CATEGORY-BASED SEARCH USING METADATABASE IN IMAGE RETRIEVAL *

Yimin Wu and Aidong Zhang

Department of Computer Science and Engineering,
State University of New York at Buffalo,
Buffalo, NY 14260, USA
{yiminkwu,azhang}@cse.buffalo.edu

ABSTRACT

In this paper, we present a self-adjustable metadatabase aiming at improving the performance of the relevance feedback module extensively used in content-based image retrieval systems. Our metadatabase provides a mechanism for accumulating the optimized relevance feedback records (which are called metadata records) obtained from previous queries. Each metadata record in the metadatabase includes optimal query, feature weights, and identifiers of relevant and/or irrelevant images, and can be effectively used to guide future queries. With the metadatabase, the relevance feedback module admits a noticeable improvement on its performance for category-based search, especially when the relevant images form multiple classes in the feature space. Experiments on a Corel image set (with 31,438 images) showed that our method has at least a 15% improvement on average precision and recall over relevance-feedback-only approaches.

1. INTRODUCTION

In image databases (which use content-based image retrieval techniques), images are indexed by low-level features such as color, texture and shape, but users often prefer to retrieve images according to high-level concepts. Since low-level features often fail to capture high-level concepts well, relevance feedback has been used as a powerful tool to bridge the gap between them. Many relevance-feedback-based image retrieval systems [1, 2, 3] have been developed so far. All these systems ask users to specify the relevance of currently retrieved images and adjust their retrieval schemes (e.g., query and feature weights) accordingly. With users in the retrieval loop, relevance-feedback-based systems can only anticipate their users to give relevance regarding a very limited number (usually less than 100) of images each time. Given the limited number of training images, it has already been very difficult to obtain the optimal retrieval scheme for one image class. And hence, most systems assume that all relevant images form one class in the feature space, even if they form multiple classes.

In this paper, we present a metadatabase to improve the performance of the relevance feedback module used in image retrieval systems. The metadatabase is a repository of metadata records obtained from previous queries. Each metadata record includes relevance feedback information (such as optimal query, feature weights, etc.), and is used to retrieve a relevant image class of the future queries. And hence, multiple image classes can be retrieved if multiple metadata records are considered as relevant by the user.

If no metadata record is considered as relevant, our relevance feedback module acts very similar with those of other systems. It optimizes the query and feature weights under the common assumption that all relevant images form a single class. What makes our relevance feedback module different from others is that it will generate a metadata record when it converges. Optimal metadata records are recorded in the metadatabase to guide future queries.

Experiments on a large-scaled image database demonstrate the effectiveness of our presented method. The proposed approach is presented in the next section. Experimental results are provided in Section 3. Finally, we conclude in Section 4.

2. CATEGORY SEARCH USING METADATABASE

2.1. Relevance Feedback Approach

In our system, each image i is represented by a feature vector $\vec{f} = [f_1, ..., f_M]^T$, where M is the dimensions of $\vec{f}$. Similarly, the query q is represented by the feature vector $\vec{q} = [q_1, ..., q_M]^T$. Database images with the top N smallest distances with the query are returned to the user. To calculate the distance between query q and image i, we use the following *weighted Euclidean* [4] distance metric:

$$d(q,i) = \sum_{m=1}^{M} w_m \times (q_m - i_m)^2, \qquad (1)$$

*THIS RESEARCH IS SUPPORTED BY THE NATIONAL SCIENCE FOUNDATION GRANTS.

where w_m specifies the importance of feature m.

We briefly introduce our method for calculating the weights of image features in this section. On the axis of feature m, we define *dominant range* G_m as the range spanned by the relevant images, as shown in Figure 1. We then define the *discriminative factor* δ_m of feature m as:

$$\delta_m = \frac{|U_m^\delta|}{|U|}, \tag{2}$$

where U is the irrelevant image set and $U_m^\delta \subseteq U$ is the subset of U falling outside of G_m. The *discriminative factor* δ_m indicates the ability of feature m in discriminating relevant images from irrelevant ones.

Denote the relevant image set by $R = \{r_1, ..., r_T\}$, the weight w_m is updated by:

$$w_m = \frac{\delta_m}{\sigma_m^R}, \tag{3}$$

where σ_m^R is the standard deviation of the sequence $R_m = [r_{1,m}, ..., r_{T,m}]$, which is obtained by stacking the mth feature values of relevant images. Unlike the Mars approach [1] which only uses $\sigma_{m,s}^R$ to re-weight features, our approach combines relevant and irrelevant images to re-weight image features.

