A SHORT-TERM AND LONG-TERM LEARNING APPROACH FOR CONTENT-BASED IMAGE RETRIEVAL

Michael Wacht[1], Juan Shan[2], and Xiaojun Qi[2]

[1]michael.wacht@gmail.com
Computer Science Department, Slippery Rock University, Slippery Rock, PA 16057
[2]juans@cc.usu.edu and xqi@cc.usu.edu
Computer Science Department, Utah State University, Logan, UT 84322-4205

ABSTRACT

This paper proposes a short-term and long-term learning approach for content-based image retrieval. The proposed system integrates the user's positive and negative feedback from all iterations to construct a semantic space to remember the user's intent in terms of the high-level hidden semantic features. The short-term learning further refines the query by updating its associated weight vector using both positive and negative examples together with the long-term-learning-based semantic space. The similarity score is computed as the dot product between the query weight vector and the high-level features of each image stored in the semantic space. Our proposed retrieval approach demonstrates a promising retrieval performance for an image database of 6000 general-purpose images from COREL, as compared with the conventional retrieval systems.

1. INTRODUCTION

Conventional content-based image retrieval (CBIR) systems [1-4] use automatically extracted low-level features such as color, texture, and shape for retrieval. However, these systems do not learn which features are associated with the semantic meanings and therefore cannot narrow down the semantic gap between low-level visual features and high-level semantics.

Recently, relevance feedback based systems [5-8] are being extensively studied where the semantic features are learned based on users' feedback on the retrieval results. Wu *et al.* [5] use both labeled and unlabeled images to construct a discriminant expectation maximization based transductive learning framework for retrieval. Tong and Chang [6] apply a support vector machine learning algorithm to generate effective relevance feedback for image retrieval. Zhou and Huang [7] propose biased discriminant analysis and transforms to address the asymmetry between the positive and negative examples for more effective retrieval. In general, these systems [5-7] aim at the short-term learning by exclusively refining the low-level features based on the feedback from the current query session. They do not utilize any previous feedback to gather knowledge for further narrowing down the semantic gap. To our knowledge, the semantic-space-based learning system [8] is the only one that accumulates user interactions and integrates both short-term and long-term learning to gradually improve the retrieval performance. However, it is time-consuming to incrementally construct the semantic space. In addition, the semantic space does not integrate any negative examples, which correspond to the failure of the current classifier in learning.

To address the limitations of current CBIR systems, we propose a short-term and long-term learning based CBIR system which integrates both positive and negative feedback. These double fusions make our proposed system achieve promising retrieval accuracy compared with the conventional CBIR systems. In specific, the long-term learning constructs an optimal semantic space by collecting the semantic information obtained from both positive and negative relevance feedback from all interactions for a variety of training query images. This semantic space remembers the user's intent and therefore provides a better representation of each image in the database in terms of the semantic meanings. The short-term learning refines the query by updating its associated weight vector using both positive and negative examples together with the optimal semantic space. Negative feedback, which is handled differently as the positive one, is utilized in our system due to the fact that it contains more information about the irrelevant features.

The remainder of the paper is organized as follows. Section 2 describes our proposed short-term and long-term based CBIR system. Section 3 illustrates the experimental results. Section 4 draws conclusions.

2. THE PROPOSED CBIR SYSTEM

The block diagram of our proposed CBIR system is shown in Fig. 1. The system first computes low-level features of the query image and returns 20 images with the highest similarity scores to the user. The user labels both positive

and negative examples according to the relevance of each retrieved image to the query image. The system then refines the weight vector (i.e., the high-level representation) of the query image by using both the user's feedback and the semantic space. This semantic space stores the high-level semantic features for each image in the database and is constructed off-line by collecting the user-feedback-based semantic information from various training query images. The system returns another set of top 20 retrieved images by using the refined high-level semantic features of the query image. This relevance feedback process together with the query refinement will repeat multiple times until the user is satisfied with the retrieval results. The following subsections will explain each component of our proposed CBIR system in detail.

