DYNAMIC SEMANTIC FEATURE-BASED LONG-TERM CROSS-SESSION LEARNING APPROACH TO CONTENT-BASED IMAGE RETRIEVAL

Zhongmiao Xiao[a], Matthew J. Clark[b], KokSheik Wong[c], and Xiaojun Qi[a]

[a]Zhongmiao.Xiao@aggiemail.usu.edu and Xiaojun.Qi@usu.edu
Department of Computer Science, Utah State University, Logan, UT 84322-4205
[b]clarkm11@up.edu
Department of Computer Science, University of Portland, Portland, OR, 97203
[c]koksheik@um.edu.my
Faculty of Computer Science & Information Technology, University of Malaya, Malaysia

ABSTRACT

This paper proposes a novel content-based image retrieval technique, which facilitates short-term (intra-query) and long-term (inter-query) learning processes by integrating accumulated users' historical relevance feedback-based semantic knowledge. The history is efficiently represented as a dynamic semantic feature of the images. As such, the high-level semantic similarity measure can be dynamically adapted based on the semantic relevance derived from the dynamic semantic features. The short-term relevance feedback technique can benefit from long-term learning. Our extensive experiments show that the proposed system outperforms three peer systems in the context of both correct and erroneous relevance feedback.

Index Terms— CBIR, dynamic semantic feature, cross-session learning, inter-query learning, relevance feedback

1. INTRODUCTION

Relevance feedback (RF) techniques [1] have been widely used in content-based image retrieval (CBIR) systems to formulate the query in an interactive process, bridge the semantic gap, and improve the retrieval performance. However, most existing RF techniques use short-term (intra-query) learning to handle query formulation in a single retrieval session. Recently, long-term (inter-query) learning extends short-term learning by studying the accumulated feedback history collected from multiple query sessions to derive the semantic meaning of database images.

Long-term learning can be roughly classified into retrieval pattern-based learning and feature vector model-based learning. Retrieval pattern-based learning is to establish the relationship between the current and previous query sessions by analyzing retrieval patterns between the sessions. If the two sessions have similar image retrieval patterns, this learning technique assumes that the user must be searching semantically similar images and therefore return images with similar retrieval patterns as the retrieval results. On the contrary, feature vector model-based learning is to bring the feature vectors of similar images close to each other by a weighting or transform scheme. Retrieval pattern-based learning is generally more effective than feature vector model-based learning since the retrieval pattern approximately represents the semantics of images from users' perspective. Here, we briefly review several representative retrieval pattern-based learning techniques.

Heisterkamp [2] applies the latent semantic analysis method on the term-by-document matrix to learn a generalization of the relationship between the current query and the search history. He *et al.* [3] use the semantic space to store retrieval patterns (labels of relevant and irrelevant images) of all query sessions and apply dot product to find semantically similar images. Han *et al.* [4] uses the memory learning technique to compute the ratio of co-positive-feedback frequency and co-feedback frequency for analyzing the relationship among query sessions. A knowledge memory model is then formed to store semantic information and learn semantic relations. Hoi *et al.* [5] apply the statistical correlation on the retrieval log to analyze the relationship between current and past retrieval sessions. Yin *et al.* [6] design a virtual-features-based technique to digest the long-term feedback history to estimate the semantic relevance between images. Although these learning techniques achieve impressive retrieval results, they require a relatively large matrix to store historical feedback information. The matrix may be sparse if queries fall into a few semantic categories, which may deteriorate the learning performance. Moreover, erroneous feedback may also lead to the storage of incorrect information and degrade the overall retrieval performance.

In this paper, we propose a novel long-term cross-session learning scheme for CBIR. First, we integrate low-level visual features (LVFs) and high-level dynamic semantic features (DSFs) in short-term learning to formulate the query in a single retrieval session. Second, we build an adaptive semantic matrix in long-term learning to store

retrieval patterns (i.e., similarity of relevant and irrelevant images) of historical query sessions. Third, we extract DSFs of database images and update DSFs of the query image in each RF iteration step by reinforcing semantically relevant features and suppressing semantically irrelevant features using DSFs of positively and negatively labeled images. Fourth, we apply integrated similarity measure to estimate the semantic relevance between images and return top retrieved images. The rest of the paper is organized as follows: Section 2 presents our proposed learning approach. Section 3 compares our system with three peer systems. Section 4 draws conclusions and presents future directions.

