Available online at www.sciencedirect.com
SCIENCE DIRECT®
Pattern Recognition 38 (2005) 2007–2021
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

Hidden annotation for image retrieval with long-term relevance feedback learning

Wei Jiang[a,b,*], Guihua Er[a,c], Qionghai Dai[a,c], Jinwei Gu[a,d]

[a]*Department of Automation, Tsinghua University, Beijing 100084, China*
[b]*Room 703, Building 32, Tsinghua University, Beijing 100084, China*
[c]*Room 520, Main Building, Tsinghua University, Beijing 100084, China*
[d]*Room 626B, Main Building, Tsinghua University, Beijing 100084, China*

Received 8 July 2004; received in revised form 3 March 2005; accepted 3 March 2005

Abstract

Hidden annotation (HA) is an important research issue in content-based image retrieval (CBIR). We propose to incorporate long-term relevance feedback (LRF) with HA to increase both efficiency and retrieval accuracy of CBIR systems. The work contains two parts. (1) Through LRF, a multi-layer semantic representation is built to automatically extract hidden semantic concepts underlying images. HA with these concepts alleviates the burden of manual annotation and avoids the ambiguity problem of keyword-based annotation. (2) For each learned concept, semi-supervised learning is incorporated to automatically select a small number of candidate images for annotators to annotate, which improves efficiency of HA.

Keywords: Content-based image retrieval; Hidden annotation; Long-term relevance feedback; Multi-layer semantic representation; Semi-supervised learning

1. Introduction

Content-based image retrieval (CBIR) has been largely explored since last decades. In the CBIR context, images the *user* wants share some semantic cue, which is called the *hidden semantic concept* underlying images. Usually CBIR systems represent and retrieve images by a set of low-level visual features, which are not directly correlated with high-level hidden semantic concepts. Thus the gap between low-level features and high-level semantic concepts has been the major difficulty which limits the development of CBIR systems [1].

Hidden annotation (HA) is an effective way to bridge this feature-to-concept gap [2–8], whose aim is to form high-level semantic attributes for images. Most previous HA systems map an image into many keywords which directly reflect its semantic meaning, and combine the keyword information with low-level features to help retrieval. One major problem of these approaches is that by far keywords annotation can only be obtained from manual labeling by many *annotators*, and the annotating process is laborious and expensive, especially for large scale databases. How to alleviate the burden of manual annotation is an important issue for efficiency of HA systems. Another problem is that keywords have ambiguity, either because the richness of natural language, such as synonyms and polysemy, or because different users may use different keywords to describe the same concept. Some works use a thesaurus for annotation [5,7] to overcome the ambiguity problem. Some others pre-confine a small number of probable keywords for the

* Corresponding author. Department of Automation, Tsinghua University, Beijing 100084, China. Tel.: +86 10 62791458; fax: +86 10 62783009 804.
E-mail address: jiangwei98@mails.tsinghua.edu.cn (W. Jiang).

0031-3203/$30.00
doi:10.1016/j.patcog.2005.03.007

image database [3,6,8] to alleviate both two problems. However the effectiveness of these approaches is modest. For a practically unknown database, it is difficult to correctly pre-confine all possible semantic concepts. The thesaurus may contain too much redundant information that is not relevant to the database or the user's preference, and the cost will be very high to ask annotators to select appropriate keywords from a large thesaurus to annotate each image. A more practical approach is needed, which can both overcome the ambiguity problem of keywords, and alleviate the burden of manual annotation by narrowing down the scope of keywords and the scope of the images for annotation. To serve these ends, more semantic information about the image database is required.

Relevance feedback (RF) [9] is an effective way to get semantic information about the image database. Through a query session, the *user* labels some images to be "relevant" or "irrelevant" to his query concept, and the system uses this feedback information to help retrieval. The *long-term relevance feedback* (LRF) approaches [10–18] memorize the feedback information from users' interaction during previous retrieval sessions, and use the cumulate information as semantic experience to help subsequent retrievals. Intuitively the semantic information we need to improve the effectiveness of HA can be provided by cumulate information from LRF learning. Previous LRF approaches [10–14,16–18] record users' feedback to help subsequent retrievals, but not to extract semantic concepts for the image database. The extracted information usually does not well reflect real-world semantics, and thus can not be directly used for HA. In Ref. [15] we proposed a *multi-layer semantic representation* (MSR) to describe real-world semantics underlying images, and implemented an algorithm to automatically extract the MSR through LRF. The MSR can be directly exploited to help HA (see Section 2 for details).

In this paper, addressing the issue of incorporating LRF learning with HA to improve the annotation efficiency and retrieval performance of CBIR systems, our work can be summarized as the following two aspects:

(1) By real retrieval from different users the content underlying the image database is summarized in the MSR through LRF learning. Compared with previous LRF approaches, the major advantage of the MSR is that the learned concepts reflect real-world semantics by recording the multi-correlation among images and extracting hidden concepts, which are distributed in multiple semantic layers, for the image database (see Section 2 for details). These concepts are automatically extracted according to images in the database, which more concisely and more adaptively summarize image content than a thesaurus or pre-confined keywords. Then the concepts are provided to annotators to annote, which greatly reduces the burden of HA.

(2) For each of the learned concepts in the MSR, the labeled images are treated as training data, and a semi-supervised learning mechanism is incorporated to automatically find a small number of candidate images for this concept. These images are given to the annotators to annotate whether they are in this concept or not, and the annotation results are added into the MSR to improve the semantic knowledge of the image database.

Compared with previous HA systems, our LRF-assisted HA system has the following advantages. (1) The learned concepts in the MSR are automatically extracted from users' retrieval through the LRF process, and the effectiveness of the MSR in describing the real-world semantics makes it possible to use long-term learned information to help HA. (2) Concepts in the MSR are adaptively learned to reflect the content of the image database. These concepts are more adaptive to previously unknown databases than pre-confined keywords. Also, concepts in MSR are much more concise than a thesaurus. Annotating these concepts instead of the thesaurus greatly alleviates the burden of HA. Further more, the learned concepts are not represented by explicit keywords, but by a set of image samples in the concepts. This image-based representation avoids the ambiguity problem of the keyword-based representation. (3) For each learned concept, a small number of candidate images are automatically selected, and are given to the annotators to annotate. The candidate image set contains more *effective images* ("relevant" images to the learned concept), and the *outliers* ("irrelevant" ones) are eliminated to improve the efficiency of HA. (4) The HA process is carried when the system is in use, and the system evolves with time. As more and more retrievals are taken by users, the MSR keeps updating, and the labeled (by system users) and annotated (by annotators) semantic knowledge keeps improving. In summary, the framework of our system can be explained as follows. The burden of annotators are shared by system users. However it actually does not add any extra burden to users, because the MSR is automatically built through LRF as a byproduct of the users' retrieval process. Extensive experiments on 12,000 images show the effectiveness of our proposed method.

The rest of this paper is organized as follows. Section 2 describes the MSR, and implements the LRF algorithm to build the MSR for the image database. Section 3 proposes a HA method assisted by the long-term learned MSR with a semi-supervised learning mechanism. The experimental results are given in Section 4. We conclude our work in Section 5.

2. MSR Learning through LRF

In this part, we introduce the previous works on LRF, followed by the motivation and detailed techniques of building the MSR through LRF.

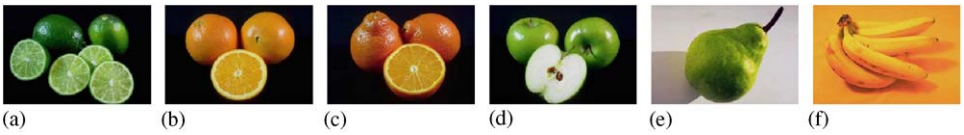

Fig. 1. An example to illustrate that semantic concepts are determined by a group of images. When (a, b, c) are "relevant" and others "irrelevant", the query concept is "orange"; when (a, d, e) are "relevant", others "irrelevant", the query concept is "green color fruit".

2.1. Previous works on LRF

There are two kinds of RF mechanisms. The *short-term relevance feedback* (SRF) learning can be viewed as a supervised learning process [19,20], and is applied to retrieval during a single query session. In each feedback round, the user labels some images as "relevant" or "irrelevant" to the query concept, and supervises the system to adjust searching in subsequent retrieval rounds. However the learned information is discarded when this query session ends. The LRF learning [10–14,16–18] memorizes the feedback information from previous users' interaction during each query session, and accumulates the information as semantic experiences to help retrieval in subsequent retrievals.

Previous LRF approaches can be classified into three categories. (1) The bi-correlation approach [17]. The statistical bi-correlations between image pairs are recorded to calculate semantic similarity between images, which is combined with low-level similarity to help retrieval. (2) The *latent semantic indexing* (LSI) method [13,14,18]. "Relevant" images for each query are memorized to form a semantic space, whose redundancy is reduced by dimensionality reduction techniques. The semantic features are combined with low-level features to help retrieval. (3) The clustering approach [10–12,16]. Images are clustered into several groups, with each group representing one hidden concept. Information from image groups is used as prior knowledge to help retrieval.

