Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 79412, Pages 1–10
DOI 10.1155/ASP/2006/79412

Manifold-Ranking-Based Keyword Propagation for Image Retrieval

Hanghang Tong,[1] Jingrui He,[1] Mingjing Li,[2] Wei-Ying Ma,[2] Hong-Jiang Zhang,[2] and Changshui Zhang[1]

[1] *Department of Automation, Tsinghua University, Beijing 100084, China*
[2] *Microsoft Research Asia, 49 Zhichun Road, Beijing 100080, China*

Received 30 August 2004; Revised 29 January 2005; Accepted 5 April 2005

A novel keyword propagation method is proposed for image retrieval based on a recently developed manifold-ranking algorithm. In contrast to existing methods which train a binary classifier for each keyword, our keyword model is constructed in a straightforward manner by exploring the relationship among all images in the feature space in the learning stage. In relevance feedback, the feedback information can be naturally incorporated to refine the retrieval result by additional propagation processes. In order to speed up the convergence of the query concept, we adopt two active learning schemes to select images during relevance feedback. Furthermore, by means of keyword model update, the system can be self-improved constantly. The updating procedure can be performed online during relevance feedback without extra offline training. Systematic experiments on a general-purpose image database consisting of 5 000 Corel images validate the effectiveness of the proposed method.

1. INTRODUCTION

The initial image retrieval methods are based on keyword annotation and can be traced back to the 1970's [1, 2]. In such approaches, images are first annotated manually with keywords, and then retrieved by their annotations. As long as the annotation is accurate and complete, keywords can accurately represent the semantics of images. However, it suffers from several main difficulties, for example, the large amount of manual labor required to annotate the whole database, and the inconsistency among different annotators in perceiving the same image [3]. Moreover, although it is possible to extract keywords for Web images from their surrounding text, such extraction might be far from accurate and complete [4].

To overcome these difficulties, an alternative scheme, content-based image retrieval (CBIR) was proposed in the early 1990's, which makes use of low-level image features instead of the keyword features to represent images, such as color [5–7], texture [8–10], and shape [11, 12]. Its advantage over keyword-based image retrieval lies in the fact that feature extraction can be performed automatically and the image's own content is always consistent [4]. Despite the great deal of research work dedicated to the exploration of an ideal descriptor for image content, its performance is far from satisfactory due to the well-known gap between visual features and semantic concepts, that is, images of dissimilar semantic content may share some common low-level features, while images of similar semantic content may be scattered in the feature space [4].

In order to narrow or bridge the gap, a great deal of work has been performed in the past years, such as exploring more powerful low-level feature representation, seeking for more suitable metric for perceptual similarity measurement [4] and so on. Furthermore, many efforts have been made to efficiently utilize the strengths of both keyword-based and content-based methods in image retrieval. Those methods can be categorized into: online, offline, and their combination [13–15].

Most, if not all, of existing online methods make use of relevance feedback (RF). For example, Lu et al. proposed in [15] using a semantic network and relevance feedback based on visual features to enhance keyword-based retrieval and update the association of keywords with images. Among others, a key issue in relevance feedback is the learning strategy. One of the most effective learning techniques used in RF is support vector machines (SVM) [16], which aims to create a classifier that separates the relevant and irrelevant images and generalizes well on unseen examples. Furthermore, to speed up the convergence to the target concept, active learning methods, such as SVM$_{\text{active}}$ [17], are also utilized to select the most informative images. However, one major problem with SVM and SVM$_{\text{active}}$ is the insufficiency of labeled examples, which might bring great degradation to the performance of the trained classifier.

On the other hand, Chang et al. in [13] proposed an offline method to perform keyword propagation based on classification. In their work, starting from a small portion of manually labeled images in the database, an ensemble of binary classifiers was trained for multiple soft annotations, which in turn assists a user to find relevant images rapidly via keyword. Jing et al. in [14] further extended this work in: (1) combining relevance feedback to refine the retrieval result; and (2) introducing labeling vector to online collect training samples and to off-line update the keyword models. However, there still exist some limitations and drawbacks: (1) their method will not work if only positive example is provided in relevance feedback; (2) the way they combine the information from relevance feedback is somewhat heuristic; (3) active learning is not considered in relevance feedback; (4) the ratio of initial manually labeled images is relatively high (ten percent in their experiment), which is still a heavy burden especially when the database is large.

In this paper, we propose a novel method to support keyword propagation for image retrieval based on a recently developed manifold-ranking algorithm [18, 19]. This work is motivated by our previous success in applying this algorithm in the scenario of query by example (QBE) [4], and can be viewed as its counterpart in the scenario of query by keyword (QBK). Firstly, in our method, the keyword model is constructed by exploring the relationship among all the images in the feature space, in contrast to inductive methods which only use the labeled images to train an ensemble of binary classifiers [13, 14]. Secondly, our method provides a very natural way to incorporate the information from relevance feedback to refine the retrieval result. Moreover, to maximally improve the performance of propagation process, active learning is investigated to select images in relevance feedback. Finally, our method also supports accumulation and memorization of knowledge learnt from user-provided relevance feedback by means of keyword model update. Different from [14], in which an extra offline training procedure is needed, our update procedure can be performed online during relevance feedback sessions.

The manifold-ranking algorithm [18, 19] is initially proposed to rank the data points or to predict the labels of unlabeled data points along their underlying manifold by analyzing their relationship in Euclidean space. In [4], we introduced it in the scenario of QBE and found out that by incorporating unlabeled data in the learning process and exploring their relationship with labeled data, this method outperforms existing classification-based ones (such as SVM) by a large margin. Motivated by this, we further apply it to keyword propagation in this paper, hoping that it will still outperform existing methods in the scenario of QBK. Like in [4], the algorithm first constructs a weighted graph using each data point as a vertex. Next, the keyword model is initialized as a keyword matrix with positive scores of the labeled images in the corresponding positions. Then these scores are iteratively propagated to nearby points via the graph. Finally, each image in the database will be given a score vector, the element of which indicates the relevance of the given image to the corresponding keyword (a larger score indicating higher relevance). By ranking all the images according to their relevance to a given keyword, it can assist a user to find relevant images quickly via keywords.

