improved performance. Recall that we begin by presenting the user with a summary of the target class, after which these images are no longer available to the user. To measure the advantage of direct visual matching, we performed a similar set of experiments, but this time with the summary displayed throughout the search session. The results are presented in Fig. 17 for the Corel database. The near-identity of the two search time distributions for early rounds might be explained by the random nature of the first few displays. Later on, when the system presents more pertinent candidates, the user might use the summary to choose a better match. But the improvement is very small, suggesting that, for most people, the level of detail in mental representations is the first-order effect in decision making.

7.2.4 Scalability

The posterior $p_t(k)$ is updated for every image k at every feedback iteration t. The complexity of the update algorithm is then linear in N, the size of the database.

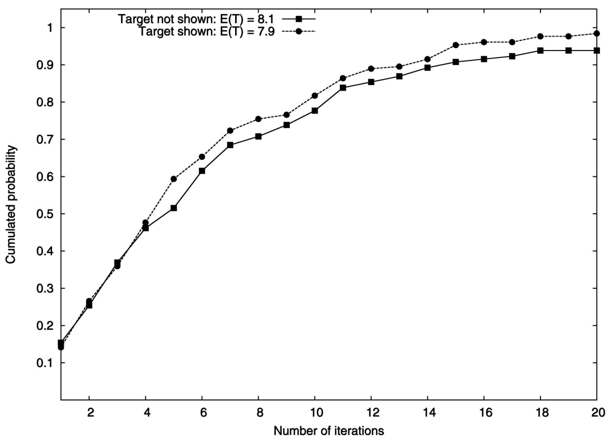


Fig. 17. Distribution of the search time for the Corel database: Displaying examples of the target class during the search does not significantly improve the results.

Our implementation is written in C++ and is not optimized for speed. Still, for our two databases, the system returns the results in less than one second on a standard PC. However, there is still a potential bottleneck for databases with millions of images. Since images are updated independently, one solution is simply a parallel version of the main algorithm.

8 Conclusion

We have presented a Bayesian framework for discovering an instance of a semantic category residing in a large, unstructured database using relevance feedback. Since the category is known only to the user of the system and since we assume no semantic annotation, the feedback is based on mental matching at the image level. Our framework centers on an evolving estimate of the probability that each member of the database belongs to the user's category. A central feature is a new Bayesian model, which includes a pair of positive and negative answer models which are designed to account for subjectivity of the user's choices and their weak correlation with the system metric. The performance of the system is validated on two fairly large databases and very reasonable search times are demonstrated.

It could be argued that semantic annotation of image databases will eventually become feasible and allow for far more efficient text-based search. In particular, searches based on visual similarity will not be necessary. There are at least two holes in this argument. First, "eventually" may be a very long time; progress in automated image interpretation is slow, especially at the level of multiple-object detection and context labeling in unconstrained scenes. Second, even if one is able to automatically annotate image databases with keywords, it is unlikely that this alone will solve the page zero problem for very large databases. (Indeed, some users may not even be able to express in words the nature of their mental pictures.) What is more plausible is that keywords provided by the user will serve to construct, online, a user-specific "prior distribution," effectively filtering a subset of images for analysis based on visual similarity. In any case, for gigantic databases, the number of images remaining as plausible candidates after textual filtering may be on the order of the sizes considered here.

Finally, our statistical framework depends only on the metric between documents; the only input is a distance matrix. Consequently, the system could, in principle, be adapted to other media. For instance, one can imagine trying to find a song in the mind of a user based on choosing among acoustical snippets in a music database, or a system for exploring a database indexed by multimodal descriptors.

Acknowledgments

This work was performed while Marin Ferecatu was a member, and Donald Geman a visiting member, of the IMEDIA research team, INRIA Rocquencourt, Le Chesnay Cedex, France. The authors are deeply grateful to Nozha Boujemaa, the director of IMEDIA, for her advice and support throughout this work. Part of this work was also carried out in the TSI Department, TELECOM ParisTech, Paris, France, where Marin Ferecatu is now a postdoctoral fellow. Finally, the work of Donald Geman was also partially supported by the US National Science Foundation under grant NSF 0427223 and by the US Office of Naval Research under contract N00014-07-1-1002.