2.2. Our motivation

However, the real-world hidden semantics has two characteristics. (1) A query concept is usually determined by the *multi-correlations* among a group of images, including both "relevant" and "irrelevant" ones. For example images in Fig. 1 come from the semantic category "fruit". When (a, b, c) are labeled as "relevant", and others "irrelevant", the query concept is "orange"; when (a, d, e) are labeled as "relevant", and others "irrelevant", the query concept is "green color fruit". Representation of the multi-correlations by statistical bi-correlations, as the bi-correlation approach does, is usually not precise enough. For example, more users label that the similarity between (a) and (b) are larger than that between (b) and (f), but this statistical bi-correlation information may be not suitable for a particular query session asking for "yellow color fruit", where the similarity between (b) and (f) should be larger than that between (a) and (b). (2) The real-world semantics should have *multiple semantic layers*, with one layer corresponding to one kind of hard partitions of the hidden semantic space. Some concepts have intrinsic *intersections*, e.g., images from "green color" and "orange" can not be hard divided, and should be in different semantic layers. The clustering approach divides the semantic space into one kind of hard partitions, which does not accord with this property.

2.3. MSR learning

We propose a MSR to reflect the real-world semantics, which has two principles to be built. (1) The MSR should have multiple semantic layers, one representing one kind of hard partitions of the semantic space. (2) To provide a concise form, each semantic layer should contain as many concepts as possible, and the number of layers should be as small as possible. Based on these criterions, the relationship between semantic concepts, and that between semantic concepts and semantic layers, can be defined in the following subsections. And an algorithm is implemented to automatically learn the MSR through LRF.

2.3.1. Relationship between concepts

Let c_i, c_j denote two hidden concepts, $\mathscr{C}_i$ and $\overline{\mathscr{C}}_i$ are the corresponding "relevant" and "irrelevant" image sets for c_i respectively, where image $\mathbf{x} \in \mathscr{C}_i$ is labeled to be "relevant" to concept c_i, and image $\mathbf{x} \in \overline{\mathscr{C}}_i$ is labeled to be "irrelevant" to c_i. So are $\mathscr{C}_j$ and $\overline{\mathscr{C}}_j$ for c_j. The relationship of c_i and c_j may fall into one of the following cases (Fig. 2):

(1) c_i is a sub-concept of c_j: $\mathscr{C}_i \cap \mathscr{C}_j \neq \phi$ and $\mathscr{C}_i \cap \overline{\mathscr{C}}_j = \phi$ and $\mathscr{C}_j \cap \overline{\mathscr{C}}_i \neq \phi$.
(2) c_i and c_j are different concepts, but they have intersection: $\mathscr{C}_i \cap \mathscr{C}_j \neq \phi$ and $\mathscr{C}_i \cap \overline{\mathscr{C}}_j \neq \phi$ and $\mathscr{C}_j \cap \overline{\mathscr{C}}_i \neq \phi$.
(3) c_i probably is equal to c_j: $\mathscr{C}_i \cap \mathscr{C}_j \neq \phi$ and $\mathscr{C}_i \cap \overline{\mathscr{C}}_j = \phi$ and $\mathscr{C}_j \cap \overline{\mathscr{C}}_i = \phi$.
(4) c_i and c_j have no relationship: $\mathscr{C}_i \cap \mathscr{C}_j = \phi$.

Define an indicator $Ct(c_i, c_j)$ to describe the relationship between c_i and c_j. For cases (1) and (2), c_i and c_j belong to different semantic layers, and we say that c_i and c_j are *incompatible*, and $Ct(c_i, c_j) = 0$; for case (3), c_i and c_j probably belong to the same layer, and we say that they

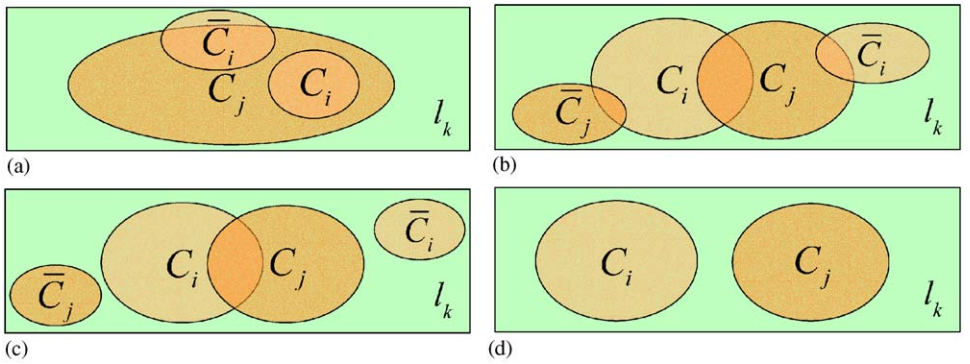

Fig. 2. The relationship between c_i and c_j. (a) c_i is a subset of c_j; (b) c_i and c_j are different concepts, but they have intersection; (c) c_i probably is equal to c_j; (d) c_i and c_j have no relationship.

are *compatible*, and $Ct(c_i, c_j) = 1$; for case (4), c_i and c_j are also compatible, but they have no relation between each other, and $Ct(c_i, c_j) = -1$. Thus $Ct(c_i, c_j)$ can be concisely given by

Definition 1. $[Ct(c_i, c_j)]$:

$$Ct(c_i, c_j) = \begin{cases} -1, & \mathscr{C}_i \cap \mathscr{C}_j = \phi, \\ 0, & \mathscr{C}_i \cap \mathscr{C}_j \neq \phi \text{ and} \\ & \quad \{\mathscr{C}_i \cap \overline{\mathscr{C}}_j \neq \phi \text{ or } \mathscr{C}_j \cap \overline{\mathscr{C}}_i \neq \phi\}, \\ 1, & \mathscr{C}_i \cap \mathscr{C}_j \neq \phi \text{ and} \\ & \quad \mathscr{C}_i \cap \overline{\mathscr{C}}_i = \phi \text{ and } \mathscr{C}_j \cap \overline{\mathscr{C}}_i = \phi. \end{cases}$$

2.3.2. Relationship between concepts and layers

Assume that l_k is one semantic layer, and there are n concepts, $c_1^k, \ldots, c_n^k$, in this layer. Set

$$n_0^k = \sum_{c_j^k \in l_k} I(Ct(c_i, c_j^k) = 0),$$
$$n_1^k = \sum_{c_j^k \in l_k} I(Ct(c_i, c_j^k) = 1), \quad (1)$$

where $I(A) = 1$ if A is true, and $I(A) = 0$ otherwise. n_0^k is the number of concepts in l_k which have intersection with c_i but are different from c_i; n_1^k is the number of concepts in l_k which may be equal to c_i. Define an indicator $Lt(c_i, l_k)$ to represent the relationship between c_i and l_k. If $n_0^k > 0$, there exists at least one concept in l_k incompatible with c_i, and c_i belongs to a different layer from l_k, and we say that c_i is *incompatible* with l_k, and $Lt(c_i, l_k) = 0$ (Fig. 3 (a)); otherwise, if $n_1^k > 1$, there are at least two concepts in l_k which may be equal to c_i, and c_i is also incompatible with l_k and belongs to another layer different from l_k, and $Lt(c_i, l_k) = 0$ (Fig. 3 (b)); if $n_1^k = 1$, c_i is equal to the concept c_j^k, $c_j^k \in l_k$, which has $Ct(c_j^k, c_i) = 1$, and we say that c_i is *compatible* with l_k and $Lt(c_i, l_k) = 1$ (Fig. 3 (c)); finally if $n_1^k = 0$, c_i has no relation with all concepts in l_k, and may be a new concept in l_k, and then c_i is also compatible with l_k and $Lt(c_i, l_k) = 1$ (Fig. 3 (d)). Thus $Lt(c_i, l_k)$ can be concisely given by the following definition:

Definition 2. $[Lt(c_i, l_k)]$:

$$Lt(c_i, l_k) = \begin{cases} 0, & n_0^k > 0 \text{ or } n_1^k > 1, \\ 1, & \text{otherwise.} \end{cases}$$

2.3.3. Algorithm implementation for MSR learning

Assume that we have already learned N hidden semantic concepts $c_1, \ldots, c_N$, which are distributed in M semantic layers $l_1, \ldots, l_M$. Suppose that in a new query session, images in $\mathscr{R}$ and $\mathscr{IR}$ are labeled to be "relevant" and "irrelevant", respectively, by the user. c_q denotes the current query concept. Let S_L and S_C denote the layer status and concept status of the current query concept respectively. S_L and S_C determine the relation between c_q and the existing semantic concepts, and can be learned based on above relationship definitions through *Algorithm*: *Semantic Status Learning* shown in Fig. 4. When $1 \leqslant S_L \leqslant M$, $1 \leqslant S_C \leqslant N$, c_q is an existing concept c_{S_C}, which is in an existing layer l_{S_L}; when $S_L = M + 1$, $S_C = N + 1$, c_q is a new concept in a new layer; when $1 \leqslant S_L \leqslant M$, $S_C = N + 1$, c_q is a new concept in an existing layer, where l_{S_L} is the lowest layer c_q may be in (assume that the MSR is built from low layers to high layers); and when $S_L = 0$, $S_C = 0$, the semantic status of c_q can not be determined, because the labeled images in $\mathscr{R}$ and $\mathscr{IR}$ are too few.

