INTER-QUERY SEMANTIC LEARNING APPROACH TO IMAGE RETRIEVAL

Scott Fechser, Ran Chang, and Xiaojun Qi

s.fechser@aggiemail.usu.edu, ran.c@aggiemail.usu.edu, and Xiaojun.Qi@usu.edu
Computer Science Department, Utah State University, Logan, UT 84322-4205

ABSTRACT

This paper presents an inter-query semantic learning approach for image retrieval with relevance feedback. The proposed system combines the kernel biased discriminant analysis (KBDA) based low-level learning and semantic log file (SLF) based high-level learning to achieve high retrieval accuracy after the first iteration. User's relevance feedback is utilized for updating both low-level and high-level features of the query image. Extensive experiments demonstrate our system outperforms three peer systems.

Index Terms**—** KBDA, CBIR, semantic log file

1. INTRODUCTION

Relevant feedback (RF) techniques have been widely used in content-based image retrieval (CBIR) to refine the user query through interactive sessions and bridge the semantic gap. Various RF techniques [1] have been proposed to improve retrieval performance. However, most RF techniques focus on query tuning in a single retrieval session, which is known as intra-query learning or short-term learning. Recently, inter-query learning, known as long-term learning, has gained attention in CBIR research. It extends intra-query learning and aims to analyze the relationship between the current and past retrieval sessions to improve retrieval performance.

Inter-query learning can be classified into retrieval pattern based learning and feature vector model based learning. The retrieval pattern based learning is to establish the relationship between the current and previous query sessions by analyzing the retrieval patterns between the sessions. The feature vector model based learning is to bring the feature vectors of similar images close to each other by a weighting scheme or a transformation technique. Here, we review representative techniques in each category.

Retrieval pattern based learning: Heisterkamp [2] applies the latent semantic analysis method on the term-by-document matrix to provide a generalization of the relationship between the current query and the search history. He *et al.* [3] use the semantic space to store the retrieval patterns (i.e., labels of relevant and irrelevant images) of query sessions and find semantically similar images. Hoi *et al.* [4] apply the statistical correlation on the retrieval log to analyze the relationship between the current and past retrieval sessions. Han *et al.* [5] uses the memory learning technique to compute the ratio of the co-positive feedback frequency and the co-feedback frequency for analyzing the relationship among query sessions. All these techniques require a matrix of size $N \times N$ to store memorized feedback information, where N is the number of images in the database. In addition, the sparsity of matrices may also make learning less useful for a large database.

Feature vector model based learning: Hang and Irwin [6] apply self-organizing map (SOM) to incorporate the historical retrieval information learned from the biased SVM and to improve retrieval performance for future queries. Gondra *et al.* [7] propose to learn one-class SVM (1SVM) from retrieval experience to represent the memberships of users' high-level concepts and store them in a concept database, which provides a mechanism for accumulating inter-query learning. They then apply fuzzy classification on the query to find appropriate regions represented by the 1SVM. Chung *et al.* [8] use data clusters to memorize the inter-query relationship. These clusters are dynamically updated by applying statistical discriminant analysis on past retrieval sessions. All these techniques require remembering the parameters captured in the learning process. Furthermore, the number of overlapping clusters may be constantly growing as the amount of inter-query learning increases.

In this paper, we seamlessly integrate both learning strategies to address their perspective shortcomings. First, we apply kernel biased discriminant analysis (KBDA) to find an optimal transform that can best discriminate the positive images from the negative images. Second, we apply a clustering algorithm to merge the feature spaces of multiple queries to avoid the constantly growing number of overlapping clusters. Third, we convert the KBDA-based clusters into a semantic log file (SLF) of size $N \times C$, with N being the number of images in the database and C being the number of clusters. Fourth, we apply the semantic learning technique on the SLF to find the images which are semantically similar to the query. The rest of the paper is organized as follows: Section 2 presents our proposed learning approach. Section 3 compares our system with peer systems. Section 4 draws conclusions and presents future directions.