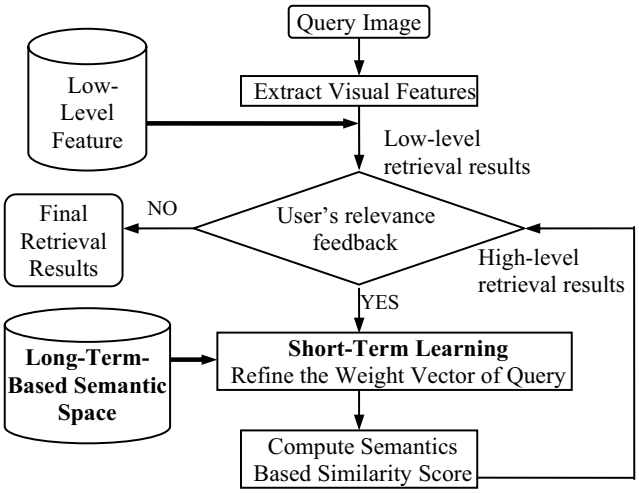

Fig. 1: The block diagram of our CBIR system

2.1. Short-term learning: single query session

Suppose that we already constructed a long-term-based semantic space B, which is a matrix of size $m \times n$, where m is the number of images in the database and n is the number of the hidden high-level semantic features. The retrieval process for a new query image is carried out as follows:

Step 1: Retrieve images using low-level features. We represent the images using 198 low-level features (i.e., 3 64-bin color histograms in the HSV space, and mean and standard deviation of 3 color components in the RGB space). The Euclidian distance is computed to measure the similarity between the query and each image in the database. Top 20 images with the smallest distances are returned as the retrieval results.

Step 2: Represent the query image using high-level features. The user labels both positive and negative examples based on the returned retrieval results. Each labeled example is represented by a semantic vector x^j with $j = 1, \ldots, s$ for the positive examples and $j = s + 1, \ldots, s + t$ for the negative examples. Each semantic vector corresponds to a row vector in B, which is determined by the index number of the labeled examples. As shown in Section 2.2, each row may contain any of the values of 1, -1, and 0. The high-level feature vector of the query is then represented as:

$$Q = (q_1, q_2, ..., q_n) \tag{1}$$

where n is the length of the feature vector and equals the number of columns in B. Each element q_i is computed as:

$$q_i = (x_i^1 \vee x_i^2 \vee \cdots \vee x_i^s) \wedge \overline{(x_i^{s+1} \vee x_i^{s+2} \vee \cdots \vee x_i^{s+t})} \tag{2}$$

where x_i^j is the ith element of the semantic vector x^j. In this calculation, we treat all negative values as 0's.

Step 3: Retrieve images using high-level features. Based on the high-level feature vector Q, the system recalculates the similarity score between the query and each image in the database using the following linear function:

$$h_{score}(W, X) = \sum_{i=1}^{n} w_i x_i \tag{3}$$

where W represents the weight vector associated with the query, X represents the semantic vector of an image in the database, and n is the length of both vectors. Initially, we set $W = Q$. This new measure ensures that the images sharing more high-level semantic features with the query always yield higher scores than those images which share fewer high-level features. Top 20 images with the highest scores are returned as the retrieval results.