References

[1] R. Datta, D. Joshi, J. Li, and J. Wang, "Image Retrieval: Ideas, Influences, and Trends of the New Age," *ACM Computing Surveys,* vol. 40, no. 2, pp. 5:1-60, 2008.

[2] M. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-Based Multimedia Information Retrieval: State-of-the-Art and Challenges," *ACM Trans. Multimedia Computing, Comm., and Applications,* vol. 2, no. 1, pp. 1-19, 2006.

[3] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-Based Image Retrieval at the End of the Early Years." *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 12, pp. 1349-1380, Dec. 2000.

[4] T. Gevers and A.W.M. Smeulders, "Content-Based Image Retrieval: An Overview," *Emerging Topics in Computer Vision,* G. Medioni and S.B. Kang, eds., Prentice-Hall, 2004.

[5] X.S. Zhou and T.S. Huang, "Relevance Feedback for Image Retrieval: A Comprehensive Review," *Multimedia Systems,* vol. 8, no. 6, pp. 536-544, 2003.

[6] B.L. Saux and N. Boujemaa, "Image Database Clustering with SVM-Based Class Personalization," *Proc. SPIE Conf. Storage and Retrieval Methods and Applications for Multimedia, part of Electronic Imaging Symp.,* 2004.

[7] J. Fauqueur and N. Boujemaa, "Mental Image Search by Boolean Composition of Region Categories," *Multimedia Tools and Applications,* vol. 31, no. 1, pp. 95-117, 2006.

[8] I.J. Cox, M.L. Miller, T.P. Minka, T. Papathomas, and P.N. Yianilos, "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation and Psychophysical Experiments," *IEEE Trans. Image Processing,* vol. 9, no. 1, pp. 20-37, Jan. 2000.

[9] Y. Fang and D. Geman, "Experiments in Mental Face Retrieval," *Proc. Audio- and Video-Based Biometric Person Authentication,* pp. 637-646, 2005.

[10] G. Carneiro, A.B. Chan, P.J. Moreno, and N. Vasconcelos, "Supervised Learning of Semantic Classes for Image Annotation and Retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 3, pp. 394-410, Mar. 2006.

[11] J. Li and J.Z. Wang, "Real-Time Computerized Annotation of Pictures," *Proc. ACM Multimedia Conf.,* pp. 911-920, 2006.

[12] M. Ferecatu and D. Geman, "Interactive Search for Image Categories by Mental Matching," *Proc. IEEE Int'l Conf. Computer Vision,* 2007.

[13] V. Vapnik, *Estimation of Dependencies Based on Empirical Data.* Springer Verlag, 1982.

[14] B. Schölkopf and A. Smola, *Learning with Kernels.* MIT Press, 2002.

[15] S. Tong and E. Chang, "Support Vector Machine Active Learning for Image Retrieval," *Proc. Ninth ACM Int'l Conf. Multimedia,* pp. 107-118, 2001, http://doi.acm.org/10.1145/500141. 500159.

[16] M. Ferecatu, N. Boujemaa, and M. Crucianu, "Semantic Interactive Image Retrieval Combining Visual and Conceptual Content Description," *ACM Multimedia Systems J.,* vol. 13, nos. 5/6, pp. 309-322, 2008.

[17] T. Huang, C. Dagli, S. Rajaram, E. Chang, M. Mandel, G. Poliner, and D. Ellis, "Active Learning for Interactive Multimedia Retrieval," *Proc. IEEE,* vol. 96, no. 4, pp. 648-667, 2008.

[18] K. Goh, E. Chang, and W. Lai, "Multimodal Concept Dependent Active Learning for Image Retrieval," *Proc. Ninth ACM Int'l Conf. Multimedia,* 2004.

[19] X. He, W. Ma, and H. Zhang, "Learning An Image Manifold for Retrieval," *Proc. Ninth ACM Int'l Conf. Multimedia,* 2004.

[20] H. Sahbi, P. Etyngier, J.-Y. Audibert, and R. Keriven, "Manifold Learning Using Robust Graph Laplacian for Interactive Image Retrieval," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[21] C. Hoi and M. Lyu, "A Novel Log-Based Relevance Feedback Technique in Content Based Image Retrieval," *Proc. Ninth ACM Int'l Conf. Multimedia,* 2004.