978-1-4244-4296-6/10/$25.00 ©2010 IEEE

2. PROPOSED INTER-QUERY LEARNING SCHEME

The block diagram of our method is shown in Fig. 1. For each query, the system first returns top *n* images based on the low-level visual similarity between the query image and each database image. The user then selects relevant (i.e., positive) images from the returned pool while treating non-selected images as irrelevant (i.e., negative) images. This is designated as intra-query feedback and is performed at each iteration step. The KBDA technique is then applied to transfer the intra-query feedback to a new feature space to better discriminate the positive images from the negative images. At the end of each query session, the intra-query feedback is saved in inter-query feedback clusters (IQFCs) and updated in the semantic log file (SLF). Specifically, the system either saves current intra-query feedback as a new cluster within the SLF or merges it with the identified clusters within the SLF to form new clusters. The accumulated collections of the KBDA-based intra-query feedbacks are designated as inter-query feedback. The semantic learning technique is finally applied on the SLF to find the semantically similar images. The following subsections will explain each component in detail.

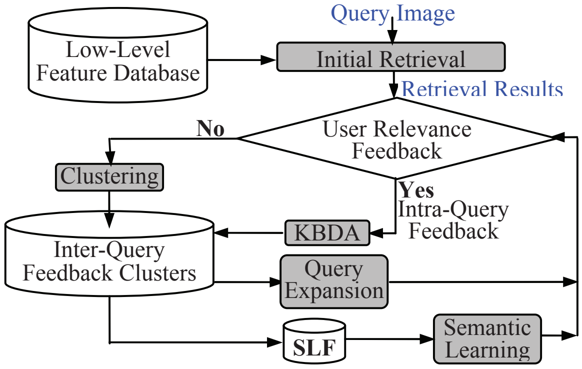

Fig. 1: The block diagram of our CBIR system.

2.1. Initial Retrieval

We use 64-bin HSV color histogram and 36 features to represent each database image. The 36 features consist of 9 color, 18 edge, and 9 texture components. Color features are the first three moments in HSV color space. Edge features correspond to an 18-bin edge direction histogram of the converted grayscale image. Texture features are the entropy of each of 9 wavelet detail subbands of the grayscale image. To measure the similarity between the query and each database image, the Euclidean distance is computed for color and texture features and the histogram intersection is computed for histogram and edge features.

2.2. Kernel Biased Discriminant Analysis (KBDA)

We employ KBDA to designate images as positive and negative due to its effectiveness in biased learning (i.e., the user is biased toward one class corresponding to positive images) [9]. The aim of KBDA is to find optimal transform that clusters only positive images while keeping negatives away. It then transfers data to a new feature space that can best discriminate the positive from the negative images. The Gaussian radial basis function (RBF) kernel is used due to its excellent performance. With asymmetric treatment biased toward the positive images, the objective function is:

$$W_{opt} = \arg\max_{w} \frac{\| W^T S^{\phi}_{neg} W \|}{\| W^T S^{\phi}_{pos} W \|} \qquad (1)$$

That is, it applies a set of weight vectors *W* to maximize the ratio between the negative covariance matrix and the biased positive covariance matrix. These two matrices are:

$$S^{\phi}_{pos} = \sum_{i=1}^{N_{pos}} \left(\phi(x_i) - m^{\phi}_{pos}\right)\left(\phi(x_i) - m^{\phi}_{pos}\right)^T \qquad (2)$$
$$S^{\phi}_{neg} = \sum_{i=1}^{N_{neg}} \left(\phi(y_i) - m^{\phi}_{pos}\right)\left(\phi(y_i) - m^{\phi}_{pos}\right)^T$$

Here, $\{x_i, i=1, \ldots, N_{pos}\}$ and $\{y_i, i=1, \ldots, N_{neg}\}$ respectively denote the positive and negative examples, Φ represents the kernel mapping function, and m^{ϕ}_{pos} is the mean vector of the positive transformed examples. This is a generalized eigen-analysis problem, where the optimal eigen-vectors associated with the eigen-values are the transform matrix for the new feature space.

2.3. Query Expansion

Based on the user's accumulative feedback on positive and negative images at each retrieval session, KBDA is applied to compute the optimal transform at each iterative step. The query is expanded as a cluster to more accurately represent itself. This cluster contains the optimal transform matrix, the indices of all positive and negative images, positive centroid of size $(N_{pos}+N_{neg})\times N_{pos}$ and negative centroid of size $(N_{pos}+N_{neg})\times N_{neg}$, both in the dot-production form suitable for the kernel evaluation, and a radius. The radius is the maximum distance from the projected positive image to the positive centroid (Dist$_1$) or the average of Dist$_1$ and Dist$_2$, which is the minimum distance from the projected negative image to the negative centroid.

