J Intell Inf Syst (2008) 31:53–84
DOI 10.1007/s10844-007-0040-5

Context-sensitive queries for image retrieval in digital libraries

G. Boccignone · A. Chianese · V. Moscato · A. Picariello

Received: 8 October 2006 / Revised: 18 February 2007 / Accepted: 27 February 2007 /
Published online: 16 March 2007

Abstract In this paper we show how to achieve a more effective Query By Example processing, by using active mechanisms of biological vision, such as saccadic eye movements and fixations. In particular, we discuss the way to generate two fixation sequences from a query image I_q and a test image I_t of the data set, respectively, and how to compare the two sequences in order to compute a similarity measure between the two images. Meanwhile, we show how the approach can be used to discover and represent the hidden semantic associations among images, in terms of categories, which in turn drive the query process.

Keywords Animate vision · Image retrieval · Image indexing

1 Introduction: Is Mona Lisa a portrait or a landscape?

In the framework of Content-Based Image Retrieval (CBIR), Query By Example (QBE) is considered a suitable and promising approach because the user handles an intuitive query representation.

G. Boccignone
Dipartimento di Ingegneria dell'Informazione e Ingegneria Elettrica,
via Ponte Melillo 1, 84084, Fisciano (SA), Italy
e-mail: boccig@unisa.it

A. Chianese · V. Moscato (✉) · A. Picariello
Dipartimento di Informatica e Sistemistica, via Claudio 21, 80125 Naples, Italy
e-mail: vmoscato@unina.it

A. Chianese
e-mail: angchian@unina.it

A. Picariello
e-mail: picus@unina.it

However, a hallmark all too easily overlooked is that when the user is performing a query, he is likely to have some semantic specification in mind, e.g. "*I want to see a portrait*," and the portrait example provided to the query engine is chosen to best represent the semantics. The main problem of such approach is that it is not always easy to translate the sematic content of a query in terms of visual features, there is an inherently weak connection between the high-level semantic concepts that humans naturally associate with images and the low-level features that the computer is relying upon (Colombo et al. 1999; Djeraba 2003).

As pointed out by Santini et al. (2001), image databases mainly work within the framework of a syntactical description of the image (a scene composed of objects, that are composed of parts, etc.), and the only meaning that can be attached to an image is its similarity with the query image; namely, the meaning of the image is determined by the interaction between the user and the database.

The main issue here is that perception indeed is a relation between the perceiver and its environment, which is determined and mediated by the goals it serves (i.e., context) (Edelman 2002). Thus, considering for instance Leonardo's Mona Lisa (Fig. 1): should it be classified as a portrait or a landscape? Clearly, the answer depends on the context at hand. In this perspective, it is useful to distinguish between the "What" and "Where" aspects of the sensory input and to let the latter serve as a scaffolding holding the would-be objects in place (Edelman 2002). Such distinction offers a solution to the basic problem of scene representation - what is where - by using the visual space as its own representation and avoids the problematic early commitment to a rigid designation of an object and to its crisp segmentation from the background (on demand problem, binding problem) (Edelman 2002). Consider

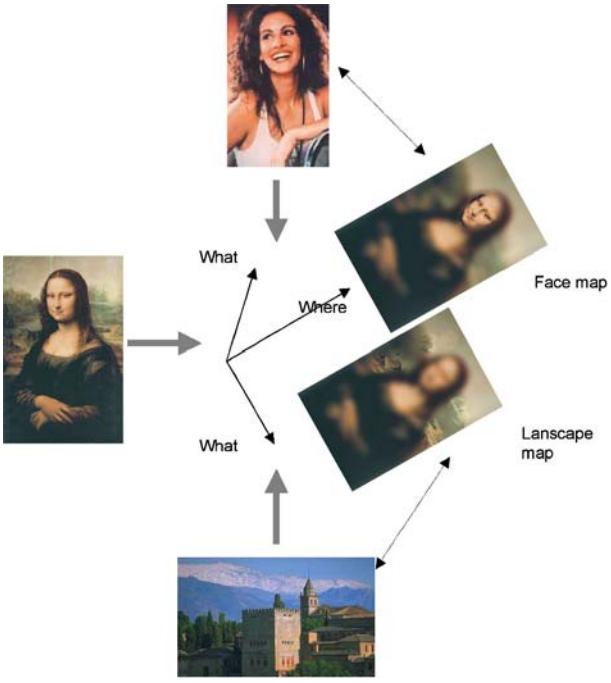

Fig. 1 The "What–Where" similarity space: the "Where" dimension (corresponding to the image location) and the two "What" dimensions (similarity to a face image and to a landscape image) are shown. Switching to one "What" dimension or to the other one, depends on the context/goal provided, here represented by a face example and a landscape example

again Fig. 1 and let Mona Lisa represent one target image I_t. An ideal unconstrained observer would scan along free viewing the picture by noting regions of interest of either the landscape and the portrait, mainly relying on physical relevance (color, contrast, etc...). However this is unlikely in real observations, since the context (goals) heavily influences the observation itself.

For example, in a face detection context, the goal is accomplished when, along visual inspection, "those" eye features are encountered "here" above "these" mouse features. On the other hand, when a landscape context is taken into account, the tree features "there" near river features "aside" may better characterize the Mona Lisa image. Clearly, in the absence of this active binding between "What" and "Where" features, the Mona Lisa picture can either be considered a portrait or a landscape; per se, it has no meaning at all.

Such dynamic binding is accomplished in natural vision through a sequence of eye movements (saccades), occurring three to four times each second; each saccade is followed by a fixation of the region of the scene, which has been focused on the high resolution part of the retina (fovea). An example of a human scanpath recorded with an eye-tracking device is provided in Fig. 2.

The computational counterpart of using gaze shifts to enable a perceptual-motor analysis of the observed world is named, after Ballard's seminal paper (Ballard 1991), `Animate Vision`.

The main contribution of this work is in the introduction of a novel representation scheme in which the "What" entities are coded by their similarities to an ensemble of reference features, and, at the same time, the "Where" aspects of the scene structure are represented by their spatial distribution with respect to the image support domain. This is obtained by generating a perceptual-motor trace of the observed image, which we denote Information Path (`IP`). Thus, the similarity of a query image I_q to a test image I_t of the data set can be assessed within the "What+Where" (`WW`) space, or equivalently by comparing their `IP`s (animate matching). In this sense we agree with (Santini et al. 2001) that the meaning that can be attached to an image is its similarity with the query image. In fact, by providing a query image, we can "shape" the `WW` space by "pinning features to a corkboard," which, in some way, corresponds

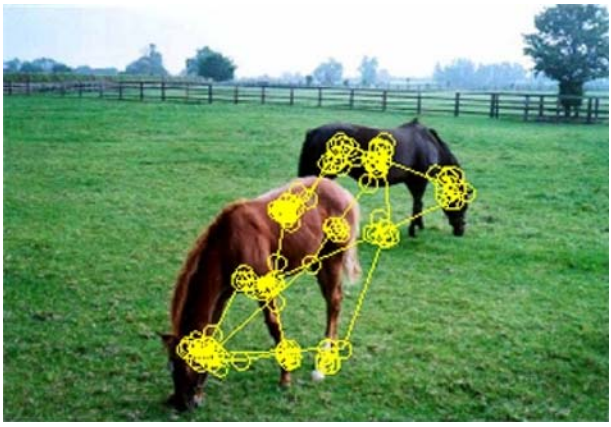

Fig. 2 A scanpath example representing the sequence of the observer's fixation points recorded while "free-viewing" the image

to shape the geometric structure of the feature space. In computer vision terms, we are exploiting "top–down" information to perform the matching.

Clearly, the approach outlined above assumes the availability of a context, and of a representation of such context in order to drive the perceptual actions in the WW space. There is a wealth of research in neurophysiology and in psychology (Fryer and Jackson 2003) showing that humans interact with the world with the aid of categories. When faced with an object or person, an individual activates a category that according to some metric best matches the given object, and in turn the availability of a category grants the individual the ability to recall patterns of behavior (stereotypes, (Fryer and Jackson 2003)) as built on past interactions with objects in a given category. In these terms, an object is not simply a physical object but a view of an interaction. The approach of grouping somehow similar images together and use these groupings (prior context) to filter out a portion of the non-relevant images for a given query is very common in the literature and allows to improve retrieval results (Newsam et al. 2001).

In the proposed system, we functionally distinguish these basic components: (1) a component which performs a "free-viewing" analysis of the images, corresponding to "bottom–up" analysis mainly relying on physical features (color, texture, shape) and derives their IPs, (2) a WW space in which different WW maps may be organized according to some selected categories (any image is to be considered the support domain upon which different maps (IPs) can be generated according to viewing purposes), (3) a query module (high level component) which acts upon the WW space by considering "top–down" information, namely, context represented through categories, and exploits animate matching to refine the search. A functional outline of the system is depicted in Fig. 3.

The paper is organized as follows. In Section 2 we briefly discuss background and related work on image indexing and retrieval problem. In Section 3, the way to map an image into the WW space is presented. In Section 4, we show how to represent context in the WW space via categories. We first discuss in general terms how categories can be clustered from a probabilistic standpoint, and in order to achieve a balanced solution of the clustering procedure a variant of the Expectation-Maximization algorithm (BEM, Balanced EM) is introduced. In Section 5 the animate query process is presented, relying on the Balanced Cluster Tree (BCT) representation of categories and the animate image matching procedure. The experimental

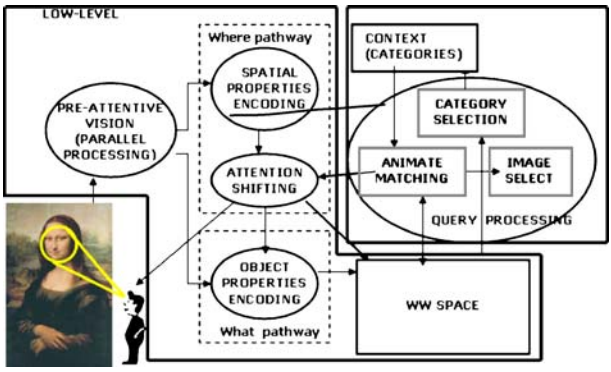

Fig. 3 A functional view of the system at a glance

protocol and related results, are discussed in Section 6. Concluding remarks are given in Section 7.

2 Related works

Traditionally, CBIR addresses the problem of finding images relevant to the users' information needs from image databases, based principally on low-level image global descriptors (color, texture and shape features) for which automatic extraction methods are available. In the past decade, systems for retrieval by visual content have been presented in the literature proposing visual features that, together with similarity measures, could provide an effective support of image retrieval (see Smeulders et al. (2000) for details). More recently, it has been realized that such global descriptors are not suitable to describe the actual objects within the images and their associated semantics. For these reasons, two main approaches have been proposed to cope with this deficiency: firstly approaches have been developed whereby the image is segmented into multiple regions, and separate descriptors are built for each region; secondly, the use of "salient points' has been suggested.

Following the first approach, different systems like PICASSO (Del Bimbo et al. 1998), SIMPLIcity (Wang et al. 2001) and Blobworld (Carson et al. 2002) have been developed. PICASSO exploits a multi-resolution color segmentation (Del Bimbo et al. 1998), in SIMPLIcity the k-means algorithm is used to cluster regions, while in Blobworld regions (blobs) are segmented via the EM algorithm. Exploited features relate to color, texture, location, and shape of regions and, the matching is accomplished through a variety of ways: using specific color distances (Del Bimbo et al. 1998), quadratic or euclidean distances (Carson et al. 2002) and integrated region matching through wavelet coefficients (Wang et al. 2001). All these systems have the problem of linking the segmented region to the actual object that is being described.

The second approach avoids the problem of segmentation altogether by choosing to describe the image and its contents in a different way. By using salient points or regions within an image, in fact, it is possible to derive a compact image description based around the local attributes of such points. It has been shown that content-based retrieval based on salient interest points and regions performs much better than global image descriptors (Hare and Lewis 2004, 2005; Sebe et al. 2003). In particular, in (Sebe et al. 2003) different operators, based on wavelet transform, are used to extract the salient point, from which region descriptors used to retrieval are built, while in Hare and Lewis (2004, 2005) salient point descriptors are evaluated using the peaks in a difference of Gaussian pyramids.

Our system follows the second approach avoiding the problem of early segmentation and exploits color, texture and shape features in the principled framework of animate vision, according to which is the way that features are dynamically organized in the WW space (Section 3) that endows them with information about the context.

It is worth recalling that the use of context/semantics is also taken into account by Wang et al. (2001), in the form of categories, by Colombo et al. (1999), Corridoni et al. (1999), in terms of color-induced sensations in paintings and clearly addressed by Santini et al. (2001), through a mechanism of similarity tuning via relevance feedback. Differently from Santini et al. (2001) and more similarly to Wang et al.

(2001), we allow for the possibility of providing the database with a preliminary context represented in terms of the likelihood to belong to a finite number of pre-specified categories.

To these purposes traditional data mining approaches, such as naive Bayes, decision-tree and SVM, can be exploited in order to classify a given image respect to its semantic belonging category. An interesting discussion of these methods is reported in Fan et al. (2005).

In our case, category discovery is obtained through a variant of the Expectation-Maximization algorithm, aimed at obtain clusters with equal number of similar images (Balanced EM, see Section 4). Such approach has the advantage to provide a means for an efficient indexing, relying on the Balanced Cluster Tree (BCT) representation of categories. The adoption of such representation avoids the well-know problems due to the fact that non-balanced partitions and the inferred index structure are not efficient in terms of time and space (Yu and Zhang 2003).

Models presented in the indexing literature are based on the key concept of proximity or similarity searching. The most promising approaches rely upon the idea of metric space, in which a similarity function is introduced by means of a distance function. In metric spaces, three types of queries are of interest: range queries retrieve all elements that are within distance r to the object; nearest neighbor queries retrieve the closest elements to the object; k-nearest neighbor queries retrieve the k closest elements to the object. The range query is widely adopted and it has been proved that the nearest neighbor query may be built over the range query concept.