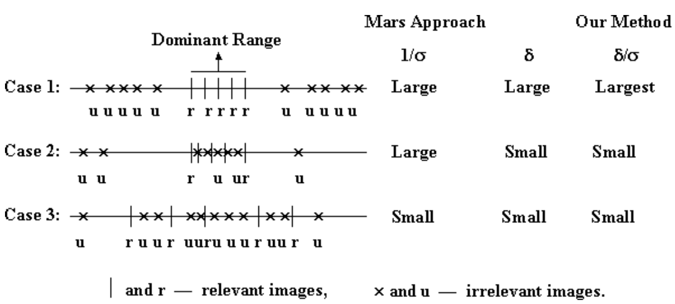


Fig. 1. The weight value of one feature in different cases.

Our method and the Mars approach [1] are compared in Figure 1. The Mars approach assigns large weights to all features that allocate relevant images together. However, it can not distinguish case 1 and case 2 and assigns large weights to both of them. On the contrary, our approach employs relevant images as well as irrelevant ones to calculate feature weights. And hence, our approach correctly distinguishes case 1 from case 2, and only assigns large weights to the former. Extensive experimental results presented in [2] demonstrate that our method consistently outperforms the Mars approach.

After each relevance feedback iteration, the optimal query $q = \{\vec{q_1}, ..., \vec{q_M}\}^T$ is updated as the average value of all relevant feature vectors.

2.2. Metadatabase for Category Search

The relevance feedback module is prone to be stuck in some sub-optimal state. However, the chance that it got stuck can be noticeably reduced if previously optimized queries can be incorporated into current database search. So we accumulate relevance feedback information in the metadatabase to guide future queries.

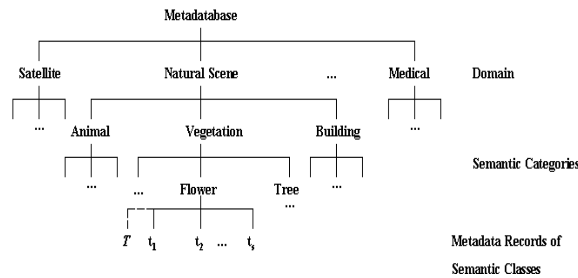


Fig. 2. The hierachy of metadatabase.

The metadatabase accumulates relevance feedback information using metadata records. Each metadata record is a 5-tuple $t =< q,W,R,U,P >$, where q is the optimal query, $W = \{w_1, ..., w_m, ...w_M\}$ is the set of image feature weights, R and U are the identifier sets of relevant and irrelevant images,respectively, and P is the precision of t given by $|R|/(|R| + |U|)$.

Metadata records in the metadatabase are organized using a tree like structure shown in Figure 2. On the top levels of the tree, semantic categories (expressed in text keywords) are introduced to associate the search interests of different users. The reason of using text keywords is that users can easily associate their search interests with appropriate text keywords if they are interested in category-based search such as *locating images of tigers*. For each semantic category c, a *synthetic metadata record* $\mathcal{T} =< MaxP, MinP >$ is introduced, where $MaxP$ and $MinP$ are the maximum and minimum precision of all the metadata records of c, respectively. To avoid generating a overwhelmingly large-sized metadatabase, the maximum number of metadata records in each semantic category is set to a threshold ζ.

In the metadatabase, metadata records are supposed to represent image classes different from each other. To guarantee this, we define the *overlap ratio* $\mathcal{O}_{t_1,t_2}$ between two records t_1 and t_2 as:

$$\mathcal{O}t_1, t_2 = \frac{|t_1.R \cap t_2.R|}{|t_1.R \cup t_2.R|}. \tag{4}$$

The *overlap ratio* $\mathcal{O}t_1, t_2$ indicates the percentage of overlapped relevant images over all relevant images recorded in t_1 and t_2. Let the metadata record set of semantic category c be $S = \{t_1, ..., t_s\}$, the overlap ratio between metadata record t and S is calculated by

$$\mathcal{O}_{t,S} = Max_{\forall t_j \in S} \mathcal{O}_{t,t_j}, \tag{5}$$

which is the maximum overlap ratio between t and any member of S. The *less precision set* $S_t \subseteq S$ of t is defined as:

$$S_t = \{\forall t_j | t_j \in S \ and \ t_j.P < t.P\}. \tag{6}$$

If a newly generated metadata record t is associated with semantic category c, S (the metadata record set of c) is adjusted by the following algorithm:

1. **If** $t.P > \mathcal{T}.maxP$, set $S = \{t\} \cup \{\forall t_j | t_j \in S \ and \ \mathcal{O}_{t_j,t} \leq \eta\}$, go to step 5.
2. **Elseif** $|S| < \zeta$ and $\mathcal{O}_{t,S} \leq \eta$, set $S = \{t\} \cup S$, go to step 5.
3. **Elseif** $\exists t_j \in S_t$ with $\mathcal{O}_{t,S'} \leq \eta$, where $S' = S - \{t_j\}$. Set $S = \{t\} \cup S'$, go to step 5.
4. **Else** Discard t, return.
5. **Update** the synthetic metadata record $\mathcal{T}$, return.

η used in steps 2 and 3 of the algorithm is an experimental threshold to guarantee that each metadata record represents a separate image class. We set η to 0.2 in our experiments since most semantic categories end up with only one metadata record if it is too small. The first three steps of the above algorithm specify the three optimization criteria. According to step 1, the metadatabase always keeps the metadata record t with the highest precision. The second step demands that t be added to S if the size of S is less than ζ, and the overlap rate between t and S is no larger than η. According to step 3, t will replace one of the records in its less precision set S_t if S_t is nonempty. A metadata record is discarded if it fails all optimization criteria.

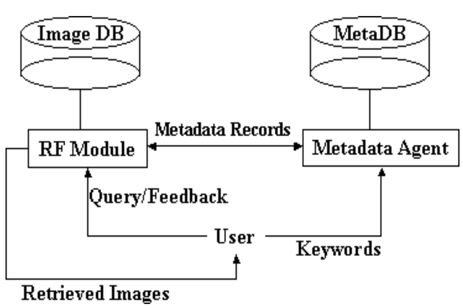


Fig. 3. An Illustration of Our System.

The interaction between relevance feedback module, metadata agent and the user are presented in Figure 3. To facilitate searching the image database, the user can locate relevant metadata records from the metadatabase. Relevant metadata records can be used by the relevance feedback module to improve its performance. If no relevant metadata record is located, the relevance feedback module learns an optimal query and feature weight set from user relevance feedback. On its convergence, the relevance feedback module asks the user to associate his or her search interests with text keywords. It then generates a metadata record to record the relevance feedback information. The metadata record is submitted to the metadata agent, which updates the metadatabase using the algorithm presented in this section. If k (> 0) relevant metadata records are located, the relevance feedback module uses all the k optimal queries and feature weight sets to search the image database, and the top N/k nearest neighbors of each optimal query are returned, where N is the total number of images returned to the user. The user then labels the retrieved images as usual, and the relevance feedback module adjusts the k optimal queries and feature weight sets accordingly.

3. EXPERIMENTS

3.1. Experiment Setup

In our experiments, each image is represented by a 59-bin feature vector,which is consisted of four image features. The first one is a 32-bin HSV color histogram[1]. The second one is the 9-bin color moments [5] defined on L*a*b color space. The third feature is a 10 bin wavelet-based texture feature [1]. The fourth feature is a 8-bin edge direction histogram [6] which captures the distribution of edge pixels in the primary 8 directions in the edge maps of images.

Our approach is tested on a image database with 31,438 Corel images, and 6,457 images from 73 semantic categories (such as tiger, rose and city, etc.) are used as queries.

Precision and *recall* are used to evaluate the performance of our method. *Precision* is the number of retrieved relevant images over the total number of retrieved images, and *recall* is the number of retrieved relevant images over the total number of relevant images. The average precision and recall of all queries are used as the overall performance. Since the performance of relevance feedback module also depends on the *scope* (i.e., the number of images returned to the user), we test our method for scopes of 40 and 80.

3.2. Result Analysis

Experimental results are shown in Table 1. In the table, P and R denote precision and recall, respectively. The performance for scopes of 40 or 80 are listed from the top down. The column under *Plain* gives the performance of traditional image retrieval approach without relevance feedback, the next column shows the performance of the Mars approach [1], *RF* column provides the performance of our relevance feedback approach presented in Section 2.1, the

%	Plain	Mars	RF	1 Class	2 Classes	3 Classes
P	9.85	17.15	21.64	38.84	48.15	53.39
R	6.42	9.88	12.86	23.36	30.69	35.51
P	6.83	13.08	16.56	27.38	34.46	38.24
R	8.18	14.10	18.04	30.33	41.47	47.73

Table 1. Comparisons when $scope = 40, 80$.

remaining three columns show the performance of using 1, 2 and 3 relevant metadata records, under the assumption that relevant images form 1, 2 and 3 classes, respectively.