After at least one inter-query feedback cluster is constructed, the retrieval process works as follows: 1) Perform the initial retrieval to returns top *n* image for the user to label. 2) The user selects the positive images from the returned pool of images. 3) Perform the search scheme to find candidate semantically similar clusters that contain the most number of current positive images and are likely to represent the query image. This search scheme works as follows: a) Count the number of positive images (e.g., N_i) that fall within the boundary of each existing cluster and compute T_p by N_i/N_p, where N_p is the total number of positive images gathered during the feedback cycle. b) Find all the clusters whose T_p's are larger than 0.8. If none of the existing clusters has the large T_p value, we will find all clusters whose T_p's are larger than 0.3. 4) Expand the query by merging the accumulative positive and negative images of the current query with the identified candidate clusters to represent the query in a cluster, whose optimal transform is updated using positive and negative images in both the current query and candidate clusters. Compute Dist_1 and Dist_2 independently. If $\mathrm{Dist}_1 < \mathrm{Dist}_2$, Dist_1 is set as the radius for the merged cluster. Otherwise, the average of Dist_1 and Dist_2 is set as the radius for the merged cluster. 5) Each database image is projected into the new feature space determined by the merged query cluster and the top n images that are within the radius of the cluster are returned for next round of RF iteration.

2.4. Clustering and IQFC Construction

At the end of each query session, a current query cluster is created. For each existing cluster, we project its positive images using the optimal transform matrix of the current query cluster and compute the median distance from these projected positive images to the positive center of the current query cluster (i.e., MDist_1). Similarly, we project the positive images in the current query cluster using the optimal transform matrix of each existing cluster and compute the median distance from these projected positive images to the positive center of each existing cluster (i.e., MDist_2). If MDist_1 is less than the radius of the current query cluster or the MDist_2 is less than the radius of the corresponding existing cluster, the current query cluster is merged with the existing cluster. The merging procedure is the same as step 4 of query expansion.

2.5. SLF Construction

The SLF stores the semantic relationship between database images and IQFCs, which are created by using unique, randomly selected training images. The IQFCs correspond to the columns of the SLF and the database images correspond to the rows of the SLF. We refer to the number of columns as the size of the SLF.

Initially empty, the SLF fills up with RF information gathered from IQFC, which is updated at the end of each query session. Specifically, the rows corresponding to the positive images in IQFC are filled with 1's at the column representing this IQFC; the rows corresponding to the negative images in IQFC are filled with -1's at the column representing this IQFC; the remaining rows are filled with 0's. The i^{th} row is the semantic feature vector (SFV) of database image x_i.

2.6. Semantic Learning

The SLF-based semantic learning and search starts with finding the semantic rows corresponding to the positive and negative images labeled in the initial retrieval. The query's semantic feature vector (QSFV) is initialized as:

$$q_k^S(t) = (s_k^{P,1} \vee \ldots \vee s_k^{P,Np}) \wedge \overline{(s_k^{N,1} \vee \ldots \vee s_k^{N,Nn})} \qquad (3)$$

where $q_k^S(t)$ is the k^{th} element of the QSFV, $s_k^{P,i}$ and $s_k^{N,i}$ are the k^{th} element of the SFV of the i^{th} positive and negative images, respectively. The values of N_p and N_n correspond to the number of positive and negative images, respectively. Here, we treat all negative values as 0's.

If the initial QSFV contains all 0's, the query update process is applied to obtain more information. This update process is also applied to the following feedback iterations where positive images reinforce the semantically positive features of the QSFV and negative images suppress the negative features of the QSFV. This process is as follows:

$$q_i^S(t+1) = \begin{cases} \alpha q_i^S(t) & if\ (S_i^P = 1\, or\, S_i^N = -1), q_i^S(t) \neq 0 \\ 1 & if\ (S_i^P = 1\, or\, S_i^N = -1), q_i^S(t) = 0 \\ q_i^S(t) & if\ S_i^P = 0\, or\, S_i^N = 0 \\ q_i^S(t)/\alpha & if\ S_i^P = -1\, or\, S_i^N = 1 \end{cases} \qquad (4)$$

where $q_i^S(t+1)$ is the i^{th} element of the updated QSFV, s_i^P and s_i^N correspond to the i^{th} element of SFVs of the positive and negative images, respectively. The parameter α is the adjustment rate. It is 1.01 to the power of the SLF size for positive images and 1.01 to the power of the base 2 logarithm of the SLF size for negative images. This is, the larger the SLF size, the more reinforcement the positive features, the more suppression the negative features.