Approaches relying on metric spaces are, for example, the BKT proposed by Burkhard and Keller (1973), the FQT of Baeza-Yates et al. (1994), the FQA of Chavez et al. (2001), the metric tree introduced by Uhlmann (1991) called VPT. Recently, the M-tree data structure (Ciaccia et al. 1997) has been demonstrated to be very efficient, providing dynamic capabilities and good I/O performance while requiring few distance computations. But it is well accepted that the majority of such techniques degrade rapidly as the dimensions of considered data space increase. Most index structures based on partition split a data set independent of its distribution patterns and have either a high degree of overlapping between bounding regions at high dimensions or inefficient space utilization.

To build an efficient index for a large data set with high dimensions, the overall data distributions or patterns should be considered to reduce the affects of arbitrary insertions and the clustering represents a suitable approach for discovering data patterns. To this reason the emerging techniques try to incorporate a clustering representation of the data into the classical indexing structures. To this purpose, Yu and Zhang (2003) have shown that cluster structures of the data set can be helpful in building an index structure for high dimensional data, which supports efficient queries. Indexing structure can be shaped in the form of a hierarchy of clusters and subclusters obtained via k-medoids. In the same vein, we propose a Balanced Cluster Tree, for performing range queries, but obtained via the balanced variant of the EM algorithm, which in turn takes advantage of animate query refinement (Section 5).

Eventually in (Section 6), we address the problem of evaluating the proposed system, which, due to its grounding in natural vision principles, requires figures of merit that go beyond the classic recall and precision measures (Corridoni et al. 1999; Hare and Lewis 2004; Santini 2000).

3 Mapping an image into the WW space

In most biological vision systems, only a small fraction of the information registered at any given time reaches levels of processing that directly influence behavior and, indeed, attention seems to play a major role in this process.

Visual attention is likely to be captured by salient points of the image. Each eye fixation attracted by such points, defines a focus of attention (FOA) on the foveated region of the scene, and the FOA sequence is denoted a saccadic scanpath (Noton and Stark 1990). According to scanpath theory, patterns that are visually similar, give rise to similar scanpaths when inspected by the same observer under the same viewing conditions (current task or context). In other terms a scanpath respect the properties of distinctiveness and invariance. that are requested to a salient points based technique (Sebe et al. 2003).

In general, the generation of a scanpath under free viewing conditions, can be accomplished in three steps:

1. selection of interesting regions;
2. features extraction from the detected regions;
3. search of the next interesting region.

To this aim, a pre-attentive image representation, undergoes specialized processing through the "Where" system devoted to localize a sequence of regions of interest, and the "What" system tailored for analyzing them. Attentive mechanisms provide tight integration of these two information pathways, since in the "What" pathway, feature extraction is performed, while being subjected to the action of the "Where" pathway and the related attention shifting mechanism, so that uninteresting responses are suppressed. In this way, the "Where" pathway allows to collect saliency points simulating human attentive inspection of an image.

In our system, the "Where" pathway is implemented by following the image pyramidal decomposition proposed by Itti et al. (1998). It linearly computes and combines three pre-attentive contrast maps (color, brightness, orientation) into a master or saliency map, which is then used to direct attention to the spatial location with the highest saliency through a winner take-all (WTA) network (attention shifting stage). The region surrounding such location represents the current FOA, say $\mathtt{F}_s$. By traversing spatial locations of decreasing saliency, it is then possible to observe a motor trace (scanpath) representing the stream of foveation points for an image $\mathtt{I}_i$, namely:

$$\text{scanpath} = \langle \mathtt{F}_s^i(\mathtt{p}_s; \tau_s) \rangle_{s=1,2,\ldots,\mathtt{N}_f} \tag{1}$$

where $\mathtt{p}_s = (\mathtt{x}_s, \mathtt{y}_s)$ is the center of FOA s, $\mathtt{N}_f$ is the number of explored FOAs (such parameter is set before the scanpath generation), and the delay parameter τ_s is the observation time spent on the FOA before a saccade shifts to $\mathtt{F}_{s+1}$, provided by the WTA net.

An inhibition mechanism avoids that a winner point is thoroughly reconsidered in the next steps. Figure 4 summarizes the process to obtain from an input image the related scanpath.

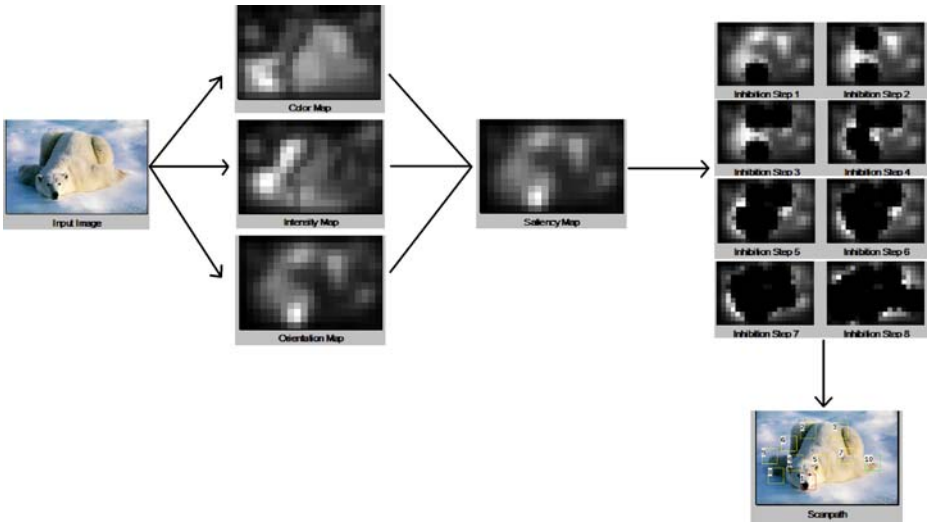

Fig. 4 The implementation of "Where" pathway. From *left* to *right*: the input image; the three conspicuity maps, representing intensity, color, orientation contrasts represented as *grey level maps* (*brighter points* are more conspicuous); the saliency map (SM) obtained by linear composition of the previous ones; eight steps of the attention shifting mechanism in which the most salient location "wins," determines the setting of the FOA, and undergoes inhibition (*darker points* in the maps) in order to allow competition among other less salient locations; the output scanpath

Note that from the "Where" pathway two dynamical features are derived: the spatial position $\mathtt{p}_s$ of each FOA and the fixation time τ_s. As demonstrated by massive experiments, the obtained scanpaths are compatible with those generated by an eye-tracker, underlying the consistent of scanpath theory.

In the "What" pathway, information is extracted from each FOA, related to color, texture and shape. In particular, for each FOA $\mathtt{F}_s^i$, the "What" pathway extracts two specific features: the color histogram $\mathtt{h}_b(\mathtt{F}_s^i)$ in the HSV representation space and the edge covariance signature $\Xi_{\mathtt{F}_s^i}$ of the image wavelet transform considering only a first level decomposition ($|\Xi| = 18$) (Mallat 1998).

Eventually, for each considered image I_i the "flow" of such features, namely the Information Path $\mathtt{IP}^i$ is generated:

$$\mathtt{IP}^i = \{\mathtt{IP}_s^i\} = \{(\mathtt{F}_s^i(\mathtt{p}_s; \tau_s), \mathtt{h}_b(\mathtt{F}_s^i), \Xi_{\mathtt{F}_s^i})\} \tag{2}$$

where $s = 1, \ldots, N_f$; an IP is thus a map, a visuomotor trace, of the image in the WW space.

Note that the process described above obtains an IP as generated under free-viewing conditions (i.e., in the absence of an observation task), which is the most general scanpath that can be recorded. Clearly, according to different viewing conditions an image may be represented by different maps in such space; such "biased" maps can be conceived as weighted IPs, or sub-paths embedded in the context-free one.

4 Endowing the WW space with context: category representation

An observer will exhibit a consistent attentive behavior while viewing a group of similar images under the same goal-driven task. This stems from the fact that we can categorize objects in categories, where each category represents a stereotyped view of the interaction with a class of objects (Fryer and Jackson 2003). Thus, in our case an image category, say $\mathcal{C}_n$, can be seen as a group of images from which, under the same viewing conditions, similar IPs could be generated.

4.1 Balanced EM learning of category clusters

We use a probabilistic framework in order to allow the association of each image (represented through its Information Path $\mathtt{IP}^i$) to different categories $\mathcal{C}_n$, $n = 1, \cdots, N_C$, and to this end we assume that an initial image set and the associated category classification have been pre-selected, through a supervised process (Duygulu et al. 2002). An efficient solution, for a very large database, is to subdivide/cluster the images belonging to a given category $\mathcal{C}_n$ into subgroups called category clusters, $\mathcal{C}_n^l$ where $l \in [1, \ldots, L_n]$ is the cluster label.

Note that each $\mathtt{IP}^i$ can be thought of as a feature vector so that the goal of clustering (MacKay 2003) is to assign a label l to the different IPs (images).

In a probabilistic setting we consider that the generic Information Path IP is an observed random variable whose values are generated by some cluster identified through a random variable $\mathcal{Z}$; we do not know in principle which cluster generates the observed data thus, $\mathcal{Z}$ is an unobserved or hidden random variable. The stochastic dependencies between variables are given by a set of parameters Θ. Namely, consider a generative model that produces a data set $\mathtt{IP} = \{\mathtt{IP}^1, \cdots, \mathtt{IP}^N\}$ consisting of N independent and identically distributed (i.i.d.) items, generated using a set of hidden clusters $\mathcal{Z} = \{z_i\}_{i=1}^N$ such that the likelihood can be written as a function of Θ:

$$p(\mathtt{IP}|\Theta) = \prod_{i=1}^{N} p(\mathtt{IP}^i|\Theta) = \prod_{i=1}^{N} \sum_{z_i} p(\mathtt{IP}^i, z_i|\Theta) \tag{3}$$

In order to use such model to perform clustering, parameters Θ must be learned. *Maximum Likelihood* (*ML*) *learning* seeks to find the parameter setting Θ^* that maximizes $p(\mathtt{IP}|\Theta)$ or the log-likelihood $\mathcal{L}(\Theta) = \log p(\mathtt{IP}|\Theta) = \sum_{i=1}^{N} \log \sum_{z_i} p(\mathtt{IP}^i, z_i|\Theta)$.

In variational approach (MacKay 2003; Neal and Hinton 1998) to ML learning, the issue of maximizing $\mathcal{L}(\Theta)$ with respect to Θ is simplified by introducing an approximating probability distribution $q(\mathcal{Z})$ over the hidden variables. It has been shown that *any* $q(\mathcal{Z})$ gives rise to a lower bound on $\mathcal{L}(\Theta)$ (MacKay 2003; Neal and Hinton 1998). By using a distinct distribution $q(z_i)$ for each data point, and via Jensen's inequality:

$$\mathcal{L}(\Theta) = \sum_{i=1}^{N} \log \sum_{z_i} p(\mathtt{IP}^i, z_i|\Theta) \geq \sum_{i=1}^{N} \sum_{z_i} q(z_i) \log \frac{p(\mathtt{IP}^i, z_i|\Theta)}{q(z_i)} = F(q, \Theta) \tag{4}$$

The lower bound $\mathtt{F}(\mathtt{q}, \Theta)$ is identified after (Neal and Hinton 1998) as the (negative) free energy:

$$\mathtt{F}(\mathtt{q}, \Theta) = \mathtt{E}_{\mathtt{q}}\left[\log \mathtt{p}(\mathtt{IP}, \mathcal{Z}|\Theta)\right] + \mathtt{H}(\mathtt{q}) \tag{5}$$

where $\mathtt{E}_{\mathtt{q}}[]$ denotes the expectation with respect to $\mathtt{q}$ and $\mathtt{H}(\mathtt{q}) = -\mathtt{E}_{\mathtt{q}}\left[\log \mathtt{q}(\mathcal{Z})\right]$ is the entropy of the hidden variables.

It is easy to show that:

$$\mathcal{L}(\Theta) = \mathtt{F}(\mathtt{q}, \Theta) + \mathcal{KL}(\mathtt{q}||p) \tag{6}$$

where $\mathcal{KL}(\mathtt{q}||p) = -\sum_{\mathtt{i}=1}^{\mathtt{N}} \sum_{\mathtt{z}_\mathtt{i}} \mathtt{q}(\mathtt{z}_\mathtt{i}) \log \frac{p(\mathtt{z}_\mathtt{i}|\mathtt{IP}^\mathtt{i}, \Theta)}{\mathtt{q}(\mathtt{z}_\mathtt{i})}$ is the Kullback–Leibler divergence (MacKay 2003) between $\mathtt{q}$ and the posterior distribution $p(\mathcal{Z}|\mathtt{IP}, \Theta)$.

Clearly $\mathtt{F}(\mathtt{q}, \Theta) = \mathcal{L}(\Theta)$ when $KL(\mathtt{q}||p) = 0$, that is when $\mathtt{q}(\mathcal{Z}) = p(\mathcal{Z}|\mathtt{IP}, \Theta)$.

A method for ML learning is the Expectation-Maximization (EM) algorithm (Dempster et al. 1977; MacKay 2003; Neal and Hinton 1998). EM alternates between an E step, which infers posterior distributions over hidden variables given a current parameter setting, and an M step, which maximises $\mathcal{L}(\Theta)$ with respect to Θ given the statistics collected from the E step. Such a set of updates can be derived using the lower bound F. At each iteration t, the E step maximises $\mathtt{F}(\mathtt{q}, \Theta)$ with respect to each of the $\mathtt{q}(\mathtt{z}_\mathtt{i})$:

$$\mathtt{q}^{(t+1)}(\mathtt{z}_\mathtt{i}) \leftarrow \arg\max_{\mathtt{q}} \mathtt{F}(\mathtt{q}, \Theta^{(t)}),\ \mathtt{i} = 1, \cdots, \mathtt{N} \tag{7}$$

and the M step maximizes $\mathtt{F}(\mathtt{q}, \Theta)$ with respect to Θ:

$$\Theta^{(t+1)} \leftarrow \arg\max_{\Theta} \mathtt{F}(\mathtt{q}^{(t+1)}, \Theta) \tag{8}$$

The E step achieves the maximum of the bound by setting $\mathtt{q}^{(t+1)}(\mathtt{z}_\mathtt{i}) = p(\mathtt{IP}^\mathtt{i}, \mathtt{z}_\mathtt{i}|\Theta^{(t)})$. It has been shown (Dempster et al. 1977; MacKay 2003; Neal and Hinton 1998) that the EM algorithm estimates the parameters so that $\mathcal{L}(\Theta^{(t)}) \leq \mathcal{L}(\Theta^{(t+1)})$ is satisfied for a sequence $\Theta^{(0)}, \Theta^{(1)}, \cdots, \Theta^{(t)}, \Theta^{(t+1)}, \cdots$, which implies that the likelihood increases monotonically and equality holds if and only if some maximum is reached.