In relevance feedback, if the user only marks relevant examples, they serve as the new query set with respect to the query keyword, and their influence can be calculated by an additional propagation process. On the other hand, if both relevant and irrelevant examples are available, their influence will be propagated, respectively, in different manners: the effect of negative examples is suppressed due to the asymmetry between relevant and irrelevant images. This online information, when combined with the initial keyword model, will help to improve the retrieval result.

To maximally improve the propagation performance, active learning can be also incorporated in relevance feedback. To be specific, we will examine two different schemes developed in [4]: (1) to select the most positive images; and (2) to select the most positive and inconsistent images.

Another important issue with keyword propagation is how to accumulate and memorize knowledge learnt from user-provided relevance feedback so that the retrieval system can be self-improved constantly. To achieve this goal, the keyword model should be updated periodically. By careful analysis, we reach the conclusion that in our method, such update procedure can be performed online during relevance feedback so that no extra offline training is needed.

The organization of the paper is as follows. In Section 2, we briefly review the two versions of manifold-ranking algorithm and its application to image retrieval in the scenario of QBE. We describe the construction of the keyword model using manifold-ranking algorithm with some analysis in Section 3. In Section 4, the initial retrieval and following feedback process of QBK scenario is presented, and the active learning schemes are also discussed. The keyword model updating is addressed in Section 5. In Section 6, we provide systematic experimental results. Finally, we conclude the paper in Section 7.

2. RELATED WORK

2.1. Manifold-ranking algorithm

The manifold-ranking algorithm is a semisupervised learning algorithm which explores the relationship among all the data points [18, 19]. It has two versions for different tasks: to rank data points and to predict the labels of unlabeled data points.

For the ranking task, it can be formulated as: given a set of points $\chi = \{x_1, \ldots, x_q, x_{q+1}, \ldots, x_n\} \subset \mathbb{R}^m$, the first q points are the queries which form the query set; the remaining points are to be ranked according to their relevance to the queries.

Let $d : \chi \times \chi \rightarrow \mathbb{R}$ denote a metric on χ which assigns each pair of points x_i and x_j a distance $d(x_i, x_j)$, and $f : \chi \rightarrow \mathbb{R}$ denote a ranking function which assigns to each point x_i a ranking score f_i. Finally, we define a vector $y = [y_1, \ldots, y_n]^T$ corresponding to the query set, in which $y_i = 1$ if x_i is a query, and $y_i = 0$ otherwise. The procedure of ranking the data points in [19] can be given as follows.

```
(1) Sort the pair-wise distances among points in ascending order. Repeat connecting the two points with an edge according to the order until a connected graph is obtained.
(2) Form the affinity matrix $W$ defined by $W_{ij} = \exp[-d^2(x_i, x_j)/2\sigma^2]$ if there is an edge linking $x_i$ and $x_j$. Let $W_{ii} = 0$.
(3) Symmetrically normalize $W$ by $S = D^{-1/2}WD^{-1/2}$ in which $D$ is the diagonal matrix with $(i, i)$-element equal to the sum of the $i$th row of $W$.
(4) Iterate $f(t + 1) = \alpha S f(t) + (1 - \alpha)y$ until convergence, where $\alpha$ is a parameter in $[0, 1)$ and $f(0) = y$.
(5) Let $f^*$ denote the limit of the sequence $\{f(t)\}$. Rank each point $x_i$ according to its ranking scores $f_i^*$.
```

Algorithm 1: Ranking data points.

For the task of predicting the labels of unlabeled data points, it can be formulated as: given a set of points $\chi = \{x_1, \dots, x_l, x_{l+1}, \dots, x_n\} \subset \mathbb{R}^m$ and a label set $\zeta = \{1, \dots, c\}$, the first l points x_i $(i \leq l)$ are labeled as $y_i \in \zeta$; and the remaining points x_u $(l + 1 \leq u \leq n)$ are to be labeled.

Define an $n \times c$ matrix F corresponding to a classification on the dataset χ by labeling each point x_i with $y_i = \arg\max_{j \leq c} F_{i,j}$. We also define an $n \times c$ matrix $Y = [Y_1, \dots, Y_c]$ with $Y_{i,j} = 1$ if x_i is labeled as $y_i = j$ and $Y_{i,j} = 0$ otherwise. The procedure of predicting labels is quite similar with that of ranking the data points [18].

An intuitive description of the above two algorithms is: a weighted graph is first formed which takes each data point as a vertex; a positive score is assigned to each query while zero to the remaining points; all the points then spread their scores to the nearby points via the weighted graph; the spread process is repeated until a global stable state is reached, and all the points will have their own scores according to which they will be ranked or to be labeled.

2.2. Application for image retrieval in the scenario of QBE

In [4], we have applied Algorithm 1 to image retrieval in the scenario of QBE. Its key points are summarized as follows.

(i) In the initial query stage in the scenario of QBE, there is only one query in the query set. The resultant ranking score of an unlabeled image is in proportion to the probability that it is relevant to the query, with large ranking score indicating high probability.

(ii) In relevance feedback, if the user only marks relevant examples, the algorithm can be easily generalized by adding these newly labeled images into the query set; on the other hand, if examples of both labels are available, they are treated differently: relevant images are also added to the query set, while for irrelevant images, we designed three schemes based on the observation that positive examples should make more contribution to the final ranking score than negative ones.

(iii) To maximally improve the ranking result, we also developed three active learning methods for selecting images in each round of relevance feedback. Namely, (1) to select the most positive images; (2) to select the most informative images; and (3) to select the most positive and inconsistent images.

```
(1–3) The same as Algorithm 1.
(4)   Iterate $F(t + 1) = \alpha SF(t) + (1 - \alpha)Y$ until convergence, where $\alpha$ is a parameter in $[0, 1)$ and $F(0) = Y$.
(5)   Let $F^*$ denote the limit of the sequence $\{F(t)\}$. Label each point $x_i$ with $y_i = \arg\max_{j \leq c} F^*_{i,j}$.
```

Algorithm 2: Predicting labels.