Then c_q is adaptively added into the MSR through *Algorithm*: *Long-Term MSR Learning* shown in Fig. 5. Each concept c_i in the MSR is represented by a quadruple $\{\mathscr{C}_i, \overline{\mathscr{C}}_i, \mathscr{H}^+, \mathscr{H}^-\}$, where $\mathscr{H}^+$ ($\mathscr{H}^-$) records the counting number of each image $\mathbf{x} \in \mathscr{C}_i$ ($\mathbf{x} \in \overline{\mathscr{C}}_i$) being labeled to be in $\mathscr{C}_i$ ($\overline{\mathscr{C}}_i$). Concepts in the MSR are represented by "relevant" and "irrelevant" image samples. This avoids the ambiguity problem of the representation by keywords.

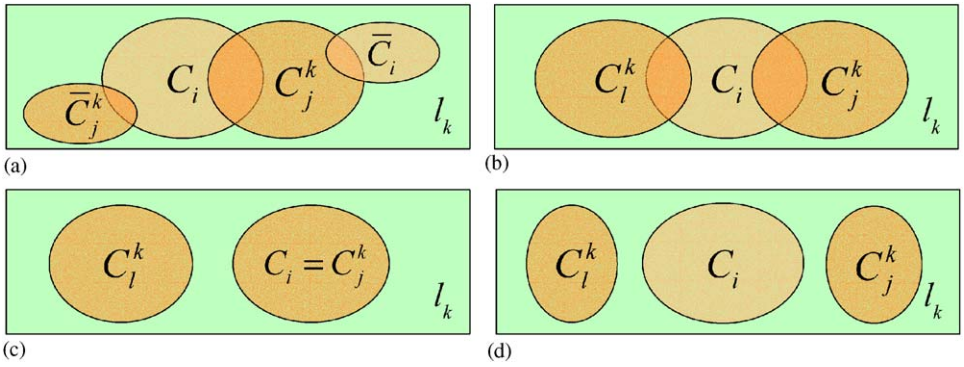

Fig. 3. Relationship between c_i and l_k. (a) $n_0^k > 0$; (b) $n_0^k = 0, n_1^k > 1$; (c) $n_0^k = 0, n_1^k = 1$; (d) $n_0^k = 0, n_1^k = 0$.

Algorithm: Semantic Status Learning

Input: $l_1, \ldots, l_M, \mathcal{C}_1, \ldots, \mathcal{C}_N, \bar{\mathcal{C}}_1, \ldots, \bar{\mathcal{C}}_N, \mathcal{R}, \mathcal{IR}$

Output: Layer status S_L, Concept status S_C

```
1  p_t = Σ_{k=1}^{M} I(Lt(c_q, l_k) = 1);                 layer number compatible with c_q
2  If (p_t = 0) { S_C = N+1; S_L = M+1; }                 new concept in new layer
     Else if (p_t = 1) {
       (1) S_L = {k : Lt(c_q, l_k) = 1};                  existing layer
       (2) If (n_1^{S_L} = 1) {                           existing concept c_i
             S_C = {i : Ct(c_q, c_i) = 1, c_i ∈ l_{S_L}};
           }
           Else { S_C = N+1; }                            new concept in l_{S_L}
     }
     Else {
       (1) m_t = Σ_{Lt(c_q,l_k)=1} n_1^k;                 number of compatible concepts in
                                                          compatible layers
       (2) If (m_t = 0) {                                 new concept in the lowest layer
             S_C = N+1;
             S_L = arg min_k {Lt(c_q, l_k) = 1};
           }
           Else if (m_t = 1) {                            existing concept in existing layer
             S_L = {k : n_1^k = 1};
             S_C = {i : Ct(c_q, c_i) = 1, c_i ∈ l_{S_L}};
           }
           Else { S_L = 0; S_C = 0; }                     status can not be determined
     }
```

Fig. 4. Algorithm for semantic status learning.

2.3.4. Post processing

There are two probable situations where a concept c_i may be mistakenly formed. (1) Images belonging to concept c_i have a great diversity in low-level features, and two subsets of $\mathscr{C}_i$ are independently extracted as $\mathscr{C}_{i1}$ and $\mathscr{C}_{i2}$ asynchronously (during the query session where $\mathscr{C}_{i1}$ is formed, the user carries so few retrieval rounds that images in $\mathscr{C}_{i2}$ are not retrieved at all. Similarly, during the query session where $\mathscr{C}_{i2}$ is formed, images in $\mathscr{C}_{i1}$ are not retrieved). When finally the true $\mathscr{C}_i$ is built, it is in another layer. Fig. 6 (a)

Algorithm: Long-Term MSR Learning:

Input: $S_L,\ S_C$

If $(S_L = M+1)$ { Create l_{M+1}, $c_{N+1} = c_q$, $c_{N+1} \in l_{M+1}$; }
Else if $(1 \le S_L \le M)$ {
  If $(1 \le S_C \le N)$ { $\mathcal{C}_{S_C} = \mathcal{C}_{S_C} \cup \mathcal{R}$; $\overline{\mathcal{C}}_{S_C} = \overline{\mathcal{C}}_{S_C} \cup \mathcal{IR}$; }
  Else { Create $c_{N+1} = c_q$, $c_{N+1} \in l_{S_L}$; }
}
Else { don't record this query; }

Fig. 5. Algorithm for long-term MSR learning.

Algorithm: Concept Merging

Input: $\mathcal{C}_i,\ \mathcal{C}_j$ to be merged

(1) Get $\mathcal{C}_{combine} = \mathcal{C}_i \cup \mathcal{C}_j - \mathcal{C}_i \cap \overline{\mathcal{C}}_j - \overline{\mathcal{C}}_j \cap \mathcal{C}_i$
(2) $\mathcal{C}_k = \mathcal{C}_{combine}$, where $k = \min\{i, j\}$
(3) Set c_k in the lowest compatible layer for $\mathcal{C}_k$.

Fig. 7. Algorithm for concept merging.

gives an example of this case. (2) Some images belonging to concept c_j are *mislabeled* by the user to be in $\overline{\mathscr{C}}_i$, and another concept c_i is formed in another layer. Fig. 6 (b) gives an example of this case. Another probable situation is that some images are judged to be in or not in a concept by different users, because of the subjective difference among different users, and two or more concepts may be formed to represent the same semantic concept. This situation is considered to fall into the second mistake. Note that mislabeling and the subjective difference among users are common problems for all CBIR systems.

Post processing is necessary to alleviate the mistakes and increase the robustness of our system. Here the concept merging technique is used to post process the built MSR. That is, for mistake (1), we merge $\mathscr{C}_{i1}$ and $\mathscr{C}_{i2}$ into $\mathscr{C}_i$; for mistake (2), we merge $\mathscr{C}_i$ and $\mathscr{C}_j$ and remove their contradict images.

- **Feature contrast similarity between concepts**

The *feature contrast model* (FCM) [21] is a psychological similarity measurement, which represents the similarity between two stimuli a, b by

$$S(A, B) = f(A \otimes B) - \alpha f(A \ominus B) - \beta f(B \ominus A),$$

where A, B are the binary features of a, b, respectively. $A \otimes B$ is the *common feature* contained by a and b, $A \ominus B$ ($B \ominus A$) is the *distinctive feature* contained by a but not b (b but not a). $f(\cdot)$ is a salient function, whose value increases monotonically when the variable in the bracket increases.