Here we choose to model our clusters through a *Finite Gaussian Mixture* (*FGM*) (MacKay 2003) where each Information Path $\mathtt{IP}^\mathtt{i}$ is generated by one among $\mathtt{L}_\mathtt{n}$ clusters, each cluster being designed as a multidimensional Gaussian distribution $\mathcal{N}(\mathtt{IP}^\mathtt{i}; \mathbf{m}_\mathtt{l}, \Sigma_\mathtt{l})$, described by parameters $\theta_\mathtt{l} = \{\mathbf{m}_\mathtt{l}, \Sigma_\mathtt{l}\}$, the mean vector and the covariance matrix of the $\mathtt{l}$-th Gaussian, respectively. Thus the likelihood function related to the Information Path $\mathtt{IP}^\mathtt{i}$ has the form of the finite mixture:

$$p(\mathtt{IP}^\mathtt{i}|\Theta) = \sum_{l=1}^{\mathtt{L}_\mathtt{n}} \alpha_\mathtt{l} \mathcal{N}(\mathtt{IP}^\mathtt{i}; \mathbf{m}_\mathtt{l}, \Sigma_\mathtt{l}) \tag{9}$$

where $\{\alpha_\mathtt{l}\}_{\mathtt{l}=1}^{\mathtt{L}_\mathtt{n}}$ are the mixing coefficients, with $\sum_{\mathtt{l}=1}^{\mathtt{L}_\mathtt{n}} \alpha_\mathtt{l} = 1$ and $\alpha_\mathtt{l} \geq 0$ for all $\mathtt{l}$.

The complete generative model $p(\mathtt{IP}, \mathcal{Z}|\Theta)$ for the FGM can be defined as follows. Denote $\Theta = \{\alpha, \mathbf{m}, \Sigma\}$ the vector of all parameters, with $\alpha = \{\alpha_\mathtt{l}\}_{\mathtt{l}=1}^{\mathtt{L}_\mathtt{n}}$, $\mathbf{m} = \{\mathbf{m}_\mathtt{l}\}_{\mathtt{l}=1}^{\mathtt{L}_\mathtt{n}}$, $\Sigma = \{\Sigma_\mathtt{l}\}_{\mathtt{l}=1}^{\mathtt{L}_\mathtt{n}}$. The set of hidden variables is $\mathcal{Z} = \{\mathtt{z}_\mathtt{i}\}_{\mathtt{i}=1}^{\mathtt{N}}$ where each hidden variable $\mathtt{z}_\mathtt{i}$ related to observation $\mathtt{IP}^\mathtt{i}$, is a 1-of-$\mathtt{L}_\mathtt{n}$ binary vector of components $\{\mathtt{z}_{\mathtt{i}\mathtt{l}}\}_{\mathtt{l}=1}^{\mathtt{L}_\mathtt{n}}$, in which a particular element $\mathtt{z}_{\mathtt{i}\mathtt{l}}$ is equal to 1 and all other elements are equal to 0, that is $\mathtt{z}_{\mathtt{i}\mathtt{l}} \epsilon \{0, 1\}$ and $\sum_\mathtt{l} \mathtt{z}_{\mathtt{i}\mathtt{l}} = 1$. In other terms, $\mathtt{z}_\mathtt{i}$ indicates which Gaussian

component is responsible for generating Information Path $\mathtt{IP}^{\mathtt{i}}$, $p(\mathtt{IP}^{\mathtt{i}}|\mathtt{z}_{\mathtt{il}} = 1, \theta_{\mathtt{l}}) = \mathcal{N}(\mathtt{IP}^{\mathtt{i}}; \mathbf{m}_{\mathtt{l}}, \Sigma_{\mathtt{l}})$. Then the complete data likelihood is given as:

$$p(\mathtt{IP}, \mathcal{Z}|\Theta) = \prod_{\mathtt{i}=1}^{\mathtt{N}} p(\mathtt{z}_{\mathtt{i}}|\alpha) p(\mathtt{IP}^{\mathtt{i}}|\mathtt{z}_{\mathtt{i}}, \mathbf{m}, \Sigma) = \prod_{\mathtt{i}=1}^{\mathtt{N}} \prod_{\mathtt{l}=1}^{\mathtt{L_n}} \alpha_{\mathtt{l}}{}^{\mathtt{z}_{\mathtt{il}}} \mathcal{N}(\mathtt{IP}^{\mathtt{i}}, \mathbf{m}_{\mathtt{l}}, \Sigma_{\mathtt{l}})^{\mathtt{z}_{\mathtt{il}}}. \quad (10)$$

By using the expression in (10) to compute the free energy via (5) and performing the maximization according to (7) and (8), then exact estimation equations for the and steps can be derived (Dempster et al. 1977; MacKay 2003). :

$$\mathtt{h}_{\mathtt{il}}^{(t)} = \mathtt{p}(\mathtt{l}|\mathtt{IP}^{\mathtt{i}}, \theta_{\mathtt{l}}^{(t)}) = \frac{\alpha_{\mathtt{l}}^{(t)} p(\mathtt{IP}^{\mathtt{i}}|\mathtt{l}, \theta_{\mathtt{l}}^{(t)})}{\sum_{\mathtt{l}=1}^{\mathtt{L_n}} \alpha_{\mathtt{l}}^{(t)} p(\mathtt{IP}^{\mathtt{i}}|\mathtt{l}, \theta_{\mathtt{l}}^{(t)})} \quad (11)$$

$$\alpha_{\mathtt{l}}^{(t+1)} = \frac{1}{\mathtt{N}} \sum_{\mathtt{i}=1}^{\mathtt{N}} \mathtt{h}_{\mathtt{il}}, \mathbf{m}_{\mathtt{l}}^{(t+1)} = \frac{\sum_{\mathtt{i}=1}^{\mathtt{N}} \mathtt{h}_{\mathtt{il}}^{(t)} \mathtt{IP}^{\mathtt{i}}}{\sum_{\mathtt{i}=1}^{\mathtt{N}} \mathtt{h}_{\mathtt{il}}^{(t)}},$$

$$\Sigma_{\mathtt{l}}^{(t+1)} = \frac{\sum_{\mathtt{i}=1}^{\mathtt{N}} \mathtt{h}_{\mathtt{il}}^{(t)} \left[\mathtt{IP}^{\mathtt{i}} - \mathbf{m}_{\mathtt{l}}^{(t+1)}\right]\left[\mathtt{IP}^{\mathtt{i}} - \mathbf{m}_{\mathtt{l}}^{(t+1)}\right]^{T}}{\sum_{\mathtt{i}=1}^{\mathtt{N}} \mathtt{h}_{\mathtt{il}}^{(t)}} \quad (12)$$

where $\mathtt{h}_{\mathtt{il}} = \mathtt{q}(z_{\mathtt{il}} = 1) = \mathtt{p}(z_{\mathtt{il}} = 1|\mathtt{IP}^{\mathtt{i}}, \Theta)$ denotes the posterior distribution of the hidden variables given the set of parameters Θ and the observed $\mathtt{IP}^{i}$.

In principle, once ML learning is completed and the parameters Θ of the FGM model recovered, the images $\mathtt{I}_{\mathtt{i}}$ of a given category $\mathcal{C}_{\mathtt{n}}$ can be partitioned in clusters $\mathcal{C}_{\mathtt{n}} = \{\mathcal{C}_{\mathtt{n}}^{1}, \mathcal{C}_{\mathtt{n}}^{2}, \ldots, \mathcal{C}_{\mathtt{n}}^{\mathtt{L^n}}\}$, where each image $\mathtt{I}_{\mathtt{i}}$, represented through $\mathtt{IP}^{i}$, is assigned to the cluster $\mathcal{C}_{\mathtt{n}}^{\mathtt{l}}$ with the posterior probability $\mathtt{p}(\mathtt{l}|\mathtt{IP}^{\mathtt{i}}, \Theta)$.

Such straightforward procedure has some drawbacks when exploited for a very large database. On the one hand the labeling of the image bears a computational cost which is linear in time with the number of clusters $\mathtt{L}^{\mathtt{n}}$ in the category. On the other hand, for retrieval purposes, such solution is not efficient with respect to indexing issues, since the clusters obtained are in general unbalanced (do not contain the same number of images). Thus, we introduce a variant of the EM algorithm which provides a balanced clustering of the observed data, so that clusters can be organized in a suitable data structure, namely a balanced tree.

The goal is to constrain, along the E step, the distribution of the hidden variables so as to provide a balanced partition of the data, and then perform a regular M step. An example to visualize the difference between unbalanced and balanced clustering results is provided in Fig. 5.

To this end, we modify the E step as follows. First, posterior probabilities $\mathtt{h}_{\mathtt{il}}$ are computed through (11); then the procedure assigns $\mathtt{N}/\mathtt{L}$ data samples to one of the L clusters with probability 1, by selecting the first $\mathtt{N}/\mathtt{L}$ samples with higher $\mathtt{h}_{\mathtt{il}}$ probability with respect to the cluster.

For instance, for $\mathtt{L} = 2$, this gives a $\{\mathtt{N}/2, \mathtt{N}/2\}$ bipartition that maximizes the free energy. Eventually, the given partition provides the hard estimate $\mathtt{q}_{\mathtt{il}} \in \{0, 1\}$. Interestingly enough the algorithm introduces a sort of classification within the E step in the same vein of the CEM algorithm (Celeux and Govaert 1992).

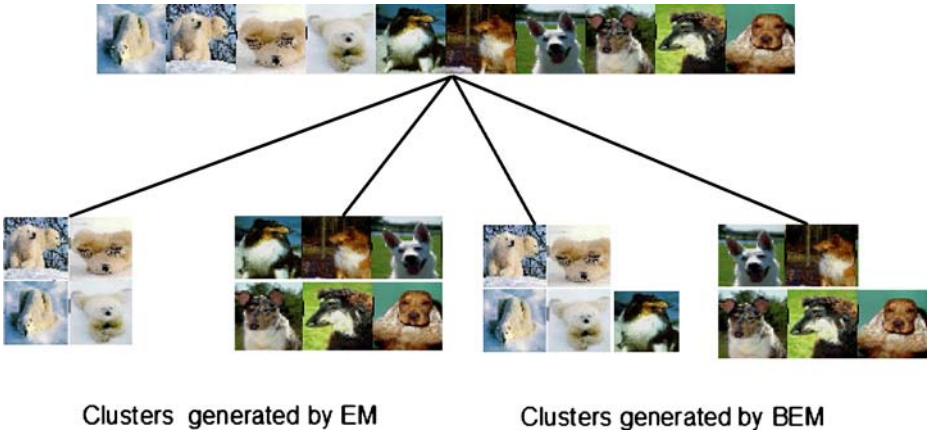

Fig. 5 Clustering results from a set of images: balanced clustering with BEM (*right*) vs. EM unbalanced clustering (*left*)

The Balanced EM algorithm (BEM) is summarized in Fig. 6.

The algorithm terminates when the convergence condition $|\mathcal{L}(\Theta^{(t+1)}) - \mathcal{L}(\Theta^{(t)})| < \epsilon$ is satisfied. In the experimental Section an example of log-likelihood maximization and convergence behavior of the algorithm will be provided (cfr. Fig. 13).

Algorithm 1 `Balanced Expectation Maximization(BEM)`

Input: the $\{\mathtt{IP}^i\}$ (i=1...N) space, the number of chosen clusters L
Output: the generated clusters described by means of a single multivariate gaussian distribution with parameters (mean and covariance) $\theta_l = \{\mathbf{m}_l, \mathbf{\Sigma}_l\}$
Initialize all $\alpha_l, \theta_l, l = 1, \cdots, L$
$t \leftarrow 1$
repeat
  {E-step}
  for $(i = 1, \ldots, N)$ **do**
    for $(l = 1, \ldots, L)$ **do**
      $h_{il}^t \leftarrow \frac{\alpha_l^t p(\mathtt{IP}^i | l, \mathbf{m}_l^t, \mathbf{\Sigma}_l^t)}{\sum_l \alpha_l^t p(\mathtt{IP}^i | l, \mathbf{m}_l^t, \mathbf{\Sigma}_l^t)}$
      $q_{il}^t \leftarrow 1$ if h_{il}^t is in the N/L highest values for class l, $q_{il}^t \leftarrow 0$ otherwise
    end for
  end for
  {M-step}
  for $(l = 1, \ldots, L)$ **do**
    $\alpha_l^{t+1} \leftarrow \frac{1}{N} \sum_i q_{il}^t$
    $\mathbf{m}_l^{t+1} \leftarrow \frac{\sum_i q_{il}^t \mathtt{IP}^i}{\sum_i q_{il}^t}$
    $\mathbf{\Sigma}_l^{t+1} \leftarrow \frac{\sum_i q_{il}^t [\mathtt{IP}^i - \mathbf{m}_l^{t+1}][\mathtt{IP}^i - \mathbf{m}_l^{t+1}]^T}{\sum_i q_{il}^t}$
  end for
  Compute $\log \mathcal{L}^{(t+1)}$
  $t \leftarrow t + 1$
until $|\log \mathcal{L}^{(t+1)} - \log \mathcal{L}^{(t)}| < \epsilon$

Fig. 6 Balanced EM algorithm

More formally, it is worth noting that the approximating distribution $\mathtt{q}$ obtained in this way, still provides a monotonically increasing likelihood. In fact, optimal balanced partitioning would require to solve, for the $\mathtt{E\text{-}step}$ the constrained optimizazion problem: $\max_{\mathtt{q}} \mathtt{F}(\mathtt{q}, \Theta)$ subject to $\sum_{\mathtt{l}=1}^{\mathtt{L}} \mathtt{q}_{\mathtt{il}} = 1, \forall \mathtt{i}, \sum_{\mathtt{i}=1}^{\mathtt{N}} \mathtt{q}_{\mathtt{il}} = \frac{\mathtt{N}}{\mathtt{L}}, \forall \mathtt{l}$, and $\mathtt{q}_{\mathtt{il}} \in \{0, 1\}, \forall \mathtt{i}, \mathtt{l}$.