2. DYNAMIC SEMANTIC FEATURE-BASED LONG-TERM CROSS-SESSION LEARNING

The proposed DSF learning system aims to accurately represent each image using high-level semantic features by capturing the intention of multiple users. Semantic concepts are learned from prior intra- and inter-query sessions. In intra-query sessions, the system uses the user's RF within a query session to update the support values of concepts being sought by the user. In inter-query sessions, the system uses previous retrieval experiences of multiple users to update the DSFs of positively and negatively labeled images. Our system seamlessly combines intra- and inter-query (cross-session) learning to facilitate faster and more accurate learning.

2.1. Overview of the Proposed CBIR System

Each image in our system is represented by both LVFs and DSFs. The 100-D LVFs consist of 64-bin HSV color histogram, 9 color (first three moments in HSV), 18 edge (18-bin edge histogram of the converted grayscale image), and 9 texture (entropy of each of nine wavelet detail subbands of the grayscale image). Initially, the DSFs of each image are empty since no knowledge has been learned. The DSFs are updated after each query session.

The dissimilarity between two images, I_i and I_j, is computed by combining the semantic and visual dissimilarity of two images as follows:

$$\begin{aligned} DisSim(I_i, I_j) &= 1 - HighSim(I_i, I_j) + LowDisSim(I_i, I_j) \\ &= 1 - DSF(I_i) \bullet DSF(I_j) + d(LVF(I_i), LVF(I_j)) \end{aligned} \quad (1)$$

where $LVF(I_i)$ represents LVFs of an image I_i, $DSF(I_i)$ represents DSFs of an image I_i, $d(LVF(I_i), LVF(I_j))$ represents the Euclidean distance between LVFs of I_i and I_j, and $DSF(I_i) \bullet DSF(I_j)$ represents the dot product between DSFs of I_i and I_j. Here, the smaller dissimilarity indicates the higher similarity between the two images.

Before any online image retrieval process starts, each database image is represented by the empty DSF and the 100-D LVFs. The number of semantic concepts learned is 0 (i.e., *SemCount*=0). This value serves as the index of a particular relevant semantic concept. It starts from 0 (indicating no relevant semantic concept has been learned) and is incremented by 1 every time a query is made (indicating a new relevant semantic concept has been learned based on the current query). In our system, this value is also used to expand the dimensionality of DSFs of database images. When a user starts the online image retrieval process, the learning process for any query image $q(t)$ (i.e., query image q at the t^{th} iteration) is as follows:

1. Add *SemCount* by 1.
2. Apply Eq. (1) to compute the dissimilarity measure between $q(t)$ and each database image D_i.
3. Return top n images, which have the smallest dissimilarity measures to $q(t)$.
4. While the user is not satisfied with the retrieval results, perform the following operations:
   - 4.1. Allow the user to label relevant (positive) images from the returned pool while treating non-labeled images as irrelevant (negative).
   - 4.2. Call **UpdateDSF** (inter-query learning function as explained in section 2.2) to update DSFs of negatively labeled images at current iteration and accumulated positively labeled images in the current query session.
   - 4.3. Call **UpdateQueryDSF** (intra-query learning function as explained in section 2.3) to update $DSF(q(t+1))$ using DSFs of positively and negatively labeled images.
   - 4.4. Call **UpdateQueryLVF** (intra-query learning function as explained in section 2.3) to update $LVF(q(t+1))$ using LVFs of positively labeled images.
   - 4.5. Apply Eq. (1) to compute $DisSim(q(t+1), D_i)$ between $q(t+1)$ and all images D_i's.
   - 4.6. Return top n images, which have the smallest dissimilarity measures to $q(t+1)$.