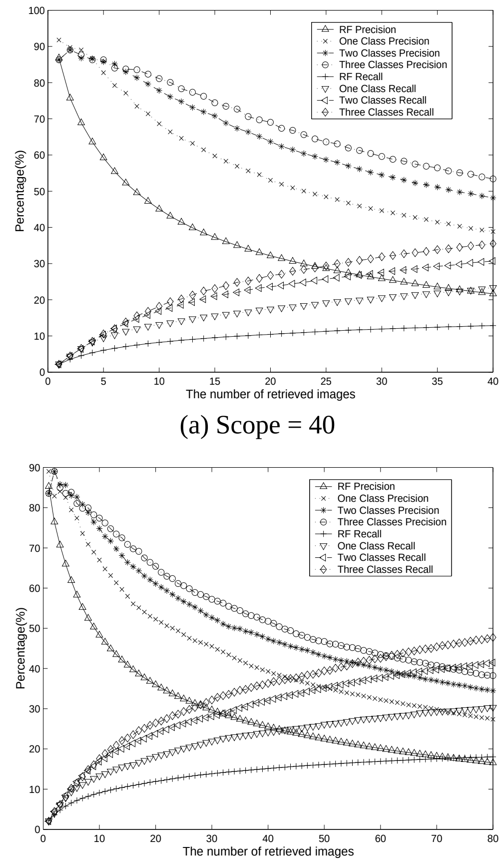
As shown in Table 1, noticeable improvement on performance can be achieved by applying relevance feedback, while our approach outperforms the Mars approach by about 5%. However, the performance of category-based search can be improved more drastically by integrating the metadatabase. As shown in Table 1, the performance is improved by 15% when using only one relevant metadata record, under the assumption that relevant images form one class. The performance keeps improving when more relevant metadata records are used. 10% improvement on performance is achieved by 2 classes over 1 class, and an additional 5% improvement is attained by 3 classes over 2 classes. The improvement on performance of multiple classes over one class demonstrates that relevant images tend to form multiple classes in the feature space.

The precision and recall curves of different approaches are shown in Figure 4. These curves show how the performance of different approaches change as the number of retrieved images increases. We can learn from these curves that the image retrieval performance is constantly improved by the integration of the metadatabase.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a metadatabase to improve the performance of category-based image retrieval. Our metadatabase provides a mechanism for accumulating the optimized relevance feedback records obtained from previous queries. With the metadatabase, the relevance feedback module admits a noticeable improvement on its performance for category-based search, especially when the relevant images form multiple classes in the feature space. Experimental results on a very large-scaled image database demonstrate the effectiveness of our proposed approach.

To extend our research, some challenging problems need to be further investigated: (1) How to apply metadatabase for queries that can not clearly be associated with text keywords? (2) How to search the metadatabase using a content-based image retrieval technique? We plan to study some of these problems in our future research.



(a) Scope = 40

(b) Scope = 80

Fig. 4. Precision and recall curves of different approaches.

5. REFERENCES

[1] Y. Rui, T.S. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in mars," in *Proc. IEEE Int. Conf. on Image Proc.*, 1997.

[2] Yimin Wu and Aidong Zhang, "A feature re-weighting approach for relevance feedback in image retrieval," in *Proc. IEEE Int. Conf. on Image Proc. (to appear)*, 2002.

[3] Y. Rui and T.S. Huang, "Optimizing learning in image retrieval," *IEEE Conf. on CVPR*, June 2000.

[4] Y. Ishikawa, R. Subramanya, and C. Faloutsos, "Mindreader: Query databases through multiple examples," in *Proc. of the 24th VLDB Conference*, 1998.

[5] W. Y. Ma and H. Zhang, *Handbook of Multimedia Computing*, chapter Content-Based Image Indexing and Retrieval, pp. 19–20, London: Routledge, 1998.

[6] A. Vailaya, A.K. Jain, and H.J. Zhang, "On image classification: City images vs. landscapes," *Pattern Recognition*, vol. 31, pp. 1921–1936, December 1998.