Unfortunately this is an NP-hard integer programming problem, but the two substeps of the $\mathtt{E\text{-}step}$, 1) the unconstrained computation of $\mathtt{h}_{\mathtt{il}}$ and 2) the mapping $\mathtt{h}_{\mathtt{il}} \rightarrow \mathtt{q}_{\mathtt{il}}$ through the assignment of $\mathtt{N/L}$ data samples to one of the $\mathtt{L}$ clusters, by selecting the first $\mathtt{N/L}$ samples with higher $\mathtt{h}_{\mathtt{il}}$, alltogether provide a greedy heuristics to achieve a locally optimal solution (Zhong and Ghosh 2003).

Most important, the $\mathtt{q}$ distribution obtained via hard-assignment still increases the log-likelihood. In general, when the distribution of the hidden variables is computed according to the standard $\mathtt{E\text{-}step}$ then $\mathtt{q} = \mathtt{p}$ gives the optimal value of the function, which is exactly the incomplete data log-likelihood $\mathtt{F}(\mathtt{p}, \Theta) = \log \mathtt{p}(\mathtt{IP}|\Theta)$. For any other distribution $\mathtt{q} \neq \mathtt{p}$ over the hidden variables, $\mathtt{F}(\mathtt{q}, \Theta) \leq \mathtt{F}(\mathtt{p}, \Theta) = \log \mathtt{p}(\mathtt{IP}|\Theta)$, but still $\mathcal{L}(\Theta^{(t)}) \geq \mathcal{L}(\Theta^{(t+1)})$ will hold and the likelihood monotonically increase at each step t of the algorithm.

This property indeed holds for the case at hand, where $\mathtt{q}$ is obtained via a hard assignment. In fact, for $\mathtt{q}$ a partition of $\mathtt{IP}^1, \cdots, \mathtt{IP}^{\mathtt{N}}$ is defined where for each $\mathtt{IP}^{\mathtt{i}}$, there exists a label $\mathtt{l}(1 \leq \mathtt{l} \leq \mathtt{L})$ such that $\mathtt{q}(\mathtt{l}|\mathtt{IP}^{\mathtt{i}}, \Theta) = 1$. Thus $\mathtt{q}(\mathtt{l}|\mathtt{IP}^{\mathtt{i}}, \Theta) \log \mathtt{q}(\mathtt{l}|\mathcal{IP}^{i}, \Theta) = 0$ for all $1 \leq \mathtt{l} \leq \mathtt{L}$ and $1 \leq \mathtt{i} \leq \mathtt{N}$ (since $0 \log 0 = 0$, (MacKay 2003)). Hence $\mathtt{H}(\mathtt{q}) = 0$ and from (5) the following holds:

$$\mathtt{F}(\mathtt{q}, \Theta) = \mathtt{E}_{\mathtt{q}}\left[\log \mathtt{p}(\mathtt{IP}, \mathcal{Z}|\Theta)\right] \leq \mathtt{F}(\mathtt{p}, \Theta) = \log \mathtt{p}(\mathtt{IP}|\Theta), \tag{13}$$

which shows that the expectation over $\mathtt{q}$ lower bounds the likelihood of the data. Further, it has been shown (Banerjee et al. 2003) that for the choice $\mathtt{q} = 1$, if $\mathtt{l} = \arg\max_{\mathtt{l}'} \mathtt{p}(\mathtt{l}|\mathtt{IP}^{\mathtt{i}}, \Theta)$ and $\mathtt{q} = 0$ otherwise, $\mathtt{E}_{\mathtt{p}}\left[\log \mathtt{p}(\mathtt{IP}, \mathcal{Z}|\Theta)\right] \leq \mathtt{E}_{q}\left[\log p(\mathcal{IP}, \mathcal{Z}|\Theta)\right]$ holds too, so that together with (13) shows that $\mathtt{q}$ is a tight lower bound.

This proofs that at each step, $\mathcal{L}(\Theta^{(t+1)}) \geq \mathcal{L}(\Theta^{(t)})$ until at least a local maximum is reached, for which $\mathcal{L}(\Theta^{(t+1)}) = \mathcal{L}(\Theta^{(t)})$. Hence, $|\mathcal{L}(\Theta^{(t+1)}) - \mathcal{L}(\Theta^{(t)})| \rightarrow 0$ ensuring convergence of the $\mathtt{BEM}$ algorithm.

4.2 Balanced cluster tree representation

By means of $\mathtt{BEM}$ procedure, each category can be represented in terms of clusters by mapping the cluster space onto the tree-structure shown in Fig. 7a, which we denote Balanced Cluster Tree ($\mathtt{BCT}$).

Given a category $\mathcal{C}_{\mathtt{n}}$ a $\mathtt{BCT}$ of depth Υ is obtained by recursively applying the balanced EM algorithm, considering at each step $\upsilon = 0, \cdots, \Upsilon - 1$ as input of $\mathtt{BEM}$ procedure the set of clusters/sub-clusters generated in the previous step.

Each tree node of level $\upsilon + 1$ is associated with one of the discovered clusters at the υ-th iteration of the $\mathtt{BEM}$ algorithm. New discovered clusters are recursively partitioned until each category cluster contains a number of $\mathtt{IP}$s lower than a fixed threshold $\mathtt{c}_{\mathtt{f}}$, representing the desired filling-coefficient (capacity) of tree leaves.

This induces a coarse-to-fine representation, namely $\mathcal{C}_{\mathtt{n}}(\upsilon) = \{\mathcal{C}_{\mathtt{n}}^1(\upsilon), \mathcal{C}_{\mathtt{n}}^2(\upsilon), \ldots, \mathcal{C}_{\mathtt{n}}^{\mathtt{L}^{\mathtt{n}}}(\upsilon)\}_{\upsilon=0,\cdots,\Upsilon-1}$. The category sub-tree level can be calculated as $\mathtt{lev}_{\upsilon} = \log_{\mathtt{L}^{\upsilon}}(\frac{\mathtt{N}_{\mathtt{n}}}{\mathtt{c}_{\mathtt{f}}})$, $\mathtt{N}_{\mathtt{n}}$ being the number of category indexing objects, and $\mathtt{L}^{\upsilon}$ the number of clusters

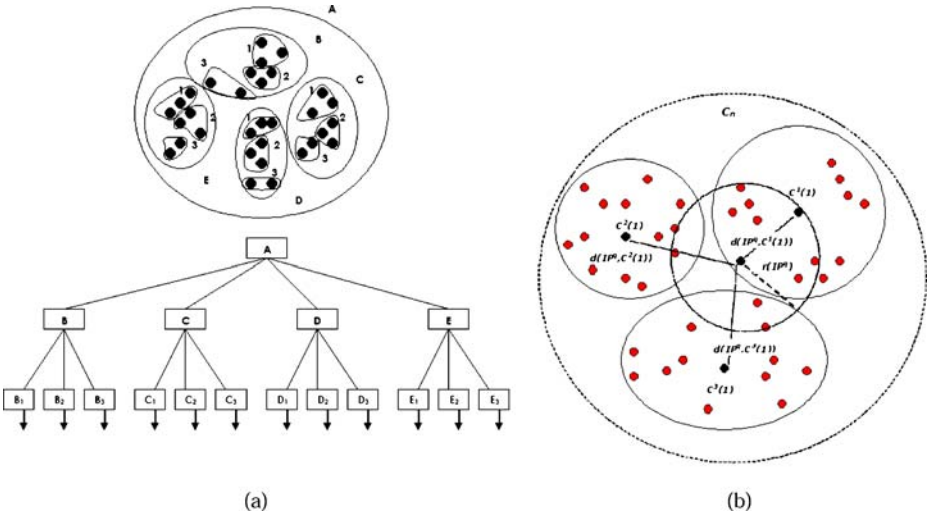

Fig. 7 **a** A 2-D representation of a BCT, **b** Range Query inside a given category $\mathcal{C}_n$: only the clusters which distance from the query object $d(IP^q, \mathcal{C}_n^1)$ is less than the query radius $r(IP^q)$ are visited

generated at υ-th BEM recursive application. In particular, as shown in Fig. 7, the root node is associated with the whole category $\mathcal{C}_n$, and the tree maintains a certain number of entry points for each node dependent on the number L^υ of wanted clusters for each tree-level; we represent the non-leaves node $\{\mathcal{C}_n^1(\upsilon), \mathcal{C}_n^2(\upsilon), \ldots, \mathcal{C}_n^{L^n}(\upsilon)\}_{\upsilon=0,\cdots,\Upsilon-1}$, at level υ by using the parameters $\mathbf{m}_n^1(\upsilon)$, and, the cluster radius $|\Sigma_n^1(\upsilon)|$, whereas leaves contain the image pointers.

Formally, we can define $BCT = \{\rho(\upsilon), \iota\}$ where the tree-nodes ("pivots," "routing nodes") and the leaves of our structure are $\rho = \langle \mathbf{m}, |\Sigma|, Ptr \rangle$ and $\iota = \langle \Gamma \rangle$, respectively. Here, $(\mathbf{m}, |\Sigma|)$ are the features representative of the current routing node, Ptr is the pointer to the parent tree-node and Γ is the set of pointer to the images on the secondary storage system. In this manner, the procedure to build our tree can be outlined by algorithm in Fig. 8 by setting $\upsilon = 1$ and $Ptr = Ptr(root_{\mathcal{C}_n})$.

Algorithm 2 Building Cluster Tree (BCT) Procedure

Input: the current level υ and the pointer Ptr to the parent node of the tree

Building Cluster Tree (υ, Ptr)

$\Upsilon = \left\lceil \log_{L_\upsilon}\left(\frac{N_n}{c_t}\right) \right\rceil$

if $\upsilon \leq \Upsilon - 1$ **then**

  for $(l = 1, ..., L_\upsilon$ **do**

    $\mathbf{m}_n^l(\upsilon), |\mathbf{\Sigma}_n^l(\upsilon)| \leftarrow BEM_{Algorithm}$

    $\rho_n^l(\upsilon) \leftarrow \{\mathbf{m}_n^l(\upsilon), |\mathbf{\Sigma}_n^l(\upsilon)|, Ptr\}$

    Building Cluster Tree $(\upsilon + 1, Ptr(\rho_n^l(\upsilon)))$

  end for

else

  $\iota_n^l(\upsilon) \leftarrow \Gamma$

end if

Fig. 8 BCT building algorithm

At this point to perform the category assignment process, we can obtain the probability, at level υ, that a test image $\mathtt{I_t}$ belongs to a category $\mathcal{C}_\mathrm{n}$ as $\mathtt{P}(\mathcal{C}_\mathrm{n}(\upsilon)|\mathtt{IP^t}) \simeq \mathtt{P}(\mathtt{IP^t}|\mathcal{C}_\mathrm{n}(\upsilon))\mathtt{P}(\mathcal{C}_\mathrm{n}(\upsilon))$, which, due to independency of clusters guaranteed by the EM algorithm, can be reformulated as:

$$\mathtt{P}(\mathcal{C}_\mathrm{n}(\upsilon)|\mathtt{IP^t}) \simeq \mathtt{P}(\mathcal{C}_n(\upsilon)) \prod_{\mathtt{l}=1}^{\mathtt{L_n}} \mathtt{p}(\mathtt{IP^t}|\mathcal{C}_\mathrm{n}^\mathtt{l}(\upsilon)) \tag{14}$$

The category discovery process can be carried out by comparing the image map $\mathtt{IP}$ with the category clusters in the $\mathtt{WW}$ space at a coarse scale ($\upsilon = 1$) and by choosing the best categories on the base of belonging probabilities of the image to the database categories obtained by (14).

Eventually, each image $\mathtt{I_t}$ is associated to probabilities of being within given categories as $\langle \mathtt{I_t} = \mathtt{P}(\mathcal{C}_\mathtt{l}|\mathtt{IP^t}), \cdots, \mathtt{P}(\mathcal{C}_\mathrm{n}|\mathtt{IP^t})\rangle$. On the other hand, given the category $\mathcal{C}_\mathrm{n}$ to which the image belongs, the search of the images can be performed by exploiting the BCT structure.

5 The animate query process

The Animate query process is where the association between the scanpath of the query image and that of the test image becomes evident. Such association is performed at two levels: the query vs. category level, which results in a selection of group of similar test image conditional on categorical prior knowledge; the query vs. most similar test image level, by exploiting attention consistency between query and test images.

More precisely, given a query image $\mathtt{I_q}$ and the dimension of the desired results set, the $\mathtt{T_k}$ most similar images are retrieved in the following steps:

- map the image in the $\mathtt{WW}$ space by computing the image path under free viewing conditions, $\mathtt{I_q} \mapsto \mathtt{IP^q}$;
- discover the best $\mathtt{K} < \mathtt{N_C}$ categories that may describe the image by using (14), but substituting $\mathtt{I_q}$ for $\mathtt{I_t}$;
- for each category $\mathcal{C}_\mathrm{n}$ among the best $\mathtt{K}$ discovered, by traversing the $\mathtt{BCT}$ associated to $\mathcal{C}_\mathrm{n}$, retrieve the $\mathtt{N_I}$ target images $\mathtt{I_t}$ within the category at minimum distance from the query image;
- refine results by choosing the $\mathtt{T_K}$ images most similar to the query image by performing a sequential scanning of the previous set of $\mathtt{KN_I}$ images and evaluating the similarity $\mathtt{A}(\mathtt{IP^t}, \mathtt{IP^q})$ between their $\mathtt{IP}$s.

Thus, in order to perform step 3 we need to efficiently browse the $\mathtt{BCT}$, while step 4 requires the specification of the similarity function $\mathtt{A} \in \mathtt{R}^+$ used to refine the results of the query process. Such two issues are addressed in the following.