3. KEYWORD MODEL CONSTRUCTION

3.1. Notation

Our keyword model is actually an $n \times c$ matrix $F = [F_1, \dots, F_c]$, where n is the total number of images in the database and c is the total number of keywords. Each image in the database corresponds to a row and each keyword corresponds to a column. The element $F_{i,q}$ $(i = 1, \dots, n;\ q = 1, \dots, c)$ of the keyword model denotes the relevance of image x_i to keyword K_q (large value indicating high relevance).

3.2. The keyword propagation process

To construct such keyword model, we need to manually label a small portion of images in the database, and then propagate their labels (keywords) to the unlabeled ones. It can be seen that Algorithm 2 can perform this task well. However, we will make some modifications as follows.

(i) Multilabels for a single image are supported. If an image is given more than one keyword in the manually labeling stage, all the corresponding elements in Y are assigned 1.

(ii) The weighted graph in step (1) is constructed as: calculate the K nearest neighbors for each point; connect two points with an edge if they are neighbors. The reason for this modification is to ensure enough connection for each point while preserving the sparse property of the weighted graph.

(iii) Since $L1$ distance can better approximate the perceptual difference between two images than other popular Minkowski distances when using either color or texture representation or both [4], it is adopted to define the edge weights in W,

$$W_{ij} = \prod_{l=1}^{m} \exp\left(- \left| x_{il} - x_{jl} \right| / \sigma_l \right), \tag{1}$$

where x_{il} and x_{jl} are the lth dimension of x_i and x_j, respectively; m is the dimensionality of the feature space; and σ_l is a positive parameter that reflects the scope of different dimensions.

(iv) Step (5) in Algorithm 2 is ignored for the purpose of soft annotation.

3.3. Analysis

We make a short analysis of the keyword propagation process by Algorithm 2 with respect to its transductive learning; multiranking and incremental learning nature.

3.3.1. Transductive learning nature

The theorem in [18] guarantees that the sequence $\{F(t)\}$ converges to (from now on, we will omit the mark "$*$")

$$F = \beta(1 - \alpha S)^{-1} Y, \tag{2}$$

where $\beta = 1 - \alpha$. Although F can be expressed in a closed form, for large scale problems, the iteration algorithm is preferable due to computational efficiency. Using Taylor expansion and omitting the constant coefficient β, we have

$$\begin{aligned} F &= (I - \alpha S)^{-1} Y = (I + \alpha S + \alpha^2 S^2 + \cdots) Y \\ &= Y + \alpha S Y + \alpha S(\alpha S Y) + \cdots . \end{aligned} \tag{3}$$

From the above equation, we can grasp the idea of the algorithm from a transductive learning point of view. F can be regarded as the sum of a series of infinite terms. The first term is simply the score of initial labels Y, the second term is to spread the scores of the initial labeled images to their nearby points, the third term is to further spread the scores, and so on. Thus the effect of unlabeled image is gradually incorporated.

Different from existing methods, such as [13, 14], in which keyword propagation is performed by training an ensemble of binary classifiers, in our method, it is performed in a much more straightforward way. While those inductive methods aim to train a classifier using labeled images which generalizes well on unlabeled images, our method is a transductive method and explores the unlabeled images in the learning stage. By doing so, we hope it will outperform the existing inductive ones.

3.3.2. Multiranking nature

Since $Y = [Y_1, \ldots, Y_c]$, and $F = [F_1, \ldots, F_c]$, the following fact will hold:

$$F_q = \beta(1 - \alpha S)^{-1} Y_q, \quad (q = 1, \ldots, c). \tag{4}$$

Define the initial query set Q^q for each keyword K_q: if a given image is labeled as keyword K_q, it is added into Q^q $(q = 1, \ldots, c)$. It can be seen that Y_q is the corresponding vector as defined in Algorithm 1 for Q^q. By doing so, we make a bridge between the two versions of manifold-ranking algorithm. The keyword propagation by Algorithm 2 can be viewed as a multiranking process: each keyword has its own initial query set; propagates its influence by step (4) of Algorithm 1 independently; and combines the results altogether.

3.3.3. Incremental learning nature

Here, we explore the incremental learning nature of the keyword propagation process. Since it can be viewed as a multiranking process, we only focus on one specific keyword K_q.

Let Q^q and Y_q be the initial query set and the corresponding vector, respectively. The ranking vector F_q can be computed as (4). Suppose that we get some new labeled examples for K_q. Let these examples compose a new query set Q^{new} and define its corresponding vector y^{new}. Adding Q^{new} into Q^q, we get a combined query set Q^{com} and its corresponding vector y^{com}. The ranking vector with respect to K_q should be updated by rerunning Algorithm 1 on Q^{com}, and the sequence $\{f^{\text{com}}(t)\}$ converges to

$$f^{\text{com}} = \beta(I - \alpha S)^{-1} y^{\text{com}}. \tag{5}$$

Note that $Q_{\text{com}} = \{Q^q, Q^{\text{new}}\}$, and $y_{\text{com}} = Y_q + y^{\text{new}}$. Thus, (5) can be converted to

$$f^{\text{com}} = \beta(I - \alpha S)^{-1} Y_q + \beta(I - \alpha S)^{-1} y^{\text{new}} = F_q + f^{\text{new}}, \tag{6}$$

where $f^{\text{new}} = \beta(I - \alpha S)^{-1} y^{\text{new}}$.

It can be seen from the above equation that the algorithm provides a natural way to incorporate the newly labeled examples: propagate their influence and simply add the result into the original ranking vector.