For a database of size n, if we treat images $\mathbf{x}_1, \ldots, \mathbf{x}_n$ as attributes for semantic concepts, then vector $\mathbf{F}(\mathscr{C}_i) = [I(\mathbf{x}_1 \in \mathscr{C}_i), \ldots, I(\mathbf{x}_n \in \mathscr{C}_i)]$ and vector $\mathbf{F}(\overline{\mathscr{C}}_i) = [I(\mathbf{x}_1 \in \overline{\mathscr{C}}_i), \ldots, I(\mathbf{x}_n \in \overline{\mathscr{C}}_i)]$ can be viewed as binary semantic feature vectors for set $\mathscr{C}_i$ and $\overline{\mathscr{C}}_i$, respectively. With this semantic feature representation, FCM can be used to measure the similarity between c_i and c_j by

$$S(\mathscr{C}_i, \mathscr{C}_j) = f(\mathscr{C}_i \otimes \mathscr{C}_j) - \alpha f(\mathscr{C}_i \ominus \mathscr{C}_j) - \beta f(\mathscr{C}_j \ominus \mathscr{C}_i), \tag{2}$$

where $f(\mathscr{C}_i \otimes \mathscr{C}_j)$ and $f(\mathscr{C}_i \ominus \mathscr{C}_j)$ are given by

$$f(\mathscr{C}_i \otimes \mathscr{C}_j) = \frac{\mathbf{F}(\mathscr{C}_i) \cdot \mathbf{F}(\mathscr{C}_j)}{\min\{\|\mathbf{F}(\mathscr{C}_i)\|^2, \|\mathbf{F}(\mathscr{C}_j)\|^2\}},$$

$$f(\mathscr{C}_i \ominus \mathscr{C}_j) = \frac{\mathbf{F}(\mathscr{C}_i) \cdot \mathbf{F}(\overline{\mathscr{C}}_j)}{\min\{\|\mathbf{F}(\mathscr{C}_i)\|^2, \|\mathbf{F}(\overline{\mathscr{C}}_j)\|^2\}}$$

$\cdot$ is the dot product. If $\alpha \neq \beta$, we emphasize images in c_i and c_j unequally. In our experiment we simply set $\alpha = \beta = 1$.

- **Concepts merging**

When $S(\mathscr{C}_i, \mathscr{C}_j) > \gamma$, $\mathscr{C}_i$ and $\mathscr{C}_j$ are merged by *Algorithm: Concept Merging* shown in Fig. 7. Parameter γ cannot be too small, since in such case some correctly extracted concepts will be removed. In our system γ is statistically set to be 0.8.

By now, the MSR can be automatically built through long-term relevance feedback. In fact no extra burden is added to

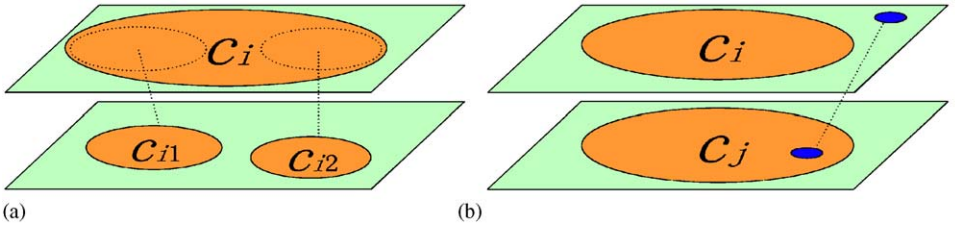

Fig. 6. Two kinds of mistakenly formed concepts. (a) c_{i1} and c_{i2} represent the same semantic concept c_i, but images in $\mathscr{C}_{i1}$ and $\mathscr{C}_{i2}$ are very different in low-level features. In a query session, user queries for an image in $\mathscr{C}_{i1}$, and images in $\mathscr{C}_{i2}$ are not retrieved at all. Then concept c_{i1} is formed. Similarly concept c_{i2} is formed. When finally c_i is formed, it will be in another layer; (b) One image in concept c_j is mislabeled to be "irrelevant" with c_j, and c_i is formed in a different layer from c_j.

system users. As more and more retrievals are carried, the MSR keeps updating, and more and more semantic concepts are extracted and more and more images are contained in the MSR.

3. Hidden annotation with MSR

Since the learned concepts in the MSR reveal real-world semantics with a set of sample images, the semantic meanings of the concepts can be easily understood by human beings. Thus it is convenient for an annotator to find which images in the rest of the image database belong to the concepts, and to associate the images with the concepts (to find all the "relevant" images for each concept from the rest of the database). Note that the concepts do not need to be represented by keyword attributes, and thus can avoid the ambiguity problem of keyword-based representation. This is the simplest HA approach assisted by the long-term learned MSR.

When the number of unlabeled images is very large, it is still laborious to annotate images for every learned concept. Ideally, if the system can automatically find all the *effective images* ("relevant" images) for each concept and provide it to the annotators to annotate, the workload of manual annotation will be greatly reduced. Practically, we can incrementally attain this goal by iterative HA. During each round of HA, the system automatically finds a relatively small number of candidate images for each concept, and gives them to the annotator to annotate. Then the annotation results are added into the semantic knowledge and another round of HA is taken. After enough rounds of HA, all the images will be annotated. Now the problem turns into how to reduce the number of HA rounds, and how to increase the annotation efficiency in each HA round.

In our system more information from the LRF-learned MSR can be exploited to help HA. Within each semantic layer of the MSR, the image database can be hard divided into several clusters, with each cluster representing one hidden semantic concept. Assume that in semantic layer l_k, the whole database $\mathscr{X}$ consists of two parts, the labeled data set $\mathscr{X}_L = \{\mathbf{x}_1^L, \ldots, \mathbf{x}_M^L\}$, and the unlabeled data set $\mathscr{X}_U = \{\mathbf{x}_1^U, \ldots, \mathbf{x}_N^U\}$. The label of $\mathbf{x}_i^L$ is y_i^L, where $y_i^L \in \{1, \ldots, p\}$, and p is the number of extracted hidden concepts in l_k. Our goal is to predict the class label y_j^U of each unlabeled image $\mathbf{x}_j^U$, and select the candidate "relevant" images for this concept for the annotator to annotate. This is a typical multi-label classification problem.

3.1. The semi-supervised learning scheme

For each concept $c_i \in l_k$, we have "relevant" set $\mathscr{C}_i$ and "irrelevant" set $\overline{\mathscr{C}}_i$. The most intuitive approach is the typical supervised learning approach, where we use $\mathscr{C}_i$ and $\overline{\mathscr{C}}_i$ to train a classifier and classify $\mathscr{X}_U$ to get class hypothesis y_j^U. Another approach is the semi-supervised approach, which uses the labeled $\mathscr{X}_L$, together with the assumption of consistency (nearby points are likely to have the same label, and points on the same structure are likely to have the same label), to learn a classifier which is sufficiently smooth with respect to the intrinsic structure revealed by known labeled and unlabeled data. Since the labeled data usually has a small part of the whole database, and might be unrepresentative, the semi-supervised learning approach, which can incorporate information from $\mathscr{X}_U$, is preferred.

There are three requirements for our semi-supervised classification problem. (1) Images in different semantic concepts may cluster in different feature subspaces, a feature selection process is needed to find the representative feature set for each concept. In the representative feature subspace, images in a concept cluster more tightly than in the original feature space, and the consistency assumption for the semi-supervised learning will be more reasonable. (2) There may be many images not belonging to any of the learned concepts, which are *outliers* of existing concepts. These outliers may belong to the concepts not extracted yet, and should be removed before the process of semi-supervised label prediction. (3) The labeled information from users' feedback is precious, and should be fully exploited. Thus in this multi-label classification problem, one-to-many classification is preferred rather than one-to-one classification, because both the "relevant" and the "irrelevant" information is used in the former mode. In summary the entire semi-supervised learning has three parts: representative feature selection, removal of outliers, and semi-supervised label propagation in one-to-many form.

3.2. Representative feature selection

Assume that an image in the database is represented by $\mathbf{x} = [x(1), \ldots, x(d)]$, and $\mathscr{F}^d = \{f_1, \ldots, f_d\}$ is the d-dimensional feature set of the database. For existing concept c_i, the "relevant" set $\mathscr{C}_i = \{\mathbf{x}_1^+, \ldots, \mathbf{x}_m^+\}$ and "irrelevant" set $\overline{\mathscr{C}}_i = \{\mathbf{x}_1^-, \ldots, \mathbf{x}_n^-\}$ are treated as training data, and the *forward sequential feature selection* (FFS) method [22] is used to find the representative feature subset $\mathscr{F}^{d_i}$ for c_i, through *Algorithm: Representative Feature Selection* shown in Fig. 8. The algorithm can be described as follows. Assume that we have already selected k feature axes, and k classifiers have been constructed along each feature axis, respectively. Among all the remaining candidate feature axes, the $k+1$th optimal feature axis is the one, along which a new classifier can be constructed, and the combined classifier by the new classifier and the former k classifiers has the smallest training error. Specifically, the *K-nearest neighbor* (KNN) classifier is adopted as the feature selection classifier. We empirically set MINERR = 0.01 and MAXDIM = 50.