Based on the above learning and retrieval process for a query, we clearly observe the following: 1) Our method provides a framework for integrating intra- and inter-query RF-based learning techniques in a single retrieval system. 2) Our method dynamically adjusts the distance between the query and database images based on query's updated DSFs and LVFs derived from both intra- and inter-query RF-based learning, and DSFs of database images derived from the cross-session-based RF history.

2.2. Inter-Query Learning

The objective of inter-query learning is to derive DSFs of database images by capturing the intention of multiple users. It uses the **UpdateDSF** function to dynamically store and update historical RF experiences from multiple users to capture more accurate semantic features. Specifically, **UpdateDSF** function propagates the learned DSFs of positively labeled images to other positively labeled images, whose DSFs are empty. The update strategies are guided by

the observation that all positively labeled images should have similar semantic concepts. The algorithmic view of **UpdateDSF** function is summarized as follows:

1. Let *Pos* denote the set of accumulated positive images in the current query session and *Neg* denote the set of negative images in the current iteration.
2. Collect positive images with non-empty DSFs in a set P_1.
$$P_1 = \{Im_i \mid Im_i \in Pos \text{ and } DSF(Im_i) \neq \phi\} \quad (2)$$
3. If $P_1 \neq \phi$, update DSFs for $\forall PosIm_j \in Pos$ by:
$$DSF(PosIm_j) = \frac{1}{|P_1|} \sum_{Im_i \in P_1} DSF(Im_i) \quad (3)$$
where $|P_1|$ denotes the number of images in set P_1.
4. Expand DSFs for $\forall PosIm_j \in Pos$ by adding a new semantic concept to its current DSFs using:
$$DSF(PosIm_j) \leftarrow DSF(PosIm_j) \cup SemCount(1) \quad (4)$$
where $\cup$ denotes the appending operation, and *SemCount*(1) denotes the operation to put a value of 1 in the dimension specified by *SemCount*.
5. Expand DSFs for $\forall NegIm_j \in Neg$ by adding a new semantic concept to its current DSFs using:
$$DSF(NegIm_j) \leftarrow DSF(NegIm_j) \cup SemCount(-1) \quad (5)$$
where *SemCount*(-1) denotes the operation to put a value of -1 in the dimension specified by *SemCount*.

It should be noted that *SemCount* is incremented by 1 after each query session. Correspondingly, the dimensionality of DSFs of database images is expanded by 1.

2.3. Intra-Query Learning

The objective of intra-query learning is to update query's DSFs using user positively and negatively labeled images during RF iterations. It also updates query's LVFs by moving the query vector towards the subspace that contains more relevant images. Two functions, **UpdateQueryDSF** and **UpdateQueryLVF**, are designed to achieve these two objectives, respectively.

UpdateQueryDSF function is to update query's DSFs by reinforcing semantically relevant features and suppressing semantically irrelevant features using DSFs of both positively and negatively labeled images. Specifically, using positively labeled images, the i^{th} element of query's DSFs at iteration t+1, i.e., $DSF(q(t+1, i))$, is updated by the following rules:

$$\begin{aligned}
&\text{if } DSF(PosIm,i) > 0 \text{ and } DSF(q(t,i)) > 0 \\
&\quad DSF(q(t+1,i)) = \alpha * DSF(q(t,i)) * DSF(PosIm,i) \\
&\text{if } (DSF(PosIm,i) > 0 \text{ and } DSF(q(t,i)) \leq 0) \\
&\quad \text{or } (DSF(PosIm,i) < 0 \text{ and } DSF(q(t,i)) \geq 0) \\
&\quad DSF(q(t+1,i)) = DSF(q(t,i)) + DSF(PosIm,i) \\
&\text{if } DSF(PosIm,i) < 0 \text{ and } DSF(q(t,i)) < 0 \\
&\quad DSF(q(t+1,i)) = \alpha * DSF(q(t,i)) * (-DSF(PosIm,i))
\end{aligned} \quad (6)$$

where $DSF(PosIm, i)$ corresponds to the i^{th} element of DSFs of a positively labeled image, $DSF(q(t, i))$ corresponds to the i^{th} element of DSFs of the query image q at the tth iteration, and the parameter α is the learning adjustment rate and is empirically set to be 1.2. Here, when a positively labeled image and the query image share the same type of semantic concepts (i.e., the values of their semantic features have same signs), we increase the magnitude of the i^{th} element of query's DSFs using both α and the i^{th} element of DSFs of the positively labeled image (refer to the 1st and 3rd conditions). Otherwise, we decrease the magnitude of the i^{th} element of query's DSFs using the i^{th} element of DSFs of the positively labeled image (refer to the 2nd condition).