4. QUERY BY KEYWORD

4.1. Initial retrieval result

After the keyword model is constructed, each image x_i $(i = 1, \ldots, n)$ in the database corresponds to a row in the matrix, indicating its relevance to different keywords; while each keyword K_q corresponds a column in the matrix $F_q = [F_{1,q}, \ldots, F_{n,q}]^T$, indicating the relevance of different images to that keyword. Thus, the similarity score of image x_i with respect to the query keyword K_q can be expressed as

$$S_i = F_{i,q}, \quad i \in \{1, \ldots, n\};\ q \in \{1, \ldots, c\}. \tag{7}$$

The initial retrieval result is given by sorting the images in the decreasing order of their similarity scores.

As point out in [13], when the query is not in the keyword set, query expansion is needed to translate the initial query. However, we will skip the details of this issue in this paper.

4.2. Relevance feedback

Benefited from its incremental learning nature, our method provides a natural way to incorporate the additional information from users to refine the similarity score in relevance feedback.

For a query keyword K_q, its initial ranking vector is F_q. Let all examples from users' feedback compose two new query sets: Q^+ for positive examples and Q^- for negative ones. We also define their corresponding vectors y^+ and y^- as Algorithm 1, except that the element of y^- is set to -1 if the corresponding image is a negative example. Using Algorithm 1, the effect of these two query sets can be written as

$$f^+ = \beta(I - \alpha S)^{-1} y^+,$$
$$f^- = \beta(I - \alpha S)^{-1} y^-. \quad (8)$$

By the incremental learning nature we analyzed in Section 3.3.3, the similarity score of image x_i with respect to K_q is updated as

$$S_i = F_{i,q} + f_i^+ + \gamma f_i^-, \quad i \in \{1, \dots, n\};\ q \in \{1, \dots, c\}, \quad (9)$$

where f_i^+ and f_i^- are the ith elements of f^+ and f^-, respectively, and $\gamma \in [0,1]$. Note that the effect of negative examples is suppressed by γ, the idea of which can be traced back to [4]: due to the asymmetry between relevant and irrelevant images, the positive and negative examples should be treated differently. Generally speaking, positive examples should make more contribution to the overall similarity score than negative ones. Here, the parameter γ controls the suppression extent: the smaller γ is, the less impact negative examples will have on the overall similarity score.

When only positive examples are available from the user's feedback or when we consider only the relevant images, we simply set $\gamma = 0$, and the similarity score is updated as

$$S_i = F_{i,q} + f_i^+, \quad i \in \{1, \dots, n\};\ q \in \{1, \dots, c\}. \quad (10)$$

4.3. Active learning

Contrary to passive learning, in which the learner randomly selects some unlabeled images and asks the user to provide their labels, active learning selects images according to some principle, hoping to speed up the convergence to the query concept. This scheme has been proven to be effective in image retrieval by previous research work [17, 20]. In [4], we have developed three active learning methods based on different principles, and each of them has its counterpart in the scenario of QBK.

The first method is to select the unlabeled images with the largest S_i, that is, the most positive images, which is widely used in previous research work [16, 21]. The motivation behind this simple scheme is to ask the user to validate the judgment of the current system on image relevance.

The second method is to select the unlabeled images with the smallest $|F_{i,q} + f_i^+ + \gamma f_i^-|$. Since the value of $F_{i,q}$ and f_i^+ indicates the relevance of an unlabeled image determined by initial labels and positive examples, respectively, while the absolute value of γf_i^- indicates the irrelevance of an unlabeled image determined by negative examples, a small value of $|F_{i,q} + f_i^+ + \gamma f_i^-|$ means that the image is judged to be relevant by the same degree as it is judged to be irrelevant, therefore, it can be considered an inconsistent one. From the perspective of information theory, such images are most informative.

The third method tries to take the advantage of the above two schemes by selecting the inconsistent images which are also quite similar to the query (most positive and inconsistent). To be specific, we select unlabeled images with the largest $F_{i,q} + f_i^+ - |F_{i,q} + f_i^+ + \gamma f_i^-|$. The scheme can be explained intuitively as follows: the selected images should not only provoke maximum disagreement among labeled examples (small $|F_{i,q} + f_i^+ + \gamma f_i^-|$), they must also be relatively confidently judged as a relevant one by the initial labels and the positive examples (large $F_{i,q} + f_i^+$).

In [4], we found out by experiments that the second scheme is not as effective as the other two. So, in this paper, we will only adopt the first (most positive) and the third (most positive and inconsistent) schemes.

For keyword propagation, another interesting matter with active learning is how to select the images for labeling in the stage of initial keyword model construction. However, we will not address this issue in this paper and will leave it to future work.

5. KEYWORD MODEL UPDATE

An important issue with keyword propagation is how to accumulate and memorize knowledge learnt from user-provided relevance feedback so that the retrieval system can be self-improved constantly. In [14], Jing et al. introduced labeling vectors to collect examples provided by the users, and their keyword model (an ensemble of binary SVM classifiers) is periodically updated by an offline training procedure.

It is very easy to incorporate such updating procedure in our method. Remember that each column F_q $(q = 1, \dots, c)$ in our keyword model corresponds to a keyword K_q. Taking the multiranking nature of keyword propagation, the updating procedure is performed on one by one column as follows.

Firstly, consider positive examples only. For a query keyword, its initial ranking vector is F_q. Let $Q^{j,+}$ $(j = 1, \dots, N^+)$ and $y^{j,+}$ denote the ensemble of the query sets and corresponding vectors from various positive feedback sessions, where N^+ is the total number of feedback sessions used to update the keyword model. Using Algorithm 1, their effect can be written as

$$f^{j,+} = \beta(I - \alpha S)^{-1} y^{j,+} \quad (j = 1, \dots, N^+). \quad (11)$$

Taking advantage of the incremental learning nature again, the final updating process can be denoted as

$$F_q \longleftarrow F_q + \sum_{j=1}^{N^+} f^{j,+} \quad (\text{for } q = 1, \dots, c). \quad (12)$$

If both positive and negative examples are considered, let $Q^{k,-}$ $(k = 1, \dots, N^-)$ and $y^{j,-}$ denote the ensemble of the query sets and corresponding vectors from various negative feedback sessions, where N^- is the total number of negative feedback sessions used to update the keyword model. By a similar analysis, we reach the following updating procedure:

$$F_q \longleftarrow F_q + \sum_{j=1}^{N^+} f^{j,+} + \gamma \sum_{k=1}^{N^-} f^{k,-} \quad (\text{for } q = 1, \dots, c), \quad (13)$$

where $f^{k,-} = \beta(I - \alpha S)^{-1} y^{k,-}$ $(k = 1, \dots, N^-)$, and $\gamma \in [0,1]$ is the controlling parameter as discussed in Section 4.2. If only positive examples are available or considered, $\gamma = 0$.