Algorithm: Representative Feature Selection

Input: $\mathcal{C}_i$, $\overline{\mathcal{C}}_i$

Output: Selected feature set $\mathcal{F}^{d_i}$

1. $\mathcal{F}_r = \mathcal{F}^d$; $\mathcal{F}^{d_i} = \phi$;
2. Iteration {
   (1) For each $f_j \in \mathcal{F}^{d_i}$ {
      a. $\mathcal{C}_i^j = \left\{x_1^+(j), \ldots, x_m^+(j)\right\}$; $\overline{\mathcal{C}}_i^j = \left\{x_1^-(j), \ldots, x_n^-(j)\right\}$;
      b. Train K-nearest neighbor classifier $\mathbf{KNN}_j$ by $\mathcal{C}_i^j$ and $\overline{\mathcal{C}}_i^j$
      }
   (2) For each $f_j \in \mathcal{F}_r$ {
      a. $\mathcal{C}_i^j = \left\{x_1^+(j), \ldots, x_m^+(j)\right\}$; $\overline{\mathcal{C}}_i^j = \left\{x_1^-(j), \ldots, x_n^-(j)\right\}$;
      b. Train $\mathbf{KNN}'$ by $\mathcal{C}_i^j$ and $\overline{\mathcal{C}}_i^j$
      c. Get $\mathbf{KNN}_{en}$ by majority voting of $\mathbf{KNN}_1, \ldots, \mathbf{KNN}_{|\mathcal{F}^{d_i}|}$, $\mathbf{KNN}'$
      d. Get leave one out cross validation training error ϵ_j by $\mathbf{KNN}_{en}$
      }
   (3) $k = \underset{j}{\operatorname{argmin}}\, \epsilon_j$, remove f_k from $\mathcal{F}_r$, add f_k into $\mathcal{F}^{d_i}$
   (4) If $\epsilon_k < \text{MINERR}$ or $|\mathcal{F}^{d_i}| > \text{MAXDIM}$, stop; Otherwise go to (1)
   }

Fig. 8. Algorithm for representative feature selection.

3.3. Removal of outliers

Within each semantic layer l_k, for each semantic concept c_i in l_k, we can train a KNN classifier KNN_i based on the projected training set in the representative feature space. Then all the remaining images are classified by KNN_i. Images which are classified to be "relevant" by each KNN_i are added into a big candidate image pool Ψ. Images outside of this image pool is considered to be outliers, i.e. images impossibly belonging to any of the existing hidden concepts in this layer. Outliers are not provided to annotators, which avoids the waste on false annotation. Note that removal of the outliers may cause the problem of false rejection, i.e. some images belonging to c_i may be falsely removed. This problem can be alleviated by iteratively carrying HA. Elimination of outliers can improve the efficiency in each round of HA.

3.4. Semi-supervised label propagation

In order to exploit both the "relevant" and "irrelevant" labeled information, for each concept, the semi-supervised label propagation algorithm is adopted to propagate the label of the labeled images to the candidate unlabeled images.

Within a semantic layer l_k, suppose that there are totally p extracted concepts in l_k. Inside the candidate image pool Ψ, with the consistency assumption, the label of an image $\mathbf{x}_i$ can be predicted by the labels of its neighbors as

$$y_i = \frac{1}{ne_i} \sum_{j \neq i} w_{ij} y_j, \tag{3}$$

where w_{ij} is the weight of each neighbor $\mathbf{x}_j$ of $\mathbf{x}_i$, which is proportional to the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$; ne_i is the number of neighbors for $\mathbf{x}_i$. In the process of label propagation, each point should receive the information from the labels of its neighbors, and adopt the information from the initial label of itself. Assume that after label propagation, $\mathbf{Q}_L = [q_1^L, \ldots, q_M^L]$ denote the predicted labels of labeled data $\mathbf{X}_L = [\mathbf{x}_1^L, \ldots, \mathbf{x}_M^L]$, and $\mathbf{Q}_U = [q_1^U, \ldots, q_N^U]$ denote the predicted labels of unlabeled data $\mathbf{X}_U = [\mathbf{x}_1^U, \ldots, \mathbf{x}_N^U]$. Independently, label propagation is carried for each concept. For c_l, $y_i^L = 1$ ($y_j^U = 1$), if $y_i^L \in \mathscr{C}_l$ ($y_j^U \in \mathscr{C}_l$), and $y_i^L = 0$ ($y_j^U = 0$) otherwise. With the constrain that the labeled data retain their labels after the process, and that for unlabeled data nearby points (in the sense of low-level similarity) have similar labels, an energy function can be given for optimization:

$$E(\mathbf{Y}_U) = \sum_{i=1}^{M} (y_i^L - q_i^L)^2 + \sum_{i,j=1}^{N} w_{ij} (q_i^U - q_j^U)^2. \tag{4}$$

An effective optimization algorithm is given in Ref. [23] to minimize this energy function. Let $(M + N) \times (M + N)$

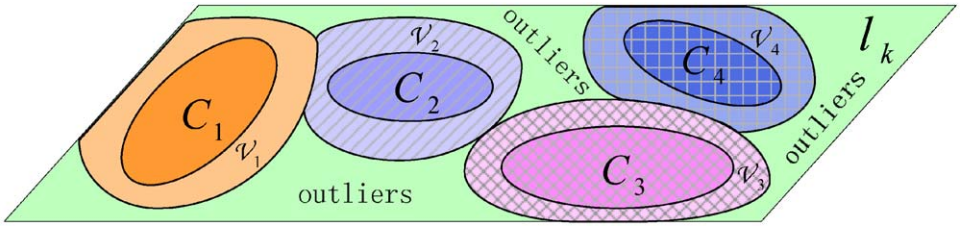

Fig. 9. Candidate $\mathcal{V}_i$ for concept c_i $(i = 1, \ldots, 4)$ in l_k after semi-supervised learning.

matrix $\mathbf{W} = \{w_{ij}\}$ denote the similarity matrix of set Ψ, where the entry w_{ij} is the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$w_{ij} = \exp\left\{-\sum_{k=1}^{d_i} \frac{[x_i(k) - x_j(k)]^2}{\sigma^2}\right\}, \tag{5}$$

where σ is half of the minimal distance between images in the positive sample set and images in the negative sample set. Split the weight matrix $\mathbf{W}$ into the following 4 blocks:

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{LL} & \mathbf{W}_{LU} \\ \mathbf{W}_{UL} & \mathbf{W}_{UU} \end{bmatrix}, \tag{6}$$

the optimum solution of Eq. (4) is given by

$$\begin{aligned} \mathbf{Q}_U &= (\mathbf{T}_{UU} - \mathbf{W}_{UU})^{-1}\mathbf{W}_{UL}\mathbf{Q}_L, \\ \mathbf{Q}_L &= \mathbf{Y}_L, \end{aligned} \tag{7}$$

where $\mathbf{T} = \mathrm{diag}\{t_{ii}\}$ is a diagonal matrix, with $t_{ii} = \sum_j w_{ij}$.

Images with $q_i^U > \theta$ in Ψ are selected as the candidate images for c_i, denoted by $\mathcal{V}_i$. In our experiment we simply set $\theta = 0.5$. After semi-supervised learning, every semantic concept propagates its influence into the unlabeled images, and obtains a set of candidate images. Fig. 9 gives an example for the candidate images of concepts in layer l_k. When the annotator wants to annotate images for concept c_i, the candidate $\mathcal{V}_i$ is provided to him to annotate. Usually the number of the candidate images is much smaller than the whole size of $\mathcal{X}_U$.

4. Experiments

The entire framework of our CBIR system is described in Fig. 10. The system consists of two parts. The part in the red dotted pane (the top part) is the retrieval process within a query session, with long-term MSR learning. The detailed techniques of this part, including the details of the SRF learner and the details of incorporating the SRF learner with the LRF learner, can be found in Ref. [15], which are not verbosely discussed here. The part in the green dashed pane (the bottom part) is the LRF-assisted HA process. In real retrieval, the automatically selected candidate images are provided to the annotator to annotate, and the annotated information are added into the learned MSR through *Algorithm: Long-Term MSR Learning* shown in Fig. 5.

The interface of our system is given in Fig. 11. The left column provides images for the user to query, which are randomly selected from the database. The user can select one as the query image to start a query session. The right main part shows the retrieval result, ordered by the "relevant" degree of images to the query concept. The bottom row gives the images for the user to label.

In the experiment, the image database has 12,000 real world images from the Corel CDs and the Internet, which come from 120 semantic categories, 100 images for each category. The low-level features used are 128-dimensional color coherence in HSV space, 9-dimensional color moment in LUV space, 10-dimensional coarseness vector, and 8-dimensional directionality. Totally 2000 queries are taken to build the MSR and evaluate the algorithms, which consist of two parts. (1) To evaluate the retrieval accuracy, we take 1000 rounds of simulated retrieval based on ground-truth categories to build the *initial MSR* and test the whole system. (2) To evaluate the effectiveness of the proposed method in processing real-world hidden semantics, we ask 10 different users to carry totally 1000 queries to construct the *final MSR* based on the initial MSR. The users have no special training, but are only told to do retrieval without changing the query concept during one query session.