Similarly, using negatively labeled images, the i^{th} element of query's DSFs at iteration t+1 is updated by:

$$\begin{aligned}
&\text{if } (DSF(NegIm,i) < 0 \text{ and } DSF(q(t,i)) > 0) \\
&\quad \text{or } (DSF(NegIm,i) > 0 \text{ and } DSF(q(t,i)) \leq 0) \\
&\quad DSF(q(t+1,i)) = \alpha * DSF(q(t,i)) \\
&\text{if } (DSF(NegIm,i) < 0 \text{ and } DSF(q(t,i)) \leq 0) \\
&\quad \text{or } (DSF(NegIm,i) > 0 \text{ and } DSF(q(t,i)) > 0) \\
&\quad DSF(q(t+1,i)) = DSF(q(t,i)) / \alpha
\end{aligned} \quad (7)$$

where $DSF(NegIm, i)$ corresponds to the i^{th} element of DSFs of a negatively labeled image. Here, we reduce the magnitude of the i^{th} element of query's DSFs when a negatively labeled image and the query image share the same type of semantic concepts (i.e., the values of their semantic features have the same signs) and increase the magnitude of the i^{th} element of DSFs otherwise. These two update strategies are guided by the following observations: 1) The query's DSFs should have similar semantic concepts as positively labeled images. 2) The query's DSFs should not have semantic concepts associated with negatively labeled images.

UpdateQueryLVF function is to update LVFs of the query using LVFs of all positively labeled images by:

$$LVF(q(t+1)) = \frac{1}{|Pos|} \sum_{Im_i \in Pos} LVF(Im_i) \quad (8)$$

where $LVF(q(t+1))$ denotes LVFs of the query image q at t+1th iteration, $LVF(Im_i)$ denotes LVFs of a positively labeled image Im_i, and $|Pos|$ denotes the number of positively labeled images.

3. EXPERIMENTAL RESULTS

We tested our CBIR system on three data sets: 2000-Flickr DB, 6000-COREL DB, and the combined 2000-Flickr and 6000-COREL DB. Flickr and COREL DBs contain 20 and 60 categories with 100 images per category, respectively.

We first designed three experiments on the 2000-Flickr DB by applying an automatic feedback scheme to perform the iterative retrieval process. 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 positive images to total returned images (e.g., 25 in our

system). We randomly chose 2%, 5%, and 10% of database images as queries to construct three adaptive semantic matrices in cross-session learning, respectively. Another three experiments were performed to incorporate the possible erroneous feedback in the real-world RF process, wherein erroneous feedback may result from user inherent subjectivity or laziness. Fig. 1 shows the average retrieval accuracy for 1800 Flickr images using different adaptive semantic matrices as learning bases. It clearly shows the retrieval accuracy is improved when more accurate DSFs of database images are learned for a larger learning base. The retrieval accuracy on the largest learning base is above 90% after the 1st and the 2nd iterations in the context of correct and erroneous RF, respectively. Therefore, we chose 10% of database images to construct the adaptive learning base for future online retrieval.

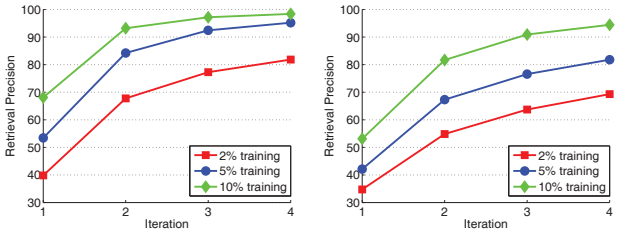

Fig. 1: Retrieval performance on 2000-Flickr DB using different number of queries and using correct (left) and 5% erroneous RF (right).