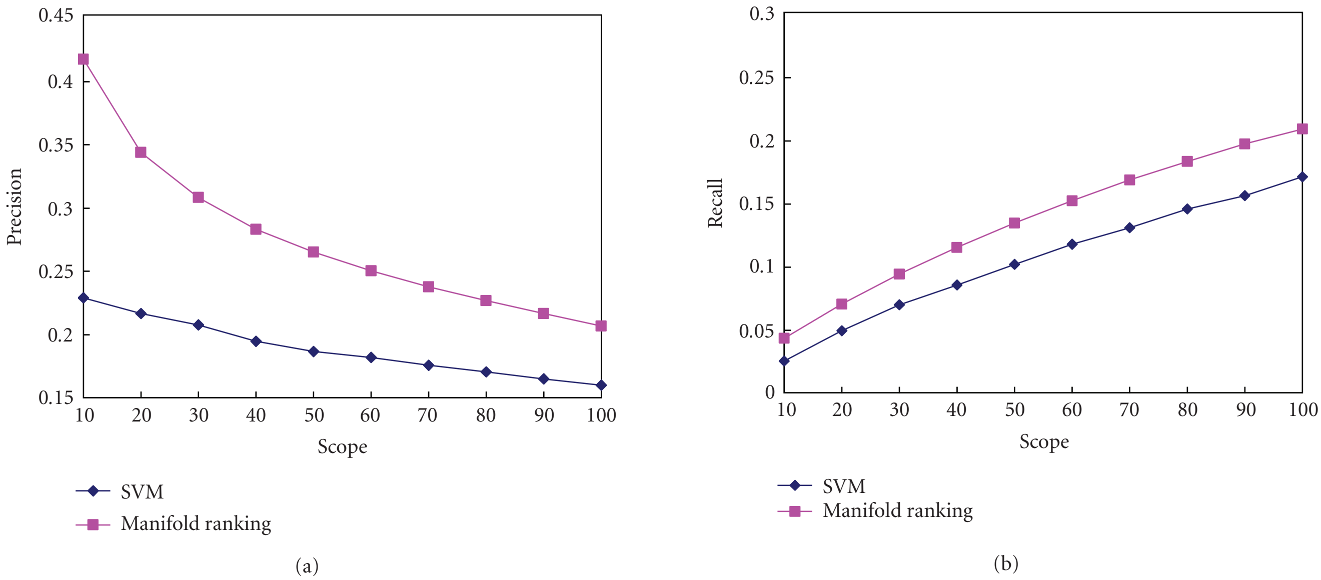

Figure 1: Comparison of the initial retrieval result between manifold ranking and SVM. Only 1% of the images were labeled for training. (a) Precision versus scope. (b) Recall versus scope.

In both cases, the ensemble of corresponding vectors actually plays a role of labeling vector as in [14]. Moreover, note that $f^{j,+}$ and $f^{k,-}$ are actually what we get during relevance feedback, thus the keyword model can be updated online during the relevance feedback sessions, and there is no extra offline training process.

6. EXPERIMENTAL RESULT

6.1. Experiment design

We have evaluated the proposed method with a general-purpose image database of 5 000 images from Corel. In our experiments, one percent (much less than that of [14]: ten percent) of all images in the database are randomly selected for manual annotation and used to train the initial keyword model. Currently, an image is labeled with only one keyword, that is, the name of the category that contains it. Some categories are sunset, mountain, eagle, beach, and (or) subsea animal. Totally, there are 50 keywords representing all images in the database. Images from the same (different) category are considered relevant (irrelevant). The precision versus scope curve is used to evaluate the performance of various methods. We use each keyword as a query. Considering the randomness of initial labels, we run 50 times of labeling and training for each query and the average retrieval result is recorded. Finally, we average the results over the total 50 queries.

Image feature has a great impact on the performance of image retrieval system. However, in this paper, our major concern is relative performance comparison, and therefore we do not perform careful feature selection. In our current implementation, the features that we use to represent each image include color histogram [7] and wavelet feature [10]. Color histogram is obtained by quantizing the HSV color space into 64 bins. To calculate the wavelet feature, we first perform 3-level Daubechies wavelet transform to the image, and then calculate the first- and second-order moments of the coefficients in High/High, High/Low, and Low/High bands at each level.

There are five parameters left to be set in the algorithm: K, α, σ_l, γ, and the iteration steps. As pointed out in [4], the algorithm is not very sensitive to the number of neighbors. In this paper, we set $K = 20$. The other four parameters are consistent with what we did in [4], that is, $\alpha = 0.99$, $\sigma_l = 0.05$, $\gamma = 0.25$, and the number of iteration steps is 50.

6.2. Initial retrieval result

Firstly, the initial retrieval is evaluated. The precision (recall) versus scope curve is shown in Figure 1. In order to perform a systematic evaluation, we vary the percentage of training data and compare the average precision (P20) and recall (R20) of top 20 retrieved images with that by SVM [14]. The precision (recall) versus the percentage of training data curve is shown in Figure 2. From the figures, it can be seen that our manifold-ranking-based method outperforms the classification-based one by a large margin, especially when only a small number of images were labeled for training. The improvement is very significant from the practical point of view.