The dot product computes the semantic similarity scores between query $q^S(t)$ and database image x_i.

$$S_{x_i}^{Sem} = x_i \cdot q^S(t) = \sum_k x_{i,k} q_k^S(t) \qquad (5)$$

3. EXPERIMENTAL RESULTS

We tested our system on 2000-Flickr images, 6000-COREL images, and the combined 2000-Flickr and 6000-COREL images. The COREL DB contains 60 distinct categories with 100 images per category. The Flickr DB contains 20 categories with 100 images per category. The images for 20 categories were obtained by searching for distinct keywords using Flickr's API. We downloaded the top 150 images for each category and manually picked 100 appropriate images.

To evaluate the effectiveness of our proposed system, we designed a set of experiments on two small DBs: the 2000-Flickr DB and the 2000-COREL DB, a subset of the 6000-COREL DB. To facilitate the evaluation process, we designed an automatic feedback scheme to construct IQFC and SLF by using KBDA to perform query expansion and clustering. A retrieved image is considered as positive if it belongs to the same category as query. The retrieval accuracy is computed as the ratio of the positive images to the total returned images. We randomly chose 2%, 5%, and 10% of images from each category of the test DB as queries to construct three kinds of LQFCs and SLFs, respectively. During this learning period, we returned 25 images at each of the four iterations of the query session. To speed up the initial learning and maximize the amount of semantic information that could be learned on the training set, we also made sure that a retrieved image would not be returned in the following iterations. The LQFCs and SLF are dynamically created or updated after each query session. After the training, the system was tested using the remaining 90% of the DB images as queries. No additional learning was performed, and images were allowed to be returned multiple times during a single query session. Fig. 2 shows the average retrieval accuracy for 1800 images using different SLFs as a learning base. It clearly shows the retrieval accuracy is improved for both DBs when more training images are used to construct LQFCs and SLF and therefore more semantic relationships among the images are stored in SLF. The retrieval accuracy is above 90% after the 1st iteration for three learning bases of the COREL DB. The retrieval accuracy is above 90% at the 4th iteration for the largest learning base (10%) of the Flickr DB. This is mainly due to the distinct semantic information in the COREL DB. Therefore, we chose 10% of the DB images as the training images to construct LQFCs and SLF in order to better represent the diversity of the images in each category.

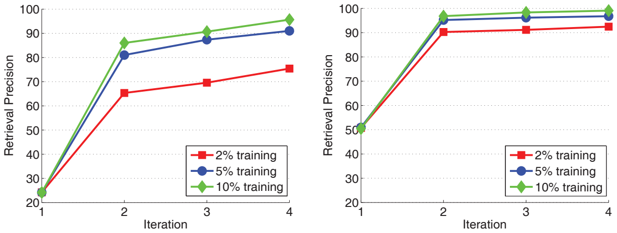

Fig. 2: Retrieval performance on 2000-Flickr DB (left) and 2000-COREL DB (right) using different training images.

We compared our system with Hoi's log-based [4], Han's memory learning [5], and manifold systems [10] on three DBs. Fig. 3 shows the average retrieval precision of four systems after using 10% of the DB images to build their perspective learning bases. It clearly shows that our system achieves the best precision and the memory learning system achieves the second best precision after the 1st iteration on three DBs. Specifically, comparing to the 2nd best system for the last two iterations on the 2000-Flickr, the 6000-COREL, and the combined DB, our system makes 5.4% and 1.9%, 15.4% and 10.4%, and 11.2% and 5.7% improvement, and achieves accuracy of 86.0% and 90.7%, 87.8% and 90.6%, and 78.1% and 82.8%, respectively. The number of clusters (the SLF size) is 49, 138, and 244 for 2k, 6k, and 8k images, respectively. That is, for these three DBs, our system uses matrices of 2k×49, 6k×138, and 8k×244 instead of matrices of 2k×2k, 6k×6k, and 8k×8k as used in the three inter-query systems to store the semantic relationships. Our computation time is also comparable to the peer systems mainly due to the small size of the matrix.