5.1 Category browsing using the BCT

When a query image $\mathtt{I_q}$ is proposed, the $\mathtt{BCT}$ representing category $\mathcal{C}_\mathrm{n}$ can be traversed for retrieving the $\mathtt{N_I}$ target images $I_\mathtt{t}$, by evaluating the similarity between $\mathtt{IP^q}$ and clusters $\mathcal{C}_\mathrm{n}^\mathtt{l}(\upsilon)$ at the different levels υ of the tree.

Recall that each cluster $\mathcal{C}_{\mathtt{n}}^{\mathtt{l}}(\upsilon)$ is represented through its mean and covariance, respectively $\mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon)$, $\Sigma_{\mathtt{n}}^{l}(\upsilon)$. To this end, it is possible to define the distance $\mathtt{d}(\mathtt{IP}^{\mathtt{q}}, \mathcal{C}_{\mathtt{n}}^{\mathtt{l}}(\upsilon))$ as the distance between $\mathtt{IP}^{\mathtt{q}}$ and the cluster center $\mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon)$ weighted by covariance $\Sigma_{\mathtt{n}}^{\mathtt{l}}(\upsilon)$ (Smeulders et al. 2000):

$$\mathtt{d}(\mathtt{IP}^{\mathtt{q}}, \mathcal{C}_{\mathtt{n}}^{\mathtt{l}}(\upsilon)) = \mathtt{e}^{-(\mathtt{IP}^{\mathtt{q}}-\mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon))^{\mathtt{T}}\Sigma_{n}^{\mathtt{l}}(\upsilon)^{-1}(\mathtt{IP}^{\mathtt{q}}-\mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon))} \tag{15}$$

It is easy to verify that such distance indeed is real-valued, finite and nonnegative and satisfies symmetry and triangle inequality properties, so that $\mathtt{d}$ is a metric on the information path space and the pair $(\mathtt{IP}, \mathtt{d})$ is a metric space. In other terms the BCT is a metric balanced tree and, as such, is suitable to support operations of classic multidimensional access methods (Ciaccia et al. 1997).

Recall that a viable search technique is the range query (Ciaccia et al. 1997), which returns the objects of our distribution that have a distance lower than a fixed range query radius $\mathtt{r}(\mathtt{IP}^{\mathtt{q}})$ with respect to the query object $\mathtt{IP}^{\mathtt{q}}$. In such approach the tree-search is based on a simple concept: the node related to the region having as center $\mathbf{m}_{n}^{l}(\upsilon)$ is visited only if $\mathtt{d}(\mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon), \mathtt{IP}^{\mathtt{q}}) \leq \mathtt{r}(\mathtt{IP}^{\mathtt{q}}) + \mathtt{r}(\mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon))$, where $\mathtt{r}(\mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon))$ is the radius of the analyzed region.

The range query algorithm starts form the root node and recursively traverses all paths which cannot be excluded from leading to objects because satisfying the above inequality. The $\mathtt{r}(\mathtt{IP}^{\mathtt{q}})$ value is usually evaluated in an experimental way (Ciaccia et al. 1997). In Fig. 7b an example of a range query is shown.

For a given tree level $\upsilon >= 1$, clearly, it is not convenient to have a fixed value of $r(\mathcal{IP}^{q})$, which rather should depend on the distribution of cluster centers surrounding the query object, at a certain level of the BCT (cfr. Fig. 7).

Thus, for each level, we consider the maximum and the minimum distances between the query object and each cluster center, $\mathtt{d}_{\mathtt{min}}^{\mathtt{q}}(\upsilon)$ and $\mathtt{d}_{\mathtt{max}}^{\mathtt{q}}(\upsilon)$, respectively. Denote for simplicity, $\mathbf{m}^{\mathtt{l}} = \mathbf{m}_{\mathtt{n}}^{\mathtt{l}}(\upsilon)$ the center of the $\mathtt{l}$-th cluster of category $\mathtt{n}$, $\mathtt{l} = 1, \ldots, \mathtt{L}^{\mathtt{n}}$, surrounding the query point, and $\mathtt{d}^{\mathtt{l}}$ the distance between the latter and cluster $\mathtt{l}$. By increasing the radius through discrete steps, $\mathtt{j} = 1, 2, \ldots$, within the interval $[\mathtt{d}_{\mathtt{min}}^{\mathtt{q}}(\upsilon), \mathtt{d}_{\mathtt{max}}^{\mathtt{q}}(\upsilon)]$ and counting the number of clusters occurring within the area spanned by the radius, $\mathtt{a}_{\mathtt{j}} = \{\#\mathbf{m}^{\mathtt{l}} | \mathtt{d}^{\mathtt{l}} \leq \mathtt{r}_{\mathtt{j}}\}$, a step-wise function:

$$\mathtt{w} = \{\overline{\mathtt{a}}_1, \overline{\mathtt{a}}_2, \ldots, \overline{\mathtt{a}}_{\mathtt{k}}\} \tag{16}$$

is obtained, where normalization $\overline{\mathtt{a}}_j = \frac{\mathtt{a}_{\mathtt{j}}}{\max_{\mathtt{j}} \mathtt{a}_{\mathtt{j}}}$ constrains $\mathtt{w}$ to take values within the interval [0, 1]. Each $\mathtt{w}$ value is thus related to the number of BCT nodes we want to explore for a given query object. In other terms, given a query object $\mathtt{IP}^{\mathtt{q}}$, by choosing a value $\mathtt{s}_{\mathtt{q}}$, which specifies the span of the search, we can automatically decide, at each level of the BCT, the range query radius at that level by using the inverse mapping $\mathtt{w} \mapsto \mathtt{r}$; for instance, by setting $\mathtt{s}_{\mathtt{q}} = 1$ exploration is performed on all cluster nodes available at that level. We have experimentally verified that such mapping is well approximated by a sigmoid function, namely: $\frac{1}{1+\exp(-\varsigma \cdot (\mathtt{s}_{\mathtt{q}} - .5))}$, where $\varsigma = 0.2$ provides the best fit.

A possible procedure to exploit range query is reported by algorithm in Fig. 9.

Eventually, it is worth remarking that, for what concerns the tree updating procedures, a naive strategy would simply re-apply the classification step of BEM algorithm. However, a more elegant and efficient solution is to exploit the category detection step to assign the new item to category $\mathcal{C}_{\mathtt{n}}$ and then exploit an on-line, incremental version of the BEM algorithm to update the related tree; the incremental

Algorithm 3 Range Query Procedure

Input: $\mathtt{s_q}$, $\mathtt{IP^q}$ and $\mathcal{C}_\mathtt{n}(v)$
Range Query$(\mathcal{C}_\mathtt{n}(v), \mathtt{s_q}, \mathtt{IP^q})$
Compute $\mathtt{d^q_{max}}(v)$ and $\mathtt{d^q_{min}}(v)$
Compute $\mathtt{r}(\mathtt{IP^q})$
for $(\mathtt{i} = 1, ..., |\mathcal{C}_\mathtt{n}(v)|$ **do**
  if $v = \Upsilon - 1$ **then**
    Save Object Pointers Γ
    break
  else if $\mathtt{d}(\mathtt{IP^q}, \mathcal{C}^1_\mathtt{n}(v)) < \mathtt{r}(\mathtt{IP^q})$ **then**
    Range Query$(\mathcal{C}^1_\mathtt{n}(v+1), \mathtt{s_q}, \mathtt{IP^q})$
  end if
end for

Fig. 9 Range query algorithm

procedure updates the sufficient statistics of the expected log-likelihood only as a function of the new data item inserted in the database, which can be done in constant time (Neal and Hinton 1998; Yamanishi et al. 2004).

5.2 Refining results using attention consistency

For defining the similarity function A, we rely upon our original assumption, the IP generation process performed on a pair of similar images under the same viewing conditions will generate similar IPs, a property that we denote *attention consistency*. In Fig. 10 two similar images with respective IPs are shown.

Hence, the image-matching problem can be reduced to an IP matching; in fact, experiments performed by Walker-Smith et al. (1997), provide evidence that when observers are asked to make a direct comparison between two simultaneously presented pictures, a repeated scanning, in the shape of a FOA by FOA comparison, occurs (Walker-Smith et al. 1997). Thus, in our system, two images are similar if homologous FOAs have similar color, texture and shape features, are in the same spatial regions of the image, and are detected with similar times. The procedure, is a sort of inexact matching, which we have preliminary experimented in Boccignone et al. (2005) for video segmentation and denoted Animate Matching.

It is summarized in Fig. 11.

Given a fixation point $\mathtt{F}^{\mathtt{t}}_{\mathtt{r}}(\mathtt{p_r}; \tau_\mathtt{r})$ in the test image $\mathtt{I_t}$ belonging to category $\mathcal{C}_\mathtt{n}$, the procedure selects the homologous point $\mathtt{F^q_s}(\mathtt{p_s}; \tau_\mathtt{s})$ in the query image $\mathtt{I_q}$ among those belonging to a local temporal window, that is $\tau_\mathtt{s} \in [\mathtt{s} - \mathtt{H}, \mathtt{s} + \mathtt{H}]$. The choice is performed by computing a local similarity $\mathtt{A}^{\mathtt{r,s}}$ for the pair $\mathtt{F^t_r}$ and $\mathtt{F^q_s}$:

$$\mathtt{A}^{\mathtt{r,s}} = \alpha_\mathtt{a}\mathtt{A}^{\mathtt{r,s}}_{\mathtt{spatial}} + \beta_\mathtt{a}\mathtt{A}^{\mathtt{r,s}}_{\mathtt{temporal}} + \gamma_\mathtt{a}\mathtt{A}^{\mathtt{r,s}}_{\mathtt{visual}} \tag{17}$$

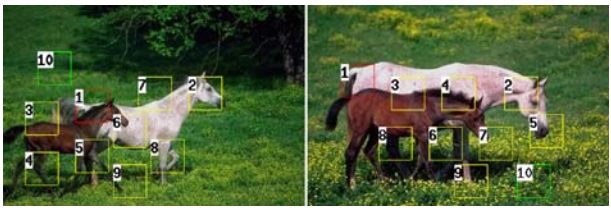

Fig. 10 Similar images with similar IPs

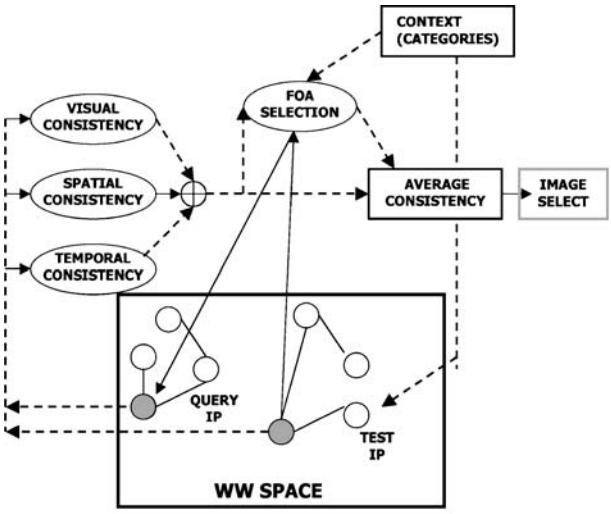

Fig. 11 Animate matching between two images represented as IPs in the WW space

where $\alpha_a, \beta_a, \gamma_a \in [0, 1]$, and by choosing the FOA s as $s = \arg\max\{A^{r,s}\}$. In other terms, the choice of the new scanpath is top–down driven by category semantics, so as to maximize the similarity of the query image with the category itself; the analyzing scanpath results to be a sub-path of the original free-viewed one. Such "best fit" is retained and eventually used to compute the consistency $A(IP^t, IP^q)$ as the average consistency of the first N'_f consistencies:

$$A = \frac{1}{N'_f} \sum_{f=1}^{N'_f} A_f^{r,s}, \tag{18}$$

where $N'_f <= N_f$, is the subset of image FOAs used for performing the matching procedure.

Right-hand terms of (17), namely $A^{r,s}_{spatial}$, $A^{r,s}_{temporal}$, $A^{r,s}_{visual}$, account for local measurements of spatial temporal and visual consistency, respectively. The former two are easily computed as $1 - d_{r,s}$ where $d_{r,s}$, generically represents the ℓ^1 distances either between (p_r, p_s) or (τ_r, τ_s) pairs, respectively.

Visual content consistency is given from the weighted mean $A^{r,s}_{visual} = \mu A^{r,s}_{col} + (1 - \mu)A^{r,s}_{tex}$, where, similarly, color and texture consistencies $A^{r,s}_{col}, A^{r,s}_{tex}$ are obtained as 1 minus the ℓ^1 distance between color histograms and between texture covariances.

The matching method above described has been widely tested on a random sample of 500 images from our image database, providing evidence of robustness with respect to physical variations of the image, in terms of increasing brightness and contrast variation, noise, translations and rotations.

An example of IP variation due to the image alterations is reported Fig. 12.

Moreover such testing stage has been useful to set, by means of ROC curves, the optimal values of all parameters for the animate matching step. In particular, the value of $N'_f = 15$ was chosen both for the matching effectiveness and the importance of earliest FOAs. The local temporal window used in the image matching algorithm

Fig. 12 An example of information path changing due to image alterations: (1,1) *Original image*; (1,2) *Brighten* 10%; (1,3) *Darken* 10%; (2,1) *More Contrast* 10%; (2,2) *Less Contrast* 10%; (2,3) *Noise Adding* 5%; (3,1) *Horizontal Shifting* 15%; (3,2) *Rotate* 90; (3,3) *Flip* 180

was set to the fixed size 4, as an experimental trade-off between retrieval accuracy and computational cost. Eventually, for what concerns the setting of equation parameters, considering again (17), we simply use $\alpha_a = \beta_a = \gamma_a = 1/3$, granting equal informational value to the three kinds of consistencies, and, similarly we set $\mu = 0.5$.

It is worth remarking that in our case traditional graph-matching algorithms are not particularly suited to the animate matching problem. Indeed here, we have to account for the presence of a temporal, sequential activity which is inherent to the animate/attentive comparison between two images (Walker-Smith et al. 1997). Also, the procedure we have conceived avoids the computational complexity typical of inexact graph matching algorithms.