6.3. Relevance feedback

In this case, we fix the total number of images that are marked by the user to 20, but vary the times of feedback and the number of feedback images each time accordingly. The combinations used in this experiment include 1 feedback

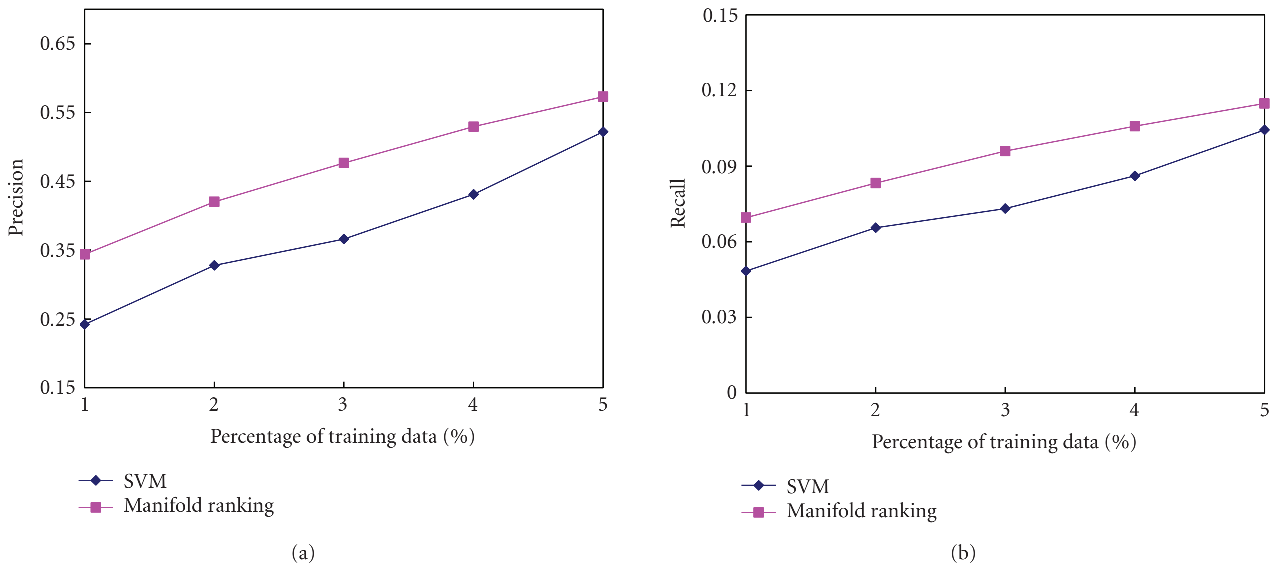

FIGURE 2: Systematic comparison between manifold ranking and SVM under different size of training data. (a) P20 versus the percentage of training data. (b) R20 versus the percentage of training data.

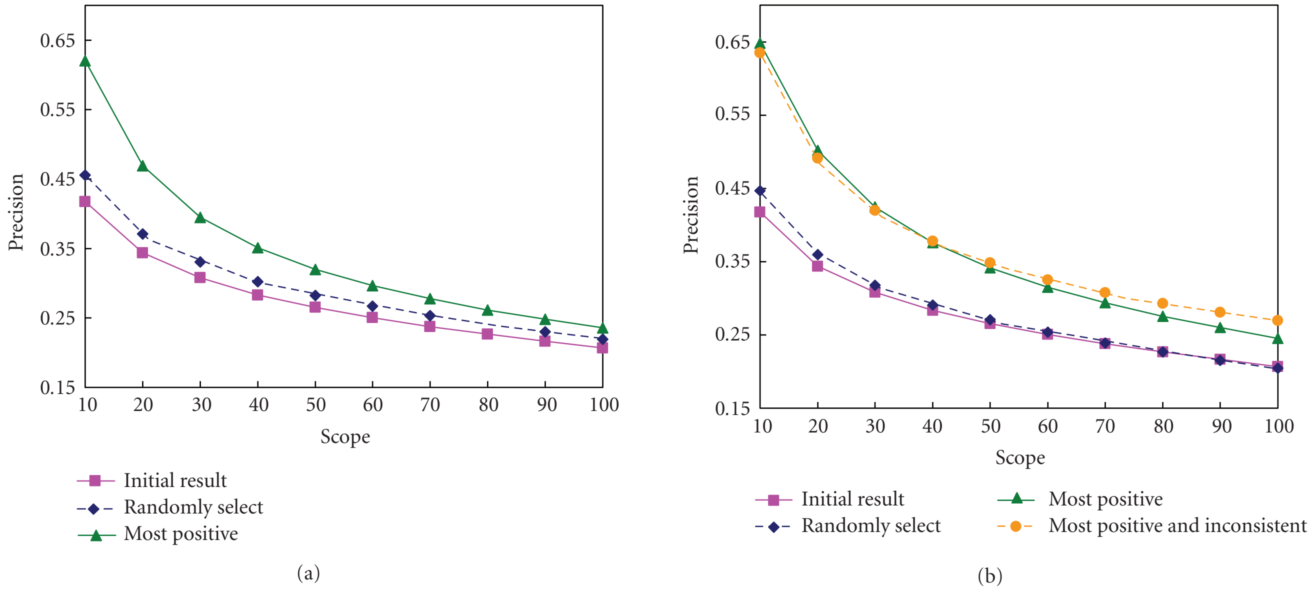

FIGURE 3: Comparison of different relevance feedback schemes (2 feedbacks with 10 images at each). (a) Only positive examples are considered. (b) Both positive and negative examples are considered.

with 20 images each time, 2 feedbacks with 10 images each time, and 4 feedbacks with 5 images each time. When both positive and negative examples are available, passive scheme (to select randomly) and two active schemes (to select most positive and to select most positive and inconsistent) are compared. When only positive examples are available, the most positive and inconsistent scheme is skipped. Note that, in the first round of relevance feedback, the most positive and inconsistent scheme is not provoked, and the most positive images are selected for users' labels. In all experiments, we find out that both of our active schemes help to improve the retrieval performance by a large margin, while the passive scheme makes little improvement. Here, we only present the results after 2 feedbacks with 10 images each time in Figure 3, and the initial result is also given as a reference.