[22] Z.-H. Zhou, K.-J. Chen, and H.-B. Dai, "Enhancing Relevance Feedback in Image Retrieval Using Unlabeled Data," *ACM Trans. Information Systems,* vol. 24, no. 2, pp. 219-244, 2006.

[23] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by Image and Video Content: The QBIC System," *Computer,* vol. 28, no. 9, pp. 23-32, Sept. 1995.

[24] A. del Bimbo and P. Pala, "Visual Image Retrieval by Elastic Matching of User Sketches," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 2, pp. 121-132, Feb. 1997.

[25] A. Chalechale, G. Naghdy, and A. Mertins, "Sketch-based Image Matching Using Angular Partitioning," *IEEE Trans. Systems, Man, and Cybernetics* vol. 35, no. 1, pp. 28-41, 2005.

[26] B. Ko and H. Byun, "Integrated Region-Based Image Retrieval Using Region's Spatial Relationships," *Proc. IEEE Int'l Conf. Pattern Recognition,* 2002.

[27] G. Caenen and E.J. Pauwels, "Logistic Regression Model for Relevance Feedback in Content-Based Image Retrieval," *Proc. Storage and Retrieval for Media Databases,* pp. 49-58, 2001.

[28] N. Vasconcelos and A. Lippman, "A Probabilistic Architecture for Content-Based Image Retrieval," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2000.

[29] N. Vasconcelos and A. Lippman, "Learning from User Feedback in Image Retrieval Systems," *Proc. Conf. Advances in Neural Information Processing Systems,* 2000.

[30] Z. Su, H.-J. Zhang, S. Li, and S. Andma, "Relevance Feedback in Content-Based Image Retrieval: Bayesian Framework, Feature Subspaces, and Progressive Learning," *IEEE Trans. Image Processing,* vol. 12, no. 8, pp. 924-937, 2003.

[31] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 10, pp. 1615-1630, Oct. 2005.

[32] M. Ferecatu, "Image Retrieval with Active Relevance Feedback Using Both Visual and Keyword-Based Descriptors," PhD dissertation, INRIA—Univ. of Versailles Saint Quentin-en-Yvelines, France, 2005.

[33] C. Vertan and N. Boujemaa, "Upgrading Color Distributions for Image Retrieval: Can We Do Better?" *Proc. Int'l Conf. Visual Information Systems,* Nov. 2000.

[34] *Introduction to MPEG-7: Multimedia Content Description Interface,* B. Manjunath, P. Salembier, and T. Sikora, eds. Wiley, 2002.

[35] I. Jolliffe, *Principal Component Analysis.* Springer-Verlag, 2002.

[36] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification.* Wiley Interscience, 2001.

[37] H. Frigui and R. Krishnapuram, "A Robust Competitive Clustering Algorithm with Applications in Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, no. 5, pp. 450-465, May 1999.

[38] L. Heyer, S. Kruglyak, and S. Yooseph, "Exploring Expression Data: Identification and Analysis of Coexpressed Genes," *Genome,* vol. 9, no. 11, pp. 1106-1115, 1999.

Marin Ferecatu received the PhD degree in computer science from the University of Versailles/INRIA. He is currently a postdoctoral fellow in the TSI Department at the Institut Telecom, Telecom ParisTech, LTCI CNRS, UMR 5141, Paris, France. His current research interests include machine learning algorithms for multimedia content description and retrieval, more specifically information mining and query personalization in large image repositories using hybrid text and visual descriptors. His scientific interests include machine learning, data clustering and classification, object recognition, and content-based multimedia retrieval.

Donald Geman received the BA degree in literature from the University of Illinois and the PhD degree in mathematics from Northwestern University. He was a distinguished professor at the University of Massachusetts until 2001, when he joined the Department of Applied Mathematics and Statistics at The Johns Hopkins University, where he is a member of the Center for Imaging Science and the Institute for Computational Medicine. He also has ongoing affiliations with INRIA and the École Normale Supérieure de Cachan in France. His current research interests include statistical learning, computer vision, and computational biology. Current projects include semantic scene interpretation, molecular cancer diagnosis, and modeling gene regulatory and protein-protein interaction networks. He is a senior member of the IEEE.