6 Experimental results

Retrieval effectiveness is usually measured in the literature through recall and precision measures (Djeraba 2003). For a given number of retrieved images (the result set `rs`), the recall $\mathtt{R} = |\mathtt{rl} \cap \mathtt{rs}|/|\mathtt{rl}|$ assesses the ratio between the number of relevant

images within rs and the total number of relevant images rl in the collection, while the precision P = |rl ∩ rs|/|rs| provides the ratio between the number of relevant images retrieved and the number of retrieved images. Unfortunately, on the one hand, from a bare practical standpoint, when dealing with large databases it is difficult to estimate even approximately (Wang et al. 2001) the recall, and, in particular, the number of relevant results that have to be retrieved. On the other hand and most important, the concept of "relevant result" is often ill-defined or, at least problematic (see Corridoni et al. (1999) and Santini et al. (2000) for an in-depth discussion).

More generally, it is not easy to evaluate a system that takes into account properties like perceptual behaviors and categorization, since this necessarily involves comparison with human performance. This entails in our case the evaluation of the matching relying upon attention consistency and categorization capabilities along the query step. To this end, we consider the following issues: (1) consistency of image similarity proposed by the matching with respect to human judgement of similarity; (2) categorization performance with respect to recall and precision figures of merit; (3) semantic relevance; (4) categorization performance with respect to human categorization. Eventually, performance in terms of retrieval efficiency has also been taken into account.

Another interesting measure to evaluate the performances of an image retrieval system is the ANMRR (*Average and Normalized Mean Retrieval Rank*), provided by MPEG-7 together with an image testing collection (MPEG-7 1999). However, the number and quality of those images is not satisfying for IR evaluation. Furthermore, the ANMRR metrics cannot cover all aspects of the evaluation problem, for it mainly focuses on the rank of the retrieval result. For these reasons, we have chosen to perform our experiments on a different data set and decided to exploit the evaluation criteria discussed above in order to obtain a more effective assessment and significant comparison with other approaches in the literature.

6.1 Experimental setting

Our image database consists of about 50,000 images collected from three main data sets: the small COREL Archive (1,000), the University of Washington Ground Truth Dataset (860) and a personal collection of images from the Internet and several commercial archives (about 38,000). In particular, the COREL archive has been used for the evaluation of categorization performance in terms of precision (Wang et al. 2001), the Washington dataset for evaluating the semantic relevance of systems (Hare and Lewis 2004, 2005) and our collection for computing the query performances respect to the human categorization. Images are coded in the JPEG format at different resolution and size, and stored, together with the related IPs, into a commercial object relational DBMS.

The IP as provided *tout court* by the "What" and "Where" streams gives rise to a high dimensional feature space spanning a 2-D subspace representing the set of FOA spatial coordinates, a 768-D (256 for component) space which represents the set of FOA HSV color histograms, a 1-D subspace which represents the set of FOA WTA fire-times and a 18-D subspace which represents the set of FOA covariance signatures of the wavelet transform. To exploit the BEM algorithm, each image is represented more efficiently by performing the following reduction: the color

histogram is obtained on the HSV components quantized by using 16, 8, 8 levels for H S and V components, respectively; the covariance signatures of wavelet transform are represented through using 18 components. Eventually the clustering space becomes a $53\mathtt{N_f}$-D space, $\mathtt{N_f} = 20$ being the number of FOAs in free viewing conditions. The value of $\mathtt{N_f}$ is chosen in a experimentally way in order to ensure that the majority of saliency regions of a set of 100 random sample images, representative of the different database categories, are correctly detected respect to the judgment of 20 human observer (the human judgments on the various images are collected using an eye-tracker).

The different BCTs related to each category have been joined by means of a root node that represents the whole space of images; thus, each node of the first tree level contains the images related to a given database category. For what concerns the BCT building step, at each level $\upsilon > 1$ of the tree (we assume the root node related to level 0), a number $\mathtt{L} = 3$ was used in the recursive application of BEM algorithm due to efficiency and effectiveness aims in the retrieval task. Moreover, for each category sub-tree the total number of level $\mathtt{lev}$ was chosen considering a leaf filling coefficient $\mathtt{c} = 15$.

Note that we assume L fixed, in that we are not concerned here with the problem of model selection, in which case L may be selected by Bayesian information criterion (BIC,(MacKay 2003)). At BCT level $\upsilon = 1$, a characterization (in terms of mean and covariance) of each category is not available, so for determining the distances between query object and clusters in the range query process, mean and covariance of the whole category IP distribution are considered.

For what concerns the BEM algorithm, non uniform initial estimates were chosen for $\alpha_{\mathtt{k}}^{(0)}$, $\mu_l^{(0)}$, $\Sigma_{\mathtt{l}}^{(0)}$ parameters; $\{\mathbf{m}_{\mathtt{l}}^{(0)}\}$ were set in the range from minimal to maximal values of $\mathtt{IP}^i$ in a constant increment; $\{\Sigma_{\mathtt{l}}^{(0)}\}$ were set in the range from 1 to $\max\{\mathtt{IP}^i\}$ in a constant increment; $\{\alpha_{\mathtt{l}}^{(0)}\}$ were set from $\max\{\mathtt{IP}^i\}$ to 1 in a constant decrement and then normalized, $\sum_{\mathtt{l}} \alpha_{\mathtt{l}}^{(0)} = 1$. We found that convergence rate is similar for both methods, convergence being achieved after $\mathtt{t} = 300$ iterations (with $\epsilon = 0.1$). Figure 13 shows how the *incomplete* data log-likelihood $\log p(\mathtt{IP}|\Theta)$ as obtained

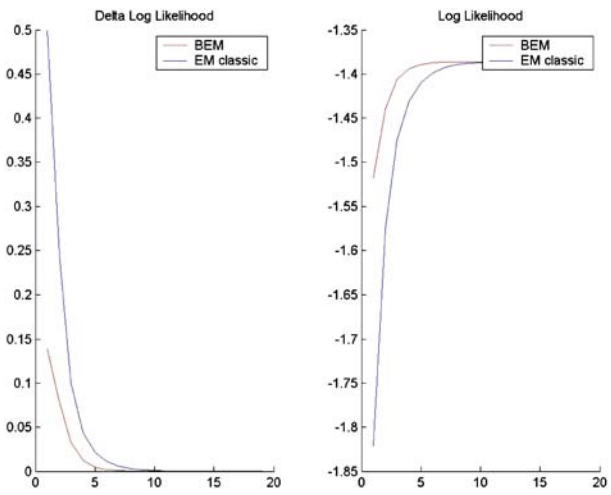

Fig. 13 Behavior of the convergence criterion $\Delta_{\mathtt{log}} = |\log \mathcal{L}^{(t+1)} - \log \mathcal{L}^{(t)}|$ (*left*) and of the log-likelihood $\log \mathtt{p}(\mathtt{IP}|\Theta)$ vs. number of iterations of the BEM algorithm compared with standard EM

by the BEM algorithm is non-decreasing at each iteration of the update, and that convergence is faster than with classic EM.

6.2 Matching effectiveness

This set of experiments aims at comparing the ranking provided by our system using the proposed similarity measure (attention consistency $\mathcal{A}$) with the ranking provided by a human observer. To such end we have slightly modified a test proposed by Santini (2000) in order to obtain a quantitative measure of the difference between the two performed rankings ("treatments," (Santini 2000)) in terms of hypothesis verification on the entire image dataset.

Consider a weighted displacement measure defined as follows (Santini 2000). Let $\mathtt{q}$ be a query on a database of $\mathtt{N}$ images that produces $\mathtt{n}$ results. There is one ordering (usually given by one or more human subjects) which is considered as the ground truth, represented as $\mathtt{L_t} = \{\mathtt{I}_1, \ldots, \mathtt{I}_n\}$. Every image in the ordering has also associated a measure of relevance $0 \leq \mathtt{S(I,q)} \leq 1$ such that (for the ground truth), $\mathtt{S}(\mathtt{I_i}, \mathtt{q}) \geq \mathtt{S}(\mathtt{I_{i+1}}, \mathtt{q})$, $\forall i$. This is compared with an (experimental) ordering $\mathtt{L_d} = \{\mathtt{I}_{\pi_1}, \ldots, \mathtt{I}_{\pi_l}\}$, where $\{\pi_1, \ldots, \pi_n\}$ is a permutation of $1, \ldots, n$. The displacement of $\mathtt{I_i}$ is defined as $\mathtt{d_q}(\mathtt{I_i}) = |\mathtt{i} - \pi_\mathtt{i}|$. The relative weighted displacement of $\mathtt{L_d}$ is defined as $\mathtt{W_q} = \frac{\sum_\mathtt{i} \mathtt{S}(\mathtt{I_i},\mathtt{I})\mathtt{d_q}(\mathtt{I_i})}{\Omega}$, where $\Omega = \lfloor \frac{\mathtt{n}^2}{2} \rfloor$ is a normalization factor. Relevance $\mathtt{S}$ is obtained from the subjects asking them to divide the results in three groups: *very similar* ($\mathtt{S}(\mathtt{I_i}, \mathtt{q}) = 1$), *quite similar* ($\mathtt{S}(\mathtt{I_i}, \mathtt{q}) = 0.5$) and *dissimilar* ($\mathtt{S}(\mathtt{I_i}, \mathtt{q}) = 0.05$).

In our experiments, on the basis of the ground truth provided by human subjects, treatments provided either by humans or by our system are compared. The goal is to determine whether the observed differences can indeed be ascribed to the different treatments or are caused by random variations. In terms of hypothesis verification, if $\mu_\mathtt{i}$ is the average score obtained with the $i\mathtt{th}$ treatment, a test is performed in order to accept or reject the null hypothesis $\mathtt{H}_0$ that all the averages $\mu_\mathtt{i}$ are the same (i.e., the differences are due only to random variations); clearly the alternate hypothesis $\mathtt{H}_\mathtt{l}$ is that the means are not equal, that is the experiment actually revealed a difference among treatments. The acceptance of $\mathtt{H}_0$ hypothesis can be checked with the $\mathtt{F}$ ratio. Assume that there are $\mathtt{m}$ treatments and $\mathtt{n}$ measurements (experiments) for each treatment. Let $\mathtt{w_{ij}}$ be the result of the jth experiment performed with the ith treatment in place. Define $\mu_\mathtt{i} = \frac{1}{\mathtt{n}} \sum_{\mathtt{j}=1}^{\mathtt{n}} \mathtt{w_{ij}}$ the average for treatment $\mathtt{i}$, $\mu = \frac{1}{\mathtt{m}} \sum_{\mathtt{i}=1}^{\mathtt{m}} \mu_\mathtt{i} = \frac{1}{\mathtt{nm}} \sum_{\mathtt{i}=1}^{\mathtt{m}} \sum_{\mathtt{j}=1}^{\mathtt{n}} \mathtt{w_{ij}}$ the total average, $\sigma_\mathtt{A}^2 = \frac{\mathtt{n}}{\mathtt{m}-1} \sum_{\mathtt{i}=1} \mathtt{m}(\mu_\mathtt{i} - \mu)^2$ the between treatments variance, $\sigma_\mathtt{W}^2 = \frac{1}{\mathtt{m(n-1)}} \sum_{\mathtt{i}=1} \mathtt{m} \sum_{\mathtt{j}=1} \mathtt{n}(\mathtt{w_{ij}} - \mu_\mathtt{i})^2$ the within treatments variance. Then, the $\mathtt{F}$ ratio is $\mathtt{F} = \frac{\sigma_\mathtt{A}^2}{\sigma_\mathtt{W}^2}$.

A high value of $\mathtt{F}$ means that the between treatments variance is preponderant with respect to the within treatment variance, that is, that the differences in the

Table 1 Mean ($\mu_\mathtt{i}$) and variance ($\sigma_\mathtt{i}^2$) of the weighted displacement for the three treatments (two human subjects and system)

	Human 1	Human 2	IP matching
$\mu_\mathtt{i}$	0.0209	0.0203	0.0190
$\sigma_\mathtt{i}^2$	$7.7771e^{-4}$	$8.1628e^{-4}$	$8.5806e^{-4}$

Table 2 The F ratio measured for pairs of distances (human vs. human and human vs. system)

F	Human 1	Human 2	IP matching
IP matching	0.3021	0.7192	0
Human 2	0.0875	0	
Human 1	0		

averages are likely to be due to the treatments. In our case we have used eight subjects selected among undergraduate student. Six students randomly chosen among the eight were employed to determine the ground truth ranking and the other two served to provide the treatments to be compared with that of our system. Four query images have been used, and for each of them a query was performed in order to provide a result set of 12 images, for a total of 48 images. Each result set was then randomly ordered and the two students were asked to rank images in the result set with respect to their similarity to the query image. Each subject was also asked to divide the ranked images in three groups: the first group consisted of images judged *very similar* to the query, the second group consisted of images judged *quite similar* to the query, and the third of *dissimilar* to the query. The mean and variance of the weighted displacement of the two subjects and of our system with respect to the ground truth are reported in Table 1.

Then, the F ratio for each pair of distances,in order to establish which differences were significant, was computed. As can be noted from Table 2 the F ratio is always less then 1 and since the critical value F_0, regardless of the confidence degree (the probability of rejecting the right hypotesis), is greater then 1, the null hypothesis can be statistically accepted. It is worth noting that the two rankings provided by the observers are consistent with one another and the attention consistency ranking is consistent with both.

6.3 Query performance via recall and precision

In this experiment we evaluate recall and precision parameters, following the systematic evaluation of image categorization performance provided by Wang et al. (2001).