4.1. The built MSR

Since the learned concepts in the MSR reveal real-world semantics, the semantic meanings of the concepts can be very easily understood by human. Thus we can add name to each concept after experiments, such as "eagle", for the sake of easy expression. Representing concepts in the MSR by post-added names, the major structure of the final MSR, which is learned after totally 2000 rounds of simulated and real retrievals, is given in Fig. 12. The figure shows that the final MSR contains 157 concepts in four semantic layers, and has many real-world concepts which are not included in ground-truth semantics. Also, the figure indicates that our proposal can be viewed as a content summarization tool for image databases, and the extracted MSR can be used as the basic content indexes for images. When more images are added into the database, they can be treated as the images

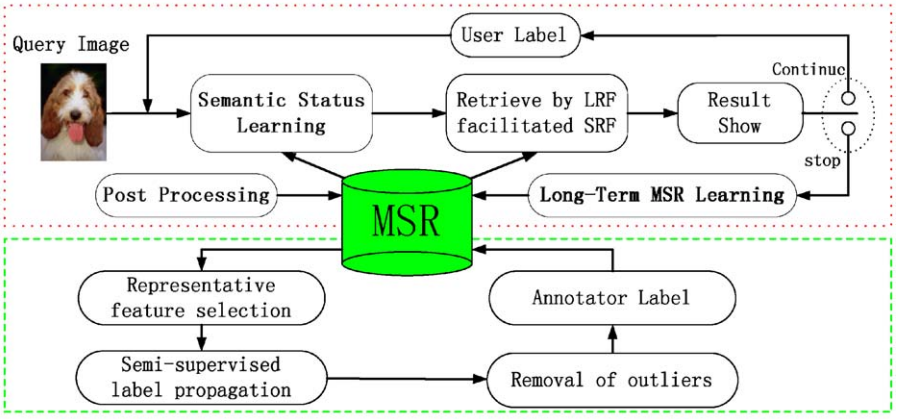

Fig. 10. The entire framework of our CBIR system. The part in the red dotted pane (the top part) is the retrieval process of a query session, with the assistance of MSR long-term information; the part in the green dashed pane (the bottom part) is the HA process assisted by the LRF information.

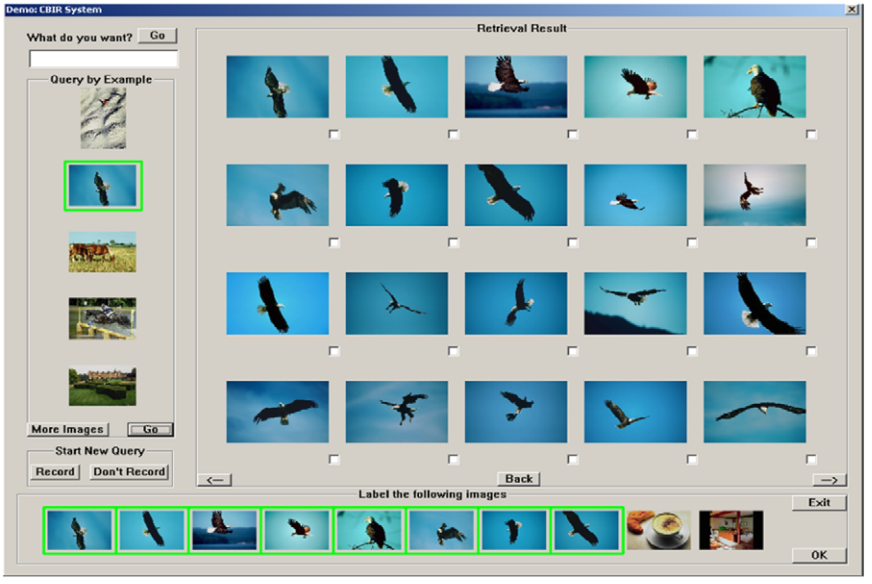

Fig. 11. The retrieval interface of our system. The user can select one of the images in the left column to start a query, and can click the button "More Images" to get more candidate images. The right main part of the interface gives the retrieval result, and the user can click "–>" and "<–" to browse more results. Also the user can click the images in the bottom row to do feedback.

which are never labeled, and the already learned MSR is also scalable to new added data.

4.2. Precision evaluation

The effectiveness of HA assisted by LRF-learned MSR is evaluated by the ground-truth semantics in the initial MSR built after the first 1000 simulated queries. In original HA, the annotator randomly selects images to annotate, and the efficiency of annotation can be measured by the *Precision*:

$$\text{Precision} = \frac{\text{relevant image number in } \mathscr{X}_U}{|\mathscr{X}_U|}.$$

Our system can automatically select candidate images for the annotator to annotate, and can achieve far better efficiency measured by

$$\text{Precision} = \frac{\text{the number of " relevant" images in } \mathscr{V}_i}{|\mathscr{V}_i|}.$$

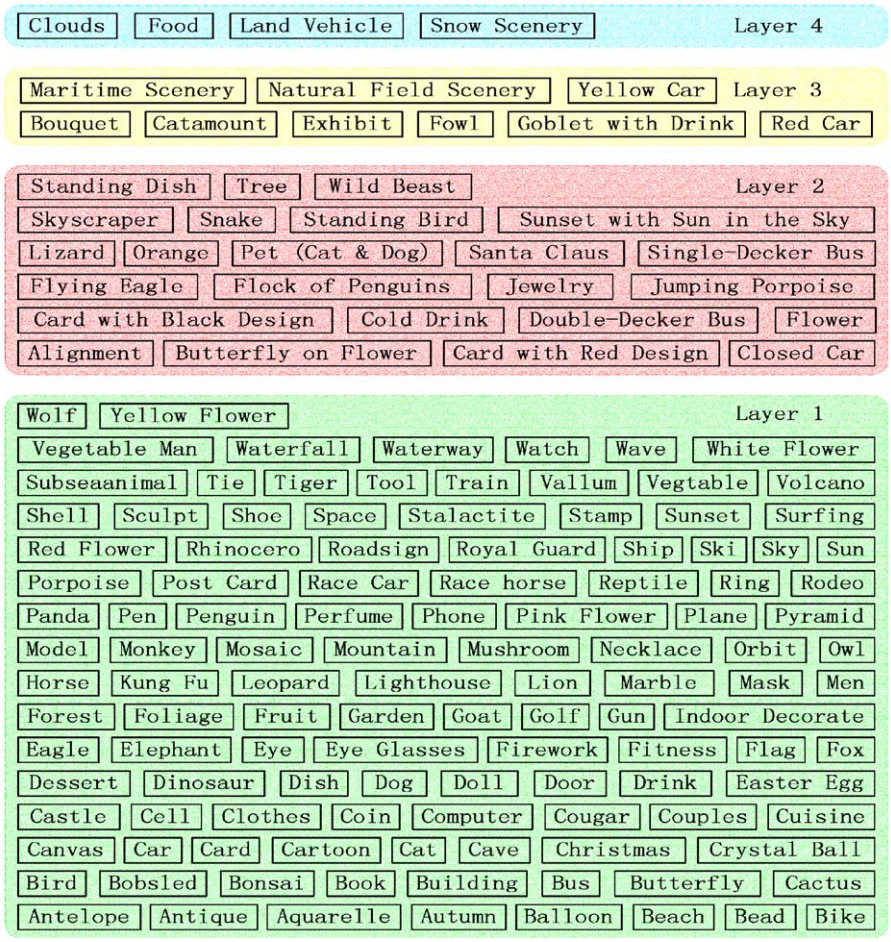

Fig. 12. The MSR built after 2000 rounds of simulated and real retrievals.

That is, for original HA, $\mathscr{V}_i = \mathscr{X}_U$. For each of the ground-truth concepts, we calculate the precision for the first round of HA in our proposal and the random annotation approach, and the average results for these two methods are given in Fig. 13. The figure shows that our system outperforms the random approach significantly. When more and more query sessions are taken, more and more images are labeled, and the Precision of our proposal decreases. This is reasonable because the process to assist HA is expected to be effective when the labeled images have a small part of the database. As for the cases where most images are annotated sufficiently, our approach will be close to the original HA method. Since practically the image database is usually very large, and the labeled images are very few, the advantage of the LRF-assisted HA approach can be expected.