Step 4: Update the weight vector associated with the query using both semantic space B and the user's feedback. Based on the user's feedback on the top 20 images returned from Step 3, the weight vector is updated as follows:

- Positive Feedback

$$w_i^{(t+1)} = \begin{cases} \alpha w_i^{(t)} & \text{if } x_i = 1 \text{ and } w_i^{(t)} \neq 0 \\ 1 & \text{if } x_i = 1 \text{ and } w_i^{(t)} = 0 \\ w_i^{(t)} & \text{if } x_i = 0 \\ w_i^{(t)} / \alpha & \text{if } x_i = -1 \end{cases} \tag{4}$$

- Negative Feedback

$$w_i^{(t+1)} = \begin{cases} \alpha w_i^{(t)} & \text{if } x_i = -1 \text{ and } w_i^{(t)} \neq 0 \\ 1 & \text{if } x_i = -1 \text{ and } w_i^{(t)} = 0 \\ w_i^{(t)} & \text{if } x_i = 0 \\ w_i^{(t)} / \alpha & \text{if } x_i = 1 \end{cases} \tag{5}$$

where $w_i^{(t)}$ is the ith element of the current weight vector, $w_i^{(t+1)}$ is the ith element of the updated weight vector, x_i is the ith element of the hidden semantic feature vector of the labeled image x, and α is the adjustment rate, which is empirically set to be 1.1. The update is repeated until all labeled positive and negative images are processed.

Step 5: Repeat Steps 3 and 4 until the user is satisfied with the retrieval results. The similarity scores between the query and each image in the database are recomputed using (3), where W is the final weight vector yielded from the previous step. Top 20 images with the highest similarity

scores are returned to the user. If the user is satisfied with the retrieval results, the query session is finished. Otherwise, the system asks the user to give feedback to update the query weight vector and next retrieval iteration starts until the user is satisfied with the retrieval results.

2.2. Long-term learning: semantic space

We adopt the vector space model [9] to represent the long-term-learning-based semantic space. This semantic space B is a matrix of size $m \times n$, where m is the number of images in the database and n is the number of the hidden high-level semantic features. That is, each row in the semantic space represents an image in the database and each column indicates the presence of a certain semantic feature for each image in the database. In our proposed system, we set $n = 0.12 \times m$ since impressive retrieval accuracy can be achieved using this reasonable semantic space size. The algorithmic view of constructing the semantic space is shown in Fig. 2.

1. Set the semantic space B to be empty.
2. Randomly select 12% of the images in the database as training images, where approximately equal number of images is chosen from each category.
3. Randomly choose the first training image and retrieve top 20 images using low-level features.
4. Label both positive and negative examples based on the returned retrieval results.
5. Add the first column to B such that the elements corresponding to the rows of the positive and negative examples are respectively set to 1 and -1, and the remaining elements are set to 0.
6. For each remaining training image i
   - 6.1 Apply steps 1 through 5 in subsection 2.1 and record all the positive and negative examples labeled at each feedback process.
   - 6.2 Add a new column to B such that the elements corresponding to the rows of all positive and negative examples are respectively set to 1 and -1, and the remaining elements are set to 0.

Fig. 2: The algorithm of constructing the semantic space

3. EXPERIMENT RESULTS

To date, we have tested our CBIR system on a general-purpose image database with 6000 images from COREL. These images have 60 categories with 100 images in each category. The categories contain different semantics, namely building, beach, horse, mountain, and the like. A retrieved image is considered to be correct if it belongs to the same semantic category as the query image. To facilitate the evaluation process, we design an automatic feedback scheme to model the short-term single query session, which consists of 8 interactions. For each interaction, the system automatically labels the images as positive examples if they are in the same semantic category as the query and labels the others as negative examples. The retrieval accuracy is computed as the ratio of the relevant retrieved images over the total retrieved images. Four experiments have been specifically designed to evaluate our proposed system.

Experiment 1: The choice of the optimal semantic space size. Different training sets, i.e., 3%, 6%, 12%, and 24% of the 2000 images of 20 categories in the database, are used to construct the semantic spaces. The remaining non-training images from each category are used as queries to ensure the fair comparison of the retrieval accuracy upon different semantic spaces. Fig. 3 shows the average retrieval accuracy on different semantic spaces at each of the 8 iterative processes. It clearly demonstrates that the retrieval accuracy increases as the size of the semantic space increases and each iteration leads to better retrieval accuracy. However, 12% and 24% semantic spaces yield comparable accuracy at each iteration and converge to almost the same accuracy after the 6th iteration. Consequently, we choose 12% semantic space as the optimal semantic space for saving the storage space and reducing the time in calculating the similarity scores for online retrieval.