4. CONCLUSIONS

We propose a novel inter-query semantic learning CBIR system with RF. Major contributions consist of: 1) Using KBDA technique for biased learning. 2) Using SLF for semantic learning. 3) Combining KBDA learning and SLF-based semantic learning to achieve high retrieval accuracy after the 1st iteration. Experimental results show the proposed system outperforms peer systems and achieves highest retrieval accuracy after the 1st iteration. Other biased learning schemes and other forms of SLF will be investigated in future research.

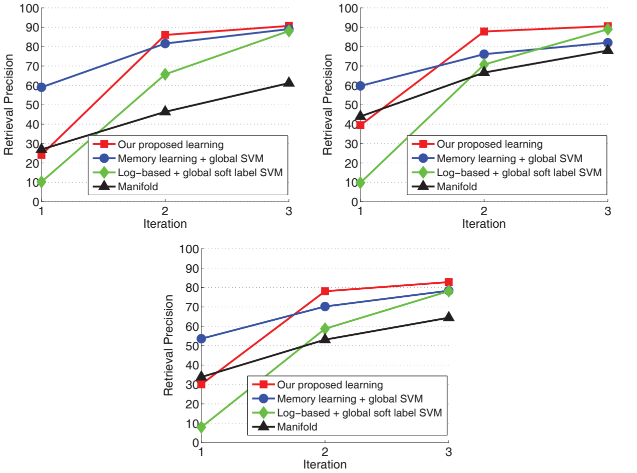

Fig. 3: Comparison of four CBIR systems. Clockwise from the upper left: 2000-Flickr, 6000-COREL, and 8000 images.

5. REFERENCES

[1] X. Zhou and T. Hung, "Relevance Feedback in Image Retrieval: a Comprehensive Review," *ACM Multimedia Sys. J.*, Vol. 8, pp. 536-544, 2003.

[2] D. Heisterkamp, "Building a Latent Semantic Index of an Image Database from Patterns of Relevance Feedback," *Proc. of ICPR*, pp. 134-137, 2002.

[3] X. He, O. King, W. Y. Ma, M. Li, and H. Zhang, "Learning a Semantic Space from User's Relevance Feedback for Image Retrieval," *IEEE Trans. CSVT*, Vol. 13, pp. 39-48, 2003.

[4] S. C. H. Hoi, M. R. Lyu, and R. Jin, A Unified Log-Based Relevance Feedback Scheme for Image Retrieval," *IEEE Trans. KDE*, Vol. 18, No. 4, pp. 509-524, 2006.

[5] J. Han, K. Ngan, M. Li, and H. Zhang, "A Memory Learning Framework for Effective Image Retrieval," *IEEE Trans. IP*, Vol. 14, No. 4, pp. 511-524, 2005.

[6] C. H. Chan and I. King, "Using Biased Support Vector Machine to Improve Retrieval Results in Image Retrieval with Self-Organizing Map," *Proc. of ICNIP*, pp. 714-719, 2004.

[7] I. Gondra, D. Heisterkamp, and J. Peng, "Improving Image Retrieval Performance by Inter-Query Learning with One-Class SVMs", *Neural Comp. & App.*, Vol. 13, No. 2, pp. 130-139, 2004.

[8] K. Chung, K. Wong, and C. Fung, "Reducing User Log Size in an Inter-Query Learning CBIR System with a Cluster Merging Approach," *Proc. of ICNN*, pp. 1184-1191, 2006.

[9] X. Zhou and T. Huang, "Small Sample Learning during Multimedia Retrieval using BiasMap," *Proc. CVPR*, pp. 11-17, 01.

[10] J. He, M. Li, H. Zhang, H. Tong, and C. Zhang, "Generalized Manifold-Ranking-Based Image Retrieval," *IEEE Trans. IP*, Vol. 15, No. 10, pp. 3170-3177, 2006.