4.3. Comparison with other algorithms

In this part, we use the ground-truth semantics in initial MSR to evaluate the effectiveness of our proposal in bridging the feature-to-concept gap in query-by-example retrieval. Two sets of comparisons are given: the comparison between our proposal with one round of HA and the traditional CBIR mechanism (only using LRF learning); and the comparison between our proposal with one round of HA and other two state-of-the-art LRF methods—He's approach [13] (the LSI approach) and Han's approach [12] (the clustering approach). For fair comparison, all the algorithms use the same SRF learner—the SVM_{Active} learner [20], which almost has the best performance compared with other SRF learners. The performance measurement is the top-k precision:

$$P_{|\mathscr{P}^t|=k} = \frac{\text{the number of “ relevant” images in } \mathscr{P}^t}{|\mathscr{P}^t|},$$

where $\mathscr{P}^t$ is images returned to the user as the retrieval result in the tth feedback round. We randomly select 10 images from each semantic category for querying, and totally carry 1200 independent queries to calculate the average precision. The query images are the same for different

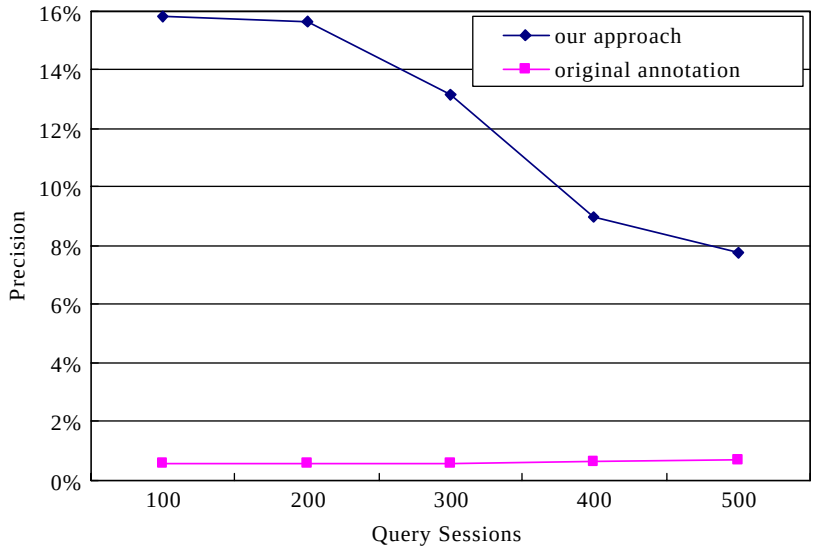

Fig. 13. Precision of the first round of HA, in our method, and in random annotation.

algorithms. In each query session, five rounds of feedback are taken.

• **Comparison with SRF learning**

The average P_{20} of our proposal and that of the SVM$_{Active}$ algorithm with different cumulate experiences (from 100 to 1000 query sessions) are given in Fig. 14, where the results for SVM$_{Active}$ correspond to the points for zero query sessions accumulation. The figure shows that our system can improve the retrieval performance consistently and significantly from the second round of retrieval. For example, the improvement of P_{20} after 1000 query sessions in round five is 50.58%. As more and more query sessions are carried, the advantage of our method is more and more obvious.

• **Comparison with other LRF algorithms**

Fig. 15 gives the average P_{20} of our algorithm and the other two LRF algorithms with different cumulate experiences (after 100, 300, 500 and 1000 query sessions' learning). The SVD semantic space of He's method [13] has a dimensionality of 60, and the important weight w for semantic similarity in Han's method [12] is 0.7. The figure shows that our LRF-assisted HA system outperforms the other two LRF methods, consistently from the first feedback round. For example the precision improvements for feedback round five after 1000 queries are 18.67% and 14.72% compared with He's approach and Han's approach respectively. As for He's method and Han's method, when more query sessions are carried, the latter one obviously has more advantages. The phenomena of this experiment can be explained as follows. As discussed in Section 2.1, He's method does not explicitly extract meaningful semantic concepts from the recorded semantic information, while Han's method and our long-term MSR learning approach both try to learn semantic concepts further, which could be expected to have better performance. On another aspect, Han's approach does not exploit precise multi-correlation among images, and the extracted semantic clusters usually inaccurately describe real-world concepts. Since our method reveals real-world semantics, the advantage can also be expected.

4.4. Evaluation of semi-supervised learning

We evaluate the effectiveness of the semi-supervised learning algorithm in selecting candidate images for labeling. We compare the candidate images selected by our proposal with those selected by the supervised SVM classifier, which has better performance for our small sample learning problem than most other supervised classifiers. The performance measurements are the precision and the recall: for concept c_i

$$\text{precision} = \frac{\text{the number of “ relevant” images in } \mathscr{V}_i}{|\mathscr{V}_i|},$$

$$\text{recall} = \frac{\text{the number of “ relevant” images in } \mathscr{V}_i}{\text{the number of “ relevant” images in } \mathscr{X}_U}.$$

Fig. 16 gives the average precision and recall in the first round of HA with different cumulate experiences (from 100 to 500 query sessions) of our method and those of the supervised SVM classifier. The SVM classifier directly uses the labeled images as training set to construct the classifier to predict the class label of unlabeled images, and selects the "relevant" predicted ones as candidate images. The figure shows that our semi-supervised learning mechanism outperforms the SVM classifier consistently in both precision and recall, and provides more effective images. The result

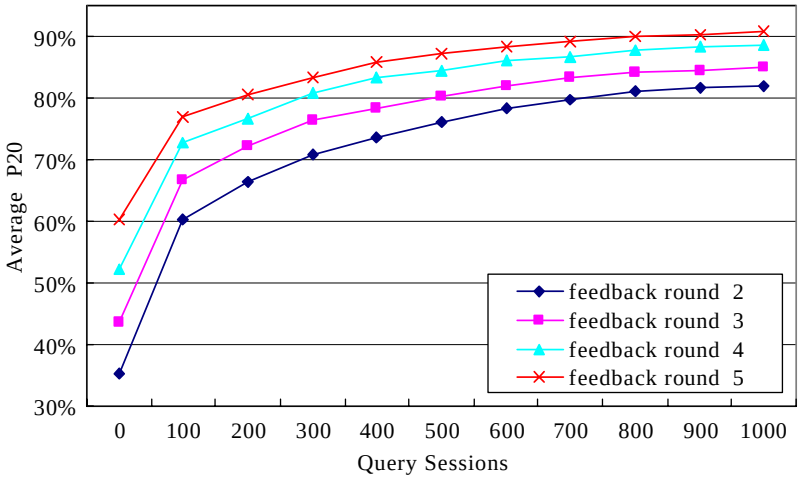

Fig. 14. Comparison of our method and the short-term SVM$_{Active}$ learner, with different cumulate long-term experiences. The result for SVM$_{Active}$ classifier corresponds to the points with zero query sessions accumulation.

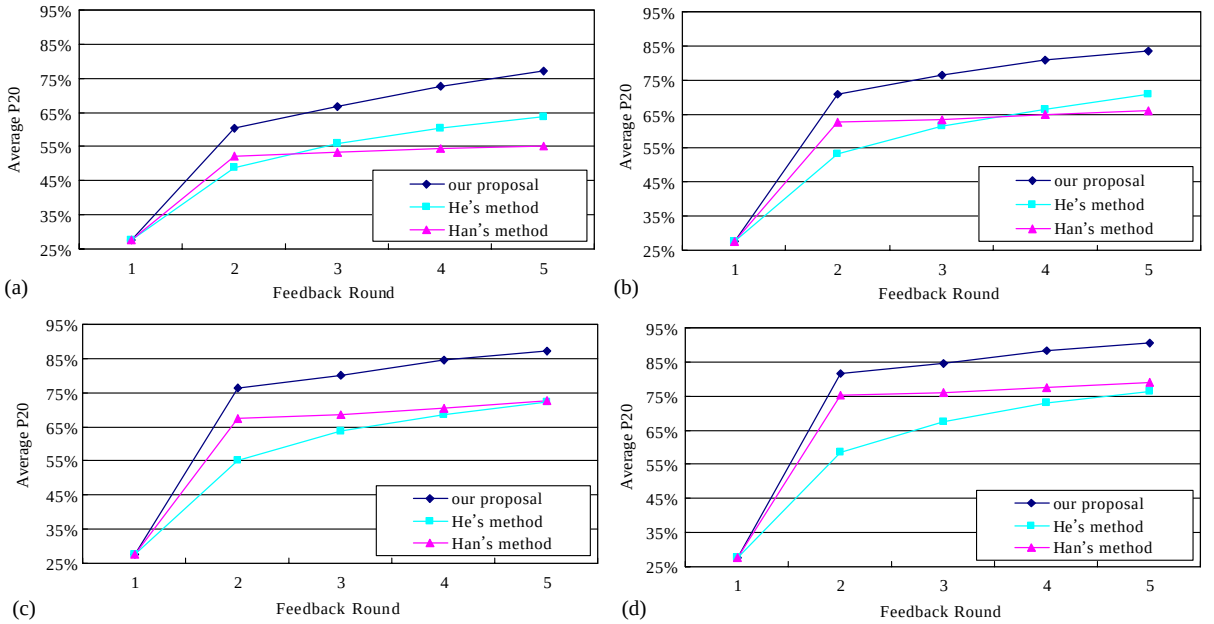

Fig. 15. Comparison of our method and He's method and Han's method in five rounds of feedback, with: (a) 100 query sessions accumulation; (b) 300 query sessions accumulation; (c) 500 query accumulation; (d) 1000 query sessions accumulation.

is consistent with our discussion in Section 3. Our semi-supervised learning uses the structure information provided by the unlabeled data, while the SVM classifier does not, and good performance of our approach can be expected.