6.4. Keyword model update

To collect training data for the updating process, each of the 50 keywords is used as the query once. In this case, users'

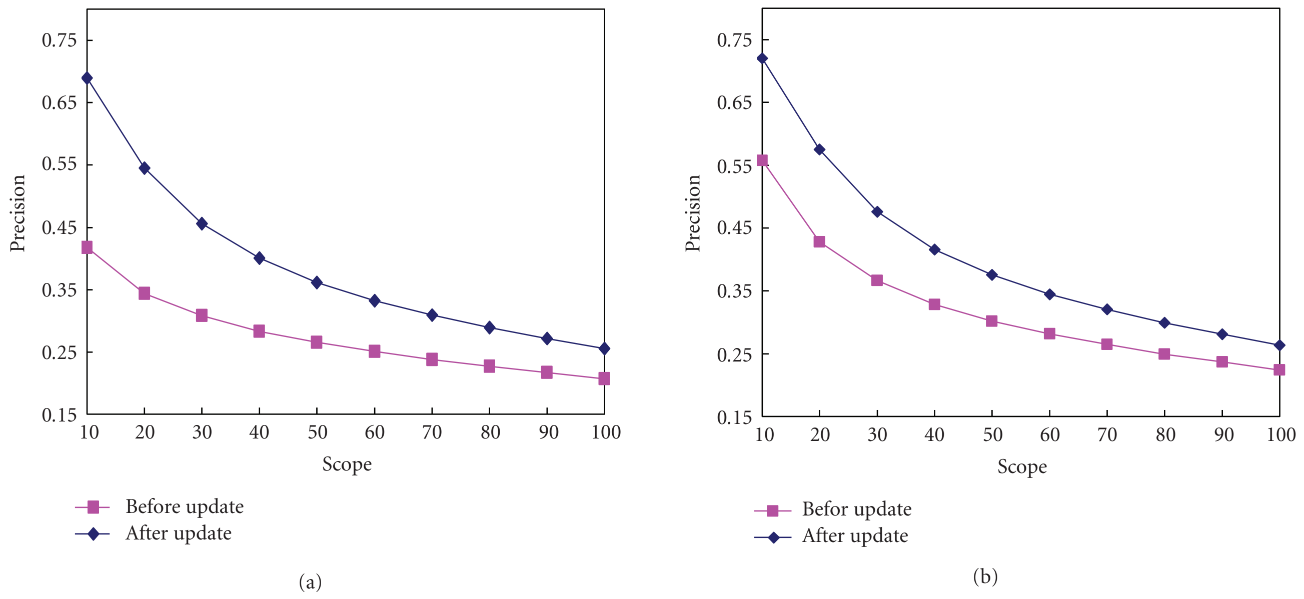

Figure 4: Comparison of performance improvement of the keyword model update process. (a) Initial retrieval result. (b) Relevance feedback (one feedback with 10 images).

feedback processes are simulated as follows. For a query image, 5 iterations of user-and-system interaction were carried out. At each iteration, the most 5 positive images are labeled by the user. Both positive and negative examples are considered. The initial retrieval result after updating the keyword model is presented in Figure 4(a), together with that without the updating procedure as a reference. The effect of updating process on subsequent relevance feedback sessions is also evaluated, and the retrieval results (one feedback with 10 images) with and without updating process are shown in Figure 4(b). It can be seen that the updating process enables the proposed system to self-improve progressively.

7. CONCLUSION

In this paper, we have proposed a novel method to support keyword propagation for image retrieval. This work is an extension to our previous work in the scenario of QBE and can be viewed as its counterpart in the scenario of QBK. Starting from a very small portion of labeled images, a keyword model is constructed by the manifold-ranking algorithm and all the images in the database are softly annotated. Different from existing methods which rely on labeled data to train an ensemble of binary classifiers, ours is a transductive one which explores the relationship among all labeled and unlabeled images in the learning stage. Such keyword model serves as a bridge that connects the semantic keyword space with the low-level feature space. The information from relevance feedback can be naturally incorporated to refine the retrieval result; and the influence of positive examples and negative ones are treated differently. Two active schemes are adopted to accelerate the convergence to the query concept. The proposed keyword model can be updated online without extra offline training process. Experiments on a general-purpose image database consisting of 5 000 Corel images demonstrate the effectiveness of the proposed method.

ACKNOWLEDGMENTS

This work is supported by the Project 60475001 of the National Natural Science Foundation of China. This work was performed at Microsoft Research Asia. The first two authors contribute equally to this paper.

REFERENCES

[1] S.-K. Chang and A. Hsu, "Image information systems: where do we go from here?" *IEEE Transactions on Knowledge and Data Engineering*, vol. 4, no. 5, pp. 431–442, 1992.

[2] H. Tamura and N. Yokoya, "Image database systems: a survey," *Pattern Recognition*, vol. 17, no. 1, pp. 29–43, 1984.

[3] H. T. Shen, B. C. Ooi, and K.-L. Tan, "Giving meanings to WWW images," in *Proceedings of 8th ACM International Conference on Multimedia*, pp. 39–47, Los Angeles, Calif, USA, October–November 2000.

[4] J. He, M. Li, H.-J. Zhang, H. Tong, and C. Zhang, "Manifold-ranking based image retrieval," in *Proceedings of 12th Annual ACM International Conference on Multimedia*, pp. 9–16, New York, NY, USA, October 2004.

[5] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image indexing using color correlograms," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pp. 762–768, San Juan, Puerto Rico, USA, June 1997.

[6] G. Pass, R. Zabih, and J. Miller, "Comparing images using color coherence vectors," in *Proceedings of 4th ACM International Conference on Multimedia*, pp. 65–73, Boston, Mass, USA, November 1996.