Table 3 The COREL subdatabase used for query evaluation

ID	Category name	Number of images
1	Africa people and villages	100
2	Beach	100
3	Building	100
4	Buses	100
5	Dinosaurs	100
6	Elephants	100
7	Flowers	100
8	Horses	100
9	Mountains and glaciers	100
10	Food	100

Table 4 Weighted precision of our system and comparison with SIMPLIcity system and color histogram method (Wang et al. 2001)

Category ID	Our $\bar{\mathtt{p}}$	SIMPLIcity $\bar{\mathtt{p}}$ (Wang et al. 2001)	Color histogram $\bar{\mathtt{p}}$ (Wang et al. 2001)
1	0.44	0.48	0.29
2	0.42	0.31	0.29
3	0.47	0.31	0.23
4	0.60	0.37	0.28
5	0.69	0.98	0.91
6	0.45	0.40	0.39
7	0.58	0.40	0.41
8	0.49	0.71	0.39
9	0.45	0.35	0.22
10	0.53	0.35	0.21

A subset composed of ten images categories, each containing 100 pictures has been chosen from the COREL database and described in Table 3. In particular such testing database has been downloaded from http://www-db.stanford.edu/IMAGE/ web site (the images are stored in JPEG format with size 384 × 256 or 256 × 384). The ten categories reflect different semantic topics. Within such data set a retrieved image can be considered a match respect to the query image if and only if it is in the same category as the query. In this way it easy to estimate precision parameter within the first 100 retrieved images for each query, and, moreover in these conditions recall is identical to precision. In particular, for recall and precision evaluation every image in the sub-database was tested as query image and the retrieval results obtained.

In Table 4, the achieved performances and a comparison with SIMPLIcity system and LUV Color Histogram methods are reported for each category in terms of average or weighted precision ($\bar{\mathtt{p}} = \frac{1}{100}\sum_{\mathtt{k}=1}^{100}\frac{\mathtt{n_k}}{\mathtt{k}}$, where $\mathtt{k} = 1...100$ and $\mathtt{n_k}$ is the number of matches in the first k retrieved images).

For performing the previous experiment, a number of clusters equal to 3 for each tree level, a max tree level equal to 6, a leaf fan-out equal to 15 and a range query strategy using $s_q = 0.5$ have been set in the BEM tree building and traversing steps.

Figure 14a shows the top 12 results related to 2 inside query cases with the number images belonging to the same query category among the first 24 proposed ones and, and Fig. 14b, the top 12 results related to 2 outside query cases using $T_K = 100$.

For the inside query, the category belonging score computed from maximum probability $\mathtt{P}(\mathcal{C}^{\mathtt{n}}|\mathtt{IP_t})$ resulted to be 69.47% corresponding to $\mathcal{C}_{\mathtt{n}}$="Dinosaurs" for the top image and 92.63% corresponding to $\mathcal{C}_{\mathtt{n}}$="Africa" for the bottom image. For queries performed with outside images the maximum category belonging score resulted to be 62.67% corresponding to $\mathcal{C}_{\mathtt{n}}$="Horses" followed by 61.45% score corresponding to $\mathcal{C}_{\mathtt{n}}$="Elephants" for the top image, and 56.83% corresponding to $\mathcal{C}_{\mathtt{n}}$="Mountains" followed by a 56.33% score corresponding to $\mathcal{C}_{\mathtt{n}}$="Beaches" for the bottom image. In the latter case, note that the top query presents image with cows and the system retrieves images from the data set by choosing "Horses" and "Elephants" categories which are most likely to represent, with respect to other categories, the semantics of the query.

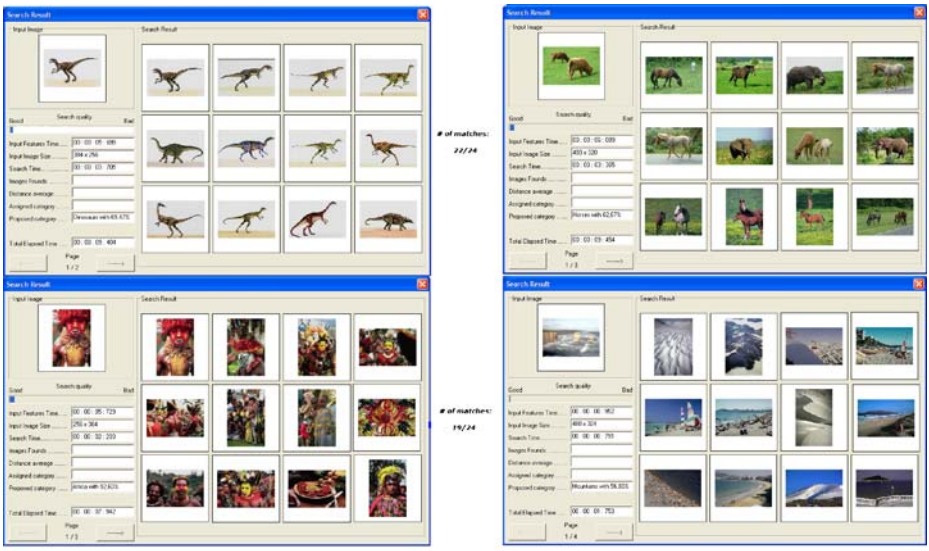

Fig. 14 Query results on the COREL subdatabase using either query images present within the data set (**a**) or outside the data set (**b**)

6.4 Semantic relevance

The problem with global descriptors is that they cannot fully describe all parts of an image having different characteristics. The use of salient regions tries to avoid such problem by developing descriptors that do capture the characteristics of each important part of an image. In order to test the effectiveness of retrieval, we have used the metric proposed in Hare and Lewis (2004) that uses semantically marked images as ground-truth against the results from our system. To such purpose, we have adopted the University of Washington Ground Truth Dataset that contains a large number of images that have been semantically marked up. For example an image may have a number of labels describing the image content (our categories), such as trees, bushes, clear sky, etc...

Given a query image with a set of labels, we should expect that the images returned by the retrieval system should have the same labels as the query image. Let $\mathtt{lab}_q$ be the set of all labels from the query image, and $\mathtt{lab}_{rs}$ be the set of labels from a returned image. The semantic relevance, $\mathtt{rel}$, of the query is definded:

$$\mathtt{rel} = \frac{\mathtt{lab}_q \cap \mathtt{lab}_{rs}}{\mathtt{lab}_q} \tag{19}$$

Table 5 Semantic relevance

Semantic relevance on rank 1 result images	Average semantic relevance on top 5 result images
49.56%	53.18%

Taking each image in the described test set in turn as a query, we calculated the animate distance to each of the other images in the result set in order to obtain a ranking of the retrieved images. We then calculated the semantic relevance for the rank one image (the closest image, not counting the query image), and we also calculated the averaged semantic relevance over the closest 5 images. The obtained results are shown in Table 5 and can be compared with the other ones discussed in Hare and Lewis (2004).

6.5 Query performance with respect to human categorization

The goal here is the evaluation of the retrieval precision of the system, with respect to the possible categories that the user has in mind when a query is performed. This measure is evaluated with respect to the whole database (50,000 images), and the following protocol has been adopted.

The not-labeled images have been grouped into about 300 categories. In order to associate the set of images to each proposed category, twenty *naive* observers were asked to perform the task on the data set, and eventually the classification has been accomplished by grouping into a category those images that the a certain number (10) of observers judged to belong to such category (it is clear that an image can belong to one or more categories).

Given a test set of 20 outside images $\mathtt{I_q}$, $\mathtt{q} = 1...20$ (in Fig. 15 some of them are shown), randomly selected out of 100 images, ten observers $\mathtt{u_j}$, $\mathtt{j} = 1...10$ (different from those that performed category identification), were asked to perform the task of choosing for each query image $\mathtt{I_q}$, the three most representative categories, say $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3$ among those describing the database. To this end, images in all categories have been presented in a hierarchial way (e.g., animals: horses, cows, etc..), to speed-up the selection process. Meanwhile, each user was asked to rank the three categories in terms of a representativeness score, within the interval [0, 100], namely: $\mathtt{R}_1^{(\mathtt{u_j},\mathtt{q})}(\mathcal{C}_1|\mathtt{I_q})$, $\mathtt{R}_2^{(\mathtt{u_j},\mathtt{q})}(\mathcal{C}_2|\mathtt{I_q})$, $\mathtt{R}_3^{(\mathtt{u_j},\mathtt{q})}(\mathcal{C}_3|\mathtt{I_q})$; the three scores were constrained to sum to 100 (e.g., a user identifies categories 1, 2, 3 for image 2 with scores 60, 30, 10).

For each image, the three most relevant categories have been chosen,according to a majority vote, by considering those that received the highest number of "hits" $\mathtt{Nh_c}$, $\mathtt{c} = 1, 2, 3$, from the observers, and each category was assigned the average score $\mathtt{R}_c^{\mathtt{q}}(\mathcal{C}_\mathtt{c}|\mathtt{I_q}) = \frac{1}{\mathtt{Nh_c}} \sum_{\mathtt{j}=1}^{\mathtt{Nh_c}} \mathtt{R}_\mathtt{c}^{(\mathtt{u_j},\mathtt{q})}(\mathcal{C}_\mathtt{c}|\mathtt{I_q})$. Results for the previous four images are reported in Table 6.

The scores $\mathtt{R}_\mathtt{c}^{\mathtt{q}}(\mathcal{C}_\mathtt{c}|\mathtt{I_q})$ are then normalized within the range [0, 1] to allow comparison with category belonging probabilities computed by the system, and the perceptually weighted precision has been calculated:

$$\mathtt{P}_\mathtt{w}^\mathtt{q} = \frac{1}{\mathtt{T_K}} \sum_{\mathtt{k}=1}^{\mathtt{T_K}} \frac{\mathtt{wn}_\mathtt{k}^\mathtt{q}}{\mathtt{k}}, \tag{20}$$

Fig. 15 Some query examples

Table 6 Representativeness score $R_c^q(\mathcal{C}_c|I_q)$ for each query image of Fig. 15

Image	User scores
1	Sunset (40%), Beaches (35%), Coasts (25%)
2	Horses (45%), People (40%), Landscapes (15%)
3	Cows (0.60%), Landscapes (0.25%), Mountains (0.15%)
4	Buildings (55%), Mountains (30%), Landscapes (15%)

where $\mathtt{wn}_\mathtt{k}^\mathtt{q}$ represents, for the query $\mathtt{q}$, the weighted average match of the $\mathtt{k}$ retrieved image with respect to user score $\mathtt{R}_\mathtt{c}^\mathtt{q}(\mathcal{C}_\mathtt{c}|\mathtt{I}_\mathtt{q})$ and belonging probability $\mathtt{P}_\mathtt{c}^\mathtt{k}(\mathcal{C}_\mathtt{c}|\mathtt{I}_\mathtt{k})$ provided by the system:

$$\mathtt{wn}_\mathtt{k}^\mathtt{q} = 1 - \frac{\sum_{\mathtt{c}=1}^{3} \mathtt{w}_\mathtt{c} |\mathtt{R}_\mathtt{c}^\mathtt{q}(\mathcal{C}_\mathtt{c}|\mathtt{I}^\mathtt{q}) - \mathtt{P}_\mathtt{c}^\mathtt{k}(\mathcal{C}_\mathtt{c}|\mathtt{I}_\mathtt{k})|}{\sum_{\mathtt{c}=1}^{3} \mathtt{w}_\mathtt{c}} \tag{21}$$

Note that a perfect match is obtained only for $\mathtt{wn}_\mathtt{k}^\mathtt{q} = 1$, that is for $|\mathtt{R}_\mathtt{c}^\mathtt{q}(\mathcal{C}_\mathtt{c}|\mathtt{I}^\mathtt{q}) - \mathtt{P}_\mathtt{c}^\mathtt{k}(\mathcal{C}_\mathtt{c}|\mathtt{I}_\mathtt{k})| = 0, \forall \mathtt{c}$. Relevance distance weights $\mathtt{w}_\mathtt{c}$ have been chosen as the decreasing values $\{1, 0.5, 0.25\}$.

In this way the perceptually weighted precision on the whole data set of 50, 000, considering the first 100 retrieved images, for the 20 tested query cases, resulted to be 0.597.

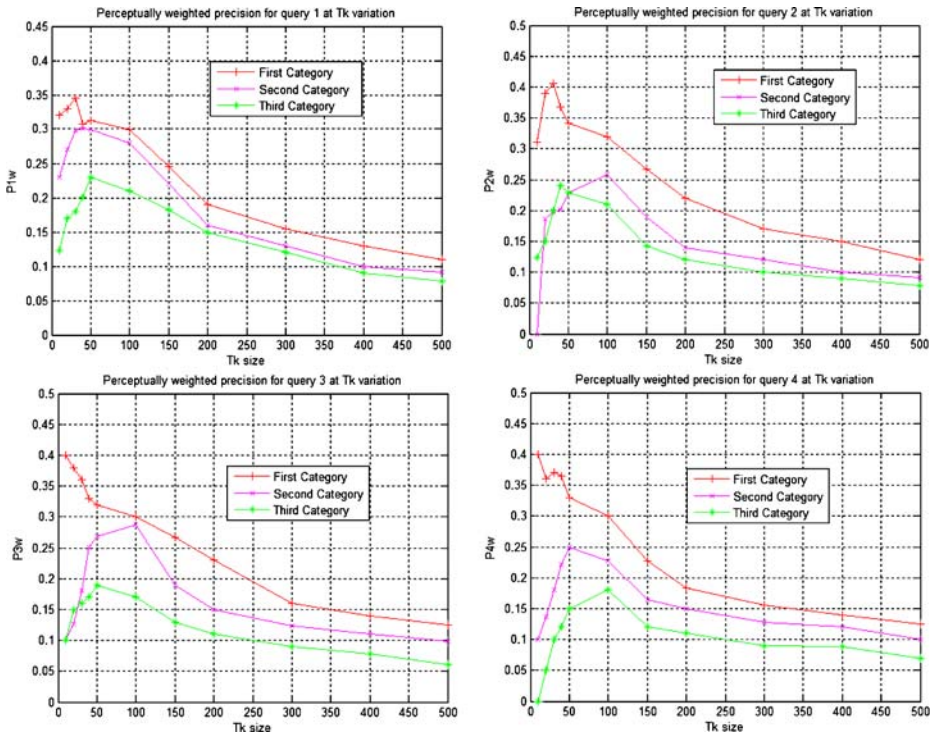

Fig. 16 Perceptually weighted precision $\mathtt{P}_\mathtt{w}^\mathtt{q}$ plotted as a function of $\mathtt{T}_\mathtt{K}$, for queries $\mathtt{q} = 1, 2, 3, 4$

Also, a query was performed for each image I_q, by considering a variable T_K of images. Figure 16, for four query cases, shows values P_w^q plotted at T_k variation. As shown in the figure, the three category belonging scores returned by system decrease to the T_K size variation, but it is possible to notice that the related proportions between system scores and user probabilities are preserved.