5. Conclusion

In this paper we address the issue of effective HA for image retrieval. Through the LRF learning process, users' feedback information is exploit to extract the MSR for the image database, which reflects the real-world semantics underlying images. The MSR confines a set of semantic concepts, which are adaptively extracted and are related to the image database. These concepts are provided to the annotators for annotation instead of the thesaurus and the pre-confined keywords, which both alleviates the burden of manual annotation and increases the adaptivity of our HA system to previously unknown databases. Further more, the concepts are represented by image samples, which avoids the ambiguity problem of the keyword-based representation. For

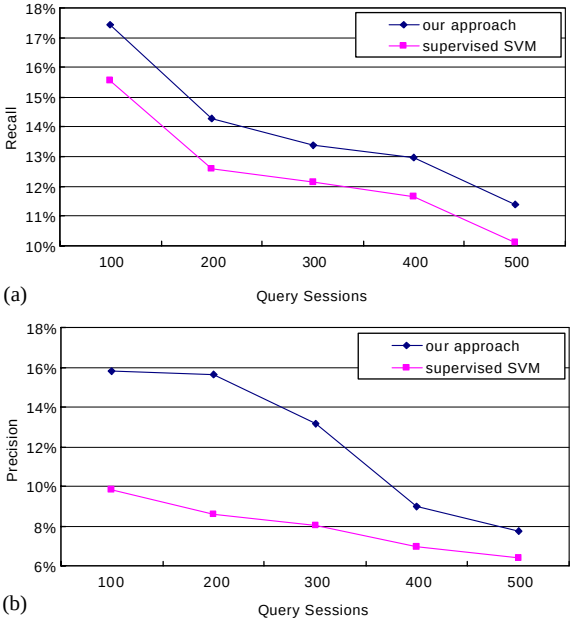

Fig. 16. Comparison of our method and the supervised SVM classifier.

each learned concept, a semi-supervised learning mechanism is adopted to automatically select a relatively small number of candidate images for the annotator to annotate, which increases the efficiency of HA. The proposed CBIR system seamlessly combines LRF with the HA mechanism to both alleviate the burden of manual annotation and bridge the gap between high-level semantic concepts and low-level features, and thus improve the retrieval performance.

Based on the MSR, more information can be extracted to reveal the relationship between images and the concepts they belong to, and to find the mapping between low-level features and high-level semantic concepts. For example, offline feature extraction and feature selection can be carried. More work will be done on these issues. Moreover, since currently our approach is data driven (it is effective for labeling images in a given database), it will be interesting to explore the ability to generalize from these labels in the future.

Acknowledgements

This work is supported by key project of National Natural Science Foundation of China (No. 60432030).

References

[1] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, Content-based image retrieval at the end of the early years, IEEE Trans. Pattern Anal. Mach. Intell. 22 (12) (2002) 1349–1380.

[2] I.J. Cox, J. Ghosn, T.V. Papathomas, P.N. Yianilos, Hidden annotation in content based image retrieval, IEEE Workshop on Content-Based Access of Image and Video Libraries, 1997, pp. 76–81.

[3] L. Gregory, J. Kittler, Using a pictorial dictionary as a high level user interface for visual information retrieval, IEEE Proceedings of the International Conference on Image Processing, vol. 2, 2003, pp. 519–522.

[4] R.W. Picard, T.P. Minka, Vision Texture for Annotation, ACM/Springer, Berlin, J. Multimedia Systems 3 (1995) 3–14.

[5] J. Yang, W.Y. Liu, H.J. Zhang, Y.T. Zhung, Thesaurus-aided approach for image browsing and retrieval, IEEE Proceedings of the International Conference on Multimedia and Expo, 2001, pp. 1135–1138.

[6] C. Zhang, T.H. Chen, Annotating retrieval database with active learning, IEEE Proceedings of the International Conference on Image Processing, vol. 2, 2003, pp. 595–598.

[7] X.S. Zhou, T.S. Huang, Unifying keywords and visual contents in image retrieval, IEEE Multimedia 9 (2) (2002) 23–33.

[8] X.Q. Zhu, W.Y. Liu, H.J. Zhang, L.D. Wu, An image retrieval and semi-automatic annotation scheme for large image databases on the Web, Proceedings of SPIE Symposium on Electronic Imaging-EI24 Internet Imaging II, vol. 4311, 2001, pp. 168–177.

[9] X.S. Zhou, T.S. Huang, Relevance feedback for image retrieval: a comprehensive review, Multimedia Systems 8 (6) (2003) 536–544.

[10] B. Bhanu, A.L. Dong, Concept learning with fuzzy clustering and relevance feedback, Eng. Appl. Artif. Intell. 15 (2002) 123–138.

[11] A.L. Dong, B. Bhanu, A new semi-supervised EM algorithm for image retrieval, IEEE Proceedings of the International Conference on Computer Vision and Pattern Recognition, 2003, pp. 662–667.

[12] J.W. Han, M.J. Li, H.J. Zhang, L. Guo, A memorization learning model for image retrieval, IEEE Proceedings of the International Conference on Image Processing, vol. 3, 2003, pp. 605–608.

[13] X.F. He, O. King, W.Y. Ma, M.J. Li, H.J. Zhang, Learning a semantic space from user's relevance feedback for image retrieval, IEEE Trans. Circuits Syst. Video Technol. 13 (1) (2003) 39–48.

[14] D.R. Heisterkamp, Building a latent semantic index of an image database from patterns of relevance feedback, IEEE Proceedings of the International Conference on Pattern Recognition, vol. 4, 2002, pp. 134–137.

[15] W. Jiang, G.H. Er, Q.H. Dai, Multi-layer semantic representation learning for image retrieval, IEEE Proceedings of the International Conference on Image Processing, vol. 4, 2004, pp. 2215–2218.

[16] C.S. Lee, W.Y. Ma, H.J. Zhang, Information Embedding Based on User Relevance Feedback for Image Retrieval, Proceedings of SPIE, Multimedia Storage and Archiving Systems IV, vol. 3846, 1999, pp. 294–304.

[17] M.J. Li, Z. Chen, H.J. Zhang, Statistical correlation analysis in image retrieval, Pattern Recognition 35 (2002) 2687–2693.

[18] K. Markus, L. Jorma, Using long-term learning to improve efficiency of content-based image retrieval, Third International Workshop on Pattern Recognition in Information Systems, Angers, France, 2003, pp. 72–79.

[19] T.S. Huang, X.S. Zhou, M. Nakazato, I. Cohen, Y. Wu, Learning in content-based image retrieval, IEEE Proceedings of the International Conference on Development and Learning, 2002, pp. 155–164.
[20] S. Tong, E. Chang, Support vector machine active learning for image retrieval, Proceedings of ACM International Multimedia Conference, 2001, pp. 107–118.
[21] A. Tvesky, Feature of similarity, Psychol. Rev. 84 (4) (1977) 327–352.
[22] K. Fukunaga, Introduction to Statistical Pattern Recognition, Second ed., Academic Press, New York, USA, 1990.
[23] X.J. Zhu, Z. Ghahramani, J. Lafferty, Semi-supervised learning using Gaussian fields and harmonic functions, Proceedings of the 20th International Conference on Machine Learning, ACM Press, 2003.

About the Author—WEI JIANG received the B.S. degree in Automation Department from Tsinghua University, China in 2002. She is currently pursuing the M.S. degree in Automation Department at Tsinghua University. Her research interests include content-based image retrieval/management, pattern recognition, image processing.

About the Author—GUIHUA ER received B.S. degree in Automation Department from Tianjin University, China, in 1984, and M.S. degree in Automation Department from Beijing Institute of Technology, China, in 1989. She is now an associate processor and the vice director of Broadband Networks & Digital Media Lab in Automation Department, Tsinghua University, China. Her research interests include multimedia database and multimedia information coding.

About the Author—QIONGHAI DAI received B.S. degree in Mathematics from Shanxi Normal University, China, in 1987, and M.E. and Ph.D. degrees in Computer Science and Automation from Northeastern University, China, in 1994 and 1996, respectively. After being a Postdoctoral Research in Automation Department, he has been with Media Lab of Tsinghua University, China, where he is currently an Associate Professor and Head of the Lab. His research interests are in signal processing, broadband networks, video processing and communication.

About the Author—JINWEI GU received the B.S. degree in Automation Department from Tsinghua University, China in 2002. Now he is a master student in Automation Department, Tsinghua University. His research interests are in pattern recognition, computer vision and intelligent information processing.