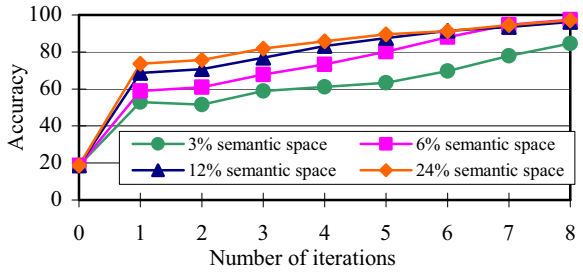

Fig 3: Retrieval accuracy upon different semantic spaces

Experiment 2: The choice of the relevance feedback. In addition to the optimal semantic space constructed in experiment 1, we also construct another 12% semantic space by exclusively using positive examples. That is, we label 1 for the relevant (i.e., positive) images and label 0 for the rest of the images in the semantic space. Fig. 4 shows the average retrieval accuracy upon these two semantic spaces at each iterative process. It clearly shows that the accuracy improves by almost 1.5 times for each iteration after including the negative feedback.

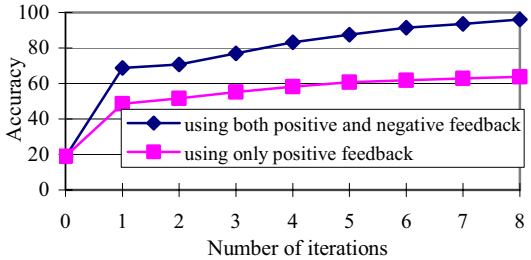

Fig. 4: Retrieval accuracy using different feedback

Experiment 3: The sensitivity to the database size. The scalability of the method is tested by performing the retrieval experiments over different databases. A total of 6 data sets are used. The number of categories in a data set varies from 10 to 60 with a step size of 10. For each data set, we construct a 12% semantic space using both positive and negative feedback. Fig. 5 shows the average retrieval results at all 8 iterations for all data sets. We observe a decrease in average retrieval accuracy as the database size increases. However, the average accuracy consistently improves after each iteration no matter the size of the database. Moreover, our system achieves an impressive accuracy of 92.81% after 8 iterations for 6000 images with 60 categories.

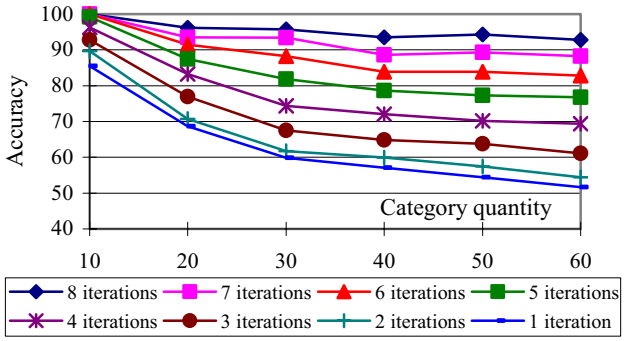

Fig. 5: Average retrieval accuracy for different databases

Experiment 4: Comparisons with other CBIR systems. We compare our proposed approach with two CBIR systems, namely the unified feature matching (UFM) [3] and the Fusion method [4]. Both systems are considered the best retrieval systems without using relevance feedback. The other relevance feedback based systems are not chosen for comparisons mainly due to the unavailability of the common data sets and the executables. The experiment is performed by using the same test images from the same 1000-image database with 10 semantic categories. The average retrieval accuracy for each category is shown in Fig. 6. We observe that our approach outperforms both UFM and Fusion approaches after 4 iterations in 3 perspectives. 1) It achieves 100% accuracy for 6 categories at the 4th iteration. The improvements of the overall accuracy over the UFM and Fusion approaches are 38.42% and 24.05%, respectively. 2) It achieves 100% accuracy for 8 categories at the 5th iteration. It outperforms the UFM and Fusion approaches by 42.63% and 27.82%, respectively, in terms of the overall accuracy. 3) It successfully retrieves beach images (category 2) and mountain images (category 9), which are similar in terms of the low-level features since both beach and mountain images may contain a large area of blue sky. These observations indicate that the integration of the high-level semantic features by using users' positive and negative feedback does improve the system performance.