6.6 Retrieval efficiency

The retrieval efficiency can be evaluated in terms of time elapsed between query formulation and presentation of results. For our system the total search time t_Q is obtained from the tree search (traversing) time t_{tree} and the query refining time t_{qref} as $t_Q = t_{tree} + t_{qref}$.

Due to the indexing structure adopted, the parameters that affect the total search time are the range query radius, obtained via the s_q value, the number of clusters L, which is fixed for each level of the BCT, the tree capacity c and the number of images within the i-th category N_i. Thus, by fixing L, c, N_i, the times t_{tree} and t_{qref} are expected to increase for increasing s_q within the interval [0, 1]. The upper bounds on such quantities can be estimated as follows.

The tree search time accounts for the CPU time t_{CPU} to compute the range query distances while traversing the tree, and the I/O time t_{IO} needed to retrieve from the disk the image IPs (the storage on disk of each IP requires 32 Kb) and to transfer them to central memory, $t_{tree} = t_{CPU} + t_{IO}$. By allocating the images of a leaf node in contiguous disk sectors (by exploiting the appropriate operating system primitives) it is possible to reduce the number of disk accesses, so that $t_{CPU} >> t_{IO}$, and $t_{tree} \approx t_{CPU}$ holds.

In the worst case, $s_q = 1$:

$$t_{tree} \approx \sum_{i=1}^{N_c} \cdot \sum_{k=0}^{[\log_L(\frac{N_i}{c})]} t_d \cdot L^k \tag{22}$$

$$t_{qref} = t_{sim} \cdot \sum_{i=1}^{N_c} \left[\frac{N_i}{N_{leaves}} \right] \cdot N_{leaves} \tag{23}$$

N_c being the number of database categories. Here t_d is the time for computing a single distance, N_{leaves} the number of tree leaves. The t_{qref} parameter takes into account the fact that our tree is balanced and each leaf contains approximately the same number of images, in general $[\frac{N_i}{N_{leaves}}] <= c$.

Both t_{tree} and t_{qref} provide upper bounds in the sense that the number of evaluated distances, in the tree traversing step, is greater than the average case since, to simplify, we are not considering that in practice at each tree-level many pruned nodes occur. In fact, by setting $s_q = 1$, all nodes of the tree are explored: thus, the number of evaluated distances is equal to the total number of such nodes and the number of retrieved leaves that satisfy the range query is equal to the total number of tree leaves; on the contrary, by choosing $s_q < 1$, at each tree-level there are many pruned nodes and the number of retrieved leaves is lower than N_{leaves}.

The actual variations of times t_{tree} and t_{qref} for an increasing range query radius are plotted in Fig. 17 (here, $c = 15, L = 3$).

The experimental curves have been obtained by using a PENTIUM IV 3GHz Server (1 GB RAM), under the Windows 2003 Server operating system. To compute

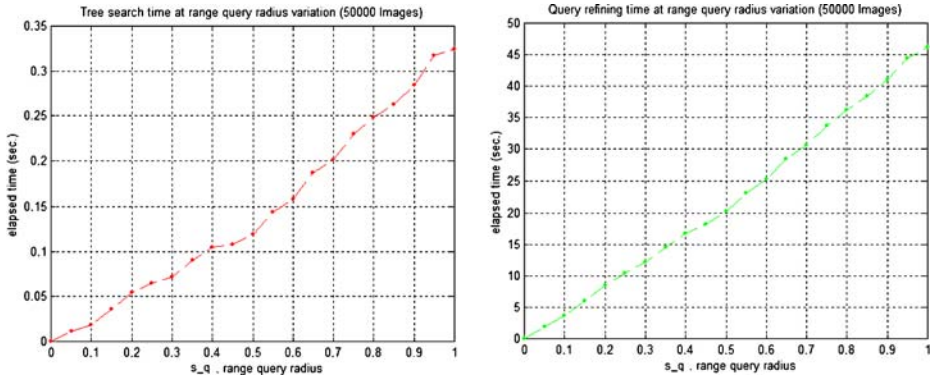

Fig. 17 Tree search and query refining time at $\mathtt{s_q}$ variation

the IP features (about 0.6 s for each image) and create the full BCT index (about 1 min for each category) on the entire database (50,000 images subdivided in about 300 categories) our system requires about 14 h. Moreover for such hardware configuration the time required for computing $\mathtt{t_d}$ is about $0.3e-4$ s. (about 25,000 CPU floating operations are necessary), and the time required for computing $\mathtt{t_{sim}}$ is about $1e-3$ s. Such results refer to the case in which the query image is present in the database; on the contrary, one extra second of CPU time is approximately spent to extract from the query image features related to the IP.

By considering $\mathtt{t_{tree}}$ and $\mathtt{t_{qref}}$, it is possible to estimate the scalability of our system and the total search times for a very large database. Assuming a database of 1,000,000 images subdivided in 2,000 categories (500 images for each category), and choosing $\mathtt{L}=3$, $\mathtt{c}=25$, we have a tree search time of about 3 s and a query refining time of about 1,000 s, in other terms, in the worst case, our system would spend about 15 min to execute a user query.

Eventually in order to have an idea of BCT performances respect to other access methods, in Fig. 18 we report the index construction time and index size at d (space-dimension) variation.

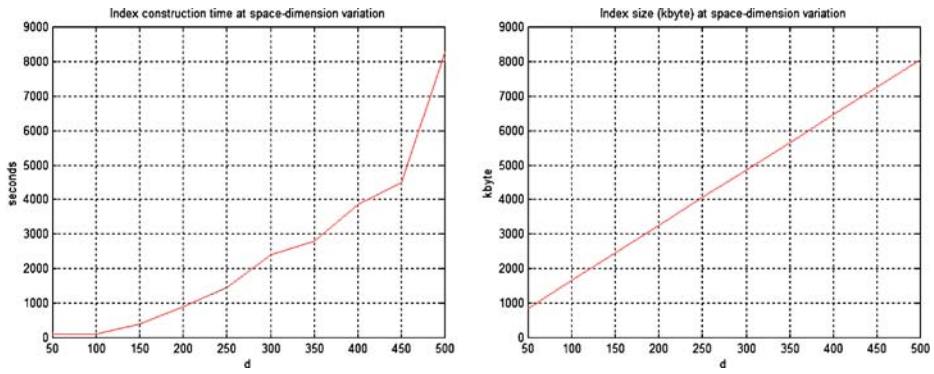

Fig. 18 Index construction time and index size at *d* variation

7 Final remarks

In this paper a novel approach to QBE has been presented. We have shown how, by embedding within image inspection algorithms active mechanisms of biological vision such as saccadic eye movements and fixations, a more effective processing can be achieved. Meanwhile, the same mechanisms can be exploited to discover and represent hidden semantic associations among images, in terms of categories, which in turn drives the query process along an animate image matching. Also, such associations allow an automatic pre-classification, which makes query processing more efficient and effective in terms of both time (the total time for presenting the output is about 4 s) and precision results.

Note that the proposed representation allows the image database to be endowed with semantics at a twofold level, namely, both at the set-up stage (learning) and at the query stage. In fact, as regards the query module it can in principle work on the given WW space learned along the training stage or by further biasing the WW by exploiting user interaction in the same vein of Santini et al. (2001). A feasible way could be that of using an interactive interface where the actions of the user (pointing, grouping, etc.) provide a feedback that can be exploited to tune on the fly parameters of the system, e.g. the category prior probability $\mathtt{P}(\mathcal{C}_{\mathrm{n}})$ or, at a lower level, the mixing coefficients in (17) to grant more information to color as opposed to texture, for instance.

Current research is devoted to such improvements as well as to extend our experiments to very large image databases. Moreover, in order to improve the effectiveness of retrieval some high-level concepts will be taken in account. To this purposes a promising approach that we are exploiting is the adoption of some *ontologies* useful to represent the semantic relations among images belonging to different categories as function of application context.

Acknowledgements The authors are grateful to the anonymous Referees and Associate Editor, for their enlightening and valuable comments that have greatly helped to improve the quality and clarity of an earlier version of this paper.

References

Baeza-Yates, R., Cunto, W., Manber, U, & Wu, S. (1994). Proximity matching using fixed-queries trees. In *Proceedings of the Fifth Combinatorial Pattern Matching* (*CPM94*), Lecture Notes in Computer Science, vol. 807 (pp. 198–212).

Ballard, D. (1991). Animate vision. *Artificial Intelligence, 48*, 57–86. (London, UK: Springer)

Burkhard, W., & Keller, R. (1973). Some approaches to best-match file searching. *Communications of the ACM, 16*(4), 230–236.

Banerjee, A., Dhillon, I. S., Ghosh, J., & Sra, S. (2003). *Clustering on hyperspheres using expectation maximization*. Technical report TR-03-07, Department of Computer Sciences, University of Texas, (February).

Boccignone, G., Chianese, A., Moscato, V., & Picariello, A. (2005). Foveated Shot Detection for Video Segmentation. *IEEE Transactions on Cicuits and Systems for Video Technology*, *15*(3), 365–377 (Marzo).

Carson, C., Belongie, S., Greenspan, H., & Malik, J. (2002). Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(8), 1026–1038.

Celeux, G., & Govaert, G. (1992). A classification EM algorithm for clustering and two stochastic versions. *Computational Statistics & Data Analysis*, *14*, 315–332.
Chavez, E., Navarro, G., Baeza-Yates, R., & Marroquin, J. M. (2001). Searching in metric space. *ACM Computing Surveys*, *33*, 273–321.
Ciaccia, P., Patella, M., & Zezula, P. (1997). M-tree: An efficient access method for similarity search in metric spaces. In *Proc. of 23rd International Conference on VLDB*, pp. 426–435.
Colombo, C., Del Bimbo, A., & Pala, P. (1999). Semantics in visual information retrieval. *IEEE MultiMedia*, *6*(3), 38–53.
Corridoni, J. M., Del Bimbo, A., & Pala, P. (1999). Image retrieval by color semantics, *Multimedia Systems*, *7*(3), 175–183.
Del Bimbo, A., Mugnaini, M., Pala, P., & Turco, F. (1998). Visual querying by color perceptive regions. *Pattern Recognition*, *31*(9), 1241–1253.
Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977) Maximum likelihood from incomplete data. *Journal of the Royal Statistical Society*, *39*, 1–38.
Duygulu, P., Barnard, K., de Freitas, N., & Forsyth, D. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Seventh European Conference on Computer Vision*, pp. 97–112.
Djeraba, C. (2003). Association and content-based retrieval. *IEEE Transactions on Knowledge and Data Engineering*, *15*(1), 118–135.
Edelman, S. (2002). Constraining the neural representation of the visual world. *Trends in Cognitive Science*, *6*(3), 125–131.
Fan, W., Davidson, I., Zadrozny, B., & Yu, P. S. (2005). An improved categorization of classifier's sensitivity on sample selection bias. In *Proocedings of International Conference on Data Mining (ICDM05)*, pp. 605–608.
Fryer, R. G., & Jackson, M. O. (2003). *Categorical cognition: A psychological model of categories and identification in decision making*. NBER Working Paper no. W9579, March.
Hare, J. S., & Lewis, P. H. (2004). Salient regions for query by image content. *Image and Video Retrieval (CIVR 2004)*, Dublin, Ireland, pp. 317–325, Springer ed.
Hare, J. S. & Lewis, P. H. (2005). On image retrieval using salient regions with vector-spaces and latent semantics. *Image and Video Retrieval (CIVR 2005)*, Singapore, Springer Ed.
Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*, 1254–1259.
MacKay, D. J. C. (2003). *Information theory, inference, and learning algorithms.* UK: Cambridge University Press.
Mallat, S. (1998). *A wavelet tour of signal processing.* San Diego, CA: Academic Press.
MPEG-7 (1999). Visual part of eXperimentation Model (XM) version 2.0. *MPEG-7 Output Document ISO/MPEG*.
Neal, R. M., & Hinton, G. E. (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants. M. J. Jordan (Ed.), *Learning in graphical models* (pp. 355–368). Cambridge, MA: MIT.
Newsam, S., Sumengen, B., & Manjunath, B. S. (2001). Category-based image retrieval. In *International Conference on Image Processing (ICIP)*, pp. 596–599.
Noton, D., & Stark, L. (1990). Scanpaths in the saccdice eye movements during pattern perception. *Visual Research*, *11*, pp. 929–942.
Santini, S. (2000). Evaluation vademecum for visual information systems. In *Proc. of SPIE*, vol. 3972. San Jose, USA.
Santini, S., Gupta, A., & Jain, R. (2001). Emergent Semantics through Interactions in image databases. *IEEE Transactions on Knowledge and Data Engineering*, *13*, 337–351.
Sebe, N., Tian, Q., Loupias, E., Lew, M., & Huang, T. (2003). Evaluation of salient point techniques. *Image and Vision Computing*, *21*, 1087–1095.
Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*, 1349–1379.
Uhlmann, J. (1991). Satisfying general proximity/similarity queries with metric trees. *Information Processing Letters*, *40*, 175–179.
Walker-Smith, G. J., Gale, A. G., & Findlay, J. M. (1997). Eye movement strategies involved in face perception. *Perception*, *6*, 313–326.

Wang, J. Z., Li, J., & Wiederhold, G. (2001). SIMPLIcity: Semantics-sensitive integrated matching for pictures libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*, 1–16, (Sept.)

Yamanishi, K., Takeuchi, J.-I., Williams, G., & Melne, P. (2004). On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. *Data Mining and Knowledge Discovery*, *8*, 275–300.

Yu, D., & Zhang, A. (2003). ClusterTree: Integration of cluster representation and nearest-neighbor search for large data sets with high dimensions. *IEEE Transactions on Knowledge and Data Engineering*, *15*(5), 1316–1337.

Zhong, S. & Ghosh, J. (2003). A unified framework for model-based clustering. *Journal of Machine Learning Research*, *4*, 1001–1037.