We compared our system with Han's memory learning [4], Hoi's log-based [5], and Yin's virtual-feature-based systems [6] on two larger DBs. Fig. 2 and Fig. 3 show the average retrieval precision of four systems on 6000-COREL DB and the combined DB after using 10% of database images to build their perspective learning bases, respectively. Our system clearly achieves the best precision when correct and erroneous feedback is involved. Comparing to the 2nd best system in the context of correct feedback, our system makes 4.32% and 1.04% improvement on 6000-COREL, and 5.45% and 0.20% improvement on the combined DB for the last two iterations, respectively. It achieves accuracy of 92.79% and 94.57% on 6000-COREL, and 82.40% and 85.80% on the combined DB for the last two iterations, respectively. In the context of erroneous feedback, our system makes 13.71% and 5.21% improvement on 6000-COREL, and 17.44% and 7.11% improvement on the combined DB for the last two iterations, respectively. It achieves accuracy of 83.28% and 86.48% on the 6000-COREL, and 69.80% and 74.10% on the combined DB for the last two iterations, respectively. As a result, our system is more resilient to erroneous feedback. This feature results from robust cross-session learning and accurate DSFs. Our retrieval time is also comparable with its peers due to the simple update process.

4. CONCLUSIONS AND FUTURE WORK

We propose a novel DSF-based long-term cross-session learning approach for CBIR. Major contributions are: 1) Integrate LVFs and DSFs in short-term learning to formulate the query in a single retrieval session. 2) Build an adaptive semantic matrix in cross-session learning to store similarity of relevant and irrelevant images of historical query sessions. 3) Update DSFs of the query image by reinforcing semantically relevant features and suppressing semantically irrelevant features. 4) Apply the integrated similarity measure to estimate the semantic relevance between images.

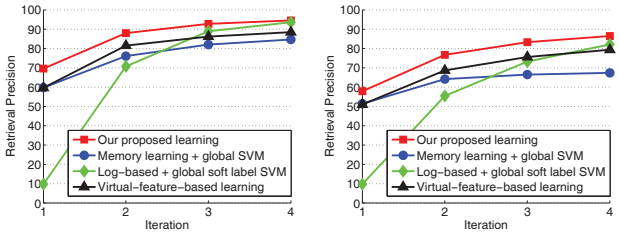

Fig. 2: Comparison of four systems on 6000-COREL DB using correct (left) and 5% erroneous RF (right).

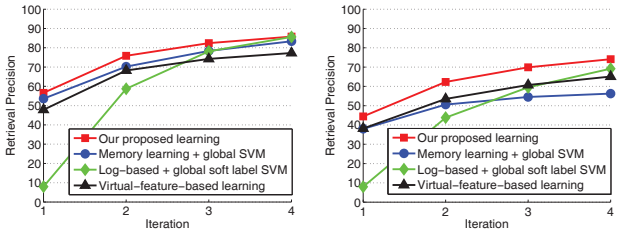

Fig. 3: Comparison of four systems on the combined DB using correct (left) and 5% erroneous RF (right).

Experimental results show our system outperforms peer systems considered and achieves highest retrieval accuracy in all iterations in terms of correct and erroneous feedback. Other compact forms of the adaptive semantic matrix will be investigated in future research.

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. R. Heisterkamp, "Building a Latent Semantic Index of an Image Database from Patterns of Relevance Feedback," *Proc. of Int. Conf. on Pattern Recognition*, 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] J. Han, K. N. Ngan, M. Li, and H. J. Zhang, "A Memory Learning Framework for Effective Image Retrieval," *IEEE Trans. Image Processing*, Vol. 14, No. 4, pp. 511-524, 2005.

[5] 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.

[6] P. Y. Yin, B. Bhanu, K. C. Chang, and A. Dong, "Long-term Cross-Session Relevance Feedback Using Virtual Features," *IEEE Trans. KDE*, Vol. 20, No. 3, pp. 352-368, 2008.