[7] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[8] F. Liu and R. W. Picard, "Periodicity, directionality, and randomness: wold features for image modeling and retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 7, pp. 722–733, 1996.

[9] B. S. Manjunath and W.-Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[10] J. Ze Wang, G. Wiederhold, O. Firschein, and S. X. Wei, "Content-based image indexing and searching using Daubechies' wavelets," *International Journal of Digital Libraries*, vol. 1, no. 4, pp. 311–328, 1998.

[11] C. Schmid and R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 5, pp. 530–535, 1997.

[12] X. S. Zhou, Y. Rui, and T. S. Huang, "Water-filling: a novel way for image structural feature extraction," in *Proceedings of IEEE International Conference on Image Processing (ICIP '99)*, vol. 2, pp. 570–574, Kobe, Japan, October 1999.

[13] E. Chang, K. Goh, G. Sychay, and G. Wu, "CBSA: content-based soft annotation for multimodal image retrieval using Bayes point machines," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 1, pp. 26–38, 2003.

[14] F. Jing, M. Li, H.-J. Zhang, and B. Zhang, "Keyword propagation for image retrieval," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '04)*, vol. 2, pp. 53–56, Vancouver, British Columbia, Canada, May 2004.

[15] Y. Lu, C. Hu, X. Zhu, H.-J. Zhang, and Q. Yang, "A unified framework for semantics and feature based relevance feedback in image retrieval systems," in *Proceedings of 8th ACM International Conference on Multimedia*, pp. 31–37, Los Angeles, Calif, USA, October–November 2000.

[16] L. Zhang, F. Lin, and B. Zhang, "Support vector machine learning for image retrieval," in *Proceedings of IEEE International Conference on Image Processing (ICIP '01)*, vol. 2, pp. 721–724, Thessaloniki, Greece, October 2001.

[17] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proceedings of 9th ACM International Conference on Multimedia*, vol. 9, pp. 107–118, Ottawa, Ontario, Canada, September–October 2001.

[18] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proceedings of 17th Annual Conference on Neural Information Processing Systems (NIPS '03)*, Vancouver, British Columbia, Canada, December 2003.

[19] D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Schölkopf, "Ranking on data manifolds," in *Proceedings of 17th Annual Conference on Neural Information Processing Systems (NIPS '03)*, Vancouver, British Columbia, Canada, December 2003.

[20] B. Li, E. Chang, and C.-S. Li, "Learning image query concepts via intelligent sampling," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '01)*, pp. 961–964, Tokyo, Japan, August 2001.

[21] F. Jing, M. Li, H.-J. Zhang, and B. Zhang, "An effective region-based image retrieval framework," in *Proceedings of 10th ACM International Conference on Multimedia*, pp. 456–465, Juan-les-Pins, France, December 2002.

Hanghang Tong received his B.S. degree in automation from Tsinghua University of China, in 2002. He is currently a M.S. candidate in Information Processing Institute, Department of Automation, Tsinghua University. His research interests include machine learning, data mining information retrieval and management, multimedia, and image processing.

Jingrui He received her B.S. degree in automation from Tsinghua University of China, in 2002. She is currently a M.S. candidate in Information Processing Institute, Department of Automation, Tsinghua University. Her research interests include machine learning, information retrieval and mining, multimedia, and computer vision.

Mingjing Li received his B.S. degree in electrical engineering from University of Science and Technology of China, in 1989, and Ph.D. degree in pattern recognition from Institute of Automation, Chinese Academy of Sciences, in 1995. He joined Microsoft Research Asia in July 1999. His research interests include handwriting recognition, statistical language modeling, and web image search.

Wei-Ying Ma received the B.S. degree in electrical engineering from the National Tsinghua University, Taiwan, in 1990, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California at Santa Barbara, in 1994 and 1997, respectively. From 1997 to 2001, he was with Hewlett-Packard Labs where he worked in the field of multimedia adaptation and distributed media services infrastructure. He joined Microsoft Research Asia in 2001. Since then, he has been leading a research group to conduct research in the areas of information retrieval, web search and mining, and multimedia management. He currently serves as an Editor for the ACM/Springer Multimedia Systems Journal and Associate Editor for the Journal of Multimedia Tools and Applications published by Kluwer Academic Publishers. He has served on the organizing and program committees of many international conferences including ACM Multimedia, ACM SIGIR, ACM CIKM, WWW, ICME, CVPR, SPIE Multimedia Storage and Archiving Systems, SPIE Multimedia Communication and Networking, and so forth. He is also the General Cochair of International Multimedia Modeling (MMM) Conference 2005 and International Conference on Image and Video Retrieval (CIVR) 2005. He has published four book chapters and over 100 international journal and conference papers.

Hong-Jiang Zhang received his Ph.D degree from the Technical University of Denmark, Lyngby, in 1991, and his B.S. degree from Zhengzhou University, Henan, China, 1982, both in electrical engineering, respectively. From 1992 to 1995, he was with the Institute of Systems Science, National University of Singapore, where he led several projects in video and image content analysis and retrieval and computer vision. From 1995 to 1999, he was a Research Manager at Hewlett-Packard Labs, Palo Alto, Calif, where he was responsible for research and development in the areas of multimedia management and intelligent imageprocessing. In 1999, he joined Microsoft Research in Beijing, where he is currently the Managing Director of Advanced Technology Center. Dr. Zhang is a Fellow of IEEE. He has coauthored/coedited four books, over 300 referred papers and book chapters, eight special issues of international journals on image and video processing, content-based media retrieval, and computer vision, as well as over 50 patents or pending applications. He is the Editor-in-Chief of IEEE Transactions on Multimedia.

Changshui Zhang was born in 1965. He received the B.S. degree in mathematics from Peking University, in 1986, and the Ph.D. degree from the Department of Automation, Tsinghua University, in 1992. He is currently a Professor in the Department of Automation, Tsinghua University. His interests include pattern recognition, artificial intelligence, image processing and evolutionary computation, and so forth.