4. CONCLUSION

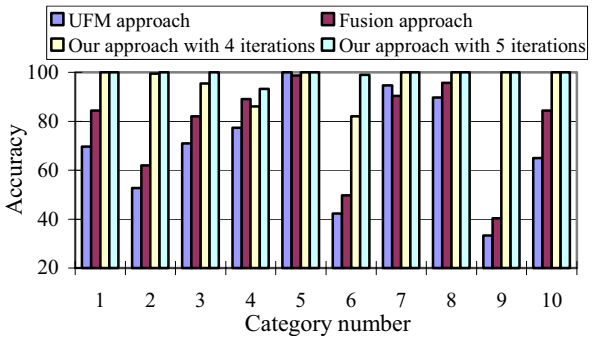

Fig. 6: Comparisons with other CBIR systems

A novel relevance feedback based CBIR system is proposed in this paper. The major contributions are: 1) Construct a long-term-learning based semantic space to record the user's positive and negative feedback. 2) Represent each image in the database using the high-level hidden semantic features learned from the user's feedback. 3) Integrate short-term-learning with long-term-learning to semantically update the query weight vector. Experiment results show that our system is not sensitive to the database size and outperform two best non-relevance-feedback-based retrieval systems. Furthermore, it always achieves high retrieval accuracy for a large database which contains similar semantic categories.

The singular value decomposition method will be considered to improve the efficiency of the semantic space. A variety of heuristics will be studied to permit the system to return the retrieval images from different ranges of scores.

5. REFERENCES

[1] M. Flickner, *et al.*, "Query by Image and Video Content: the QBIC System," *IEEE Computer*, Vol. 28, pp. 23-32, Sept. 1995.

[2] W. Y. Ma and B. S. Manjunath, "Netra: A Toolbox for Navigating Large Image Databases," *ACM Multimedia Syst.*, Vol. 7, pp. 184–198, 1999.

[3] Y. Chen and J. Wang, "A Region-Based Fuzzy Feature Matching Approach to CBIR," *IEEE Trans. PAMI*, Vol. 23, No. 9, pp. 1252-1267, 2002.

[4] X. Qi and Y. Han, "A Novel Fusion Approach to CBIR," *Pattern Recognition*, Vol. 38, No. 12, pp. 2449-2465, 2005.

[5] Y. Wu, Q. Tian, and T. S. Huang, "Discriminant EM Algorithm with Application to Image Retrieval," in *Proc. IEEE Conf. CVPR*, Vol. 1, pp. 222-227, June 2000.

[6] S. Tong and E. Chang, "Support Vector Machine Active Learning for Image Retrieval," in *Proc. ACM Multimedia*, Ottawa, ON, Canada, pp. 107-118, Sept. 2001.

[7] X. S. Zhou and T. S. Huang, "Small Sample Learning During Multimedia Retrieval Using BiasMap," in *Proc. IEEE CVPR*, Kauai, HI, Vol. 1, pp. 111-117, Dec. 2001.

[8] Xiaofei He, *et al.* "Learning a Semantic Space From User's Relevance Feedback for Image Retrieval," IEEE *Transactions CSVT*, Vol. 13, No. 1, pp. 39-48, Jan. 2003.

[9] G. Salton and M. McGill, *Introduction to Modern Information Retrieval*. New York: McGraw-Hill, 1983.