A FUZZY COMBINED LEARNING APPROACH TO CONTENT-BASED IMAGE RETRIEVAL

Samuel Barrett[a], Ran Chang[b], and *Xiaojun Qi[b]*

thesambarrett@gmail.com, ran.c@aggiemail.usu.edu and Xiaojun.Qi@usu.edu
[a]Computer Science Department, The University of Texas at Austin, TX 78712
[b]Computer Science Department, Utah State University, Logan, UT 84322-4205

ABSTRACT

We propose a fuzzy combined learning approach to construct a relevance feedback-based content-based image retrieval (CBIR) system for efficient image search. Our system uses a composite short-term and long-term learning approach to learn the semantics of an image. Specifically, the short-term learning technique applies fuzzy support vector machine (FSVM) learning on user labeled and additional chosen image blocks to learn a more accurate boundary for separating the relevant and irrelevant blocks at each feedback iteration. The long-term learning technique applies a novel semantic clustering technique to adaptively learn and update the semantic concepts at each query session. A predictive algorithm is also applied to find images most semantically related to the query based on the semantic clusters generated in the long-term learning. Our extensive experimental results demonstrate the proposed system outperforms several state-of-the-art peer systems in terms of both retrieval precision and storage space.

Index Terms— Content-based image retrieval, short-term learning, long-term learning, fuzzy support vector machine learning, semantic clustering technique

1. INTRODUCTION

Relevance feedback techniques have been widely used in CBIR to bridge the semantic gap between low-level features and high-level semantics. They allow the user to label the returned images as relevant or irrelevant. Such labeled examples are further used to refine retrieval results by short-term learning and/or long-term learning techniques.

Short-term learning techniques aim to find out which images are relevant to the user's query over the course of a single query. They include query updating [1, 2, 3] and machine learning techniques [4, 5, 6]. However, query updating techniques cannot capture semantics (e.g., mountain, beach, etc.) by updating the query concept using low-level features. Machine learning techniques cannot achieve good and reliable classification by using a small number of imbalanced feedback examples. Furthermore, the semantic knowledge obtained in the entire feedback process is not remembered. To overcome the above shortcomings, long-term learning techniques have been proposed to find out the relationships among images over the course of multiple queries. They include the statistical correlation [7], semantic space-based [8], log-based [9], and memory learning techniques [10]. However, all these techniques require a large matrix to store memorized feedback information. The sparsity of the matrices may also make learning not useful for a large-scale database. Erroneous feedback further leads to store incorrect semantic information and degrade the retrieval performance.

To address the limitations of current CBIR systems, we propose a composite learning method to bridge the semantic gap. We first apply the FSVM-based short-term learning technique to learn a more accurate boundary for separating the user labeled relevant and irrelevant blocks at each feedback iteration. Specifically, we divide an image into five blocks and apply the k-means algorithm to choose additional blocks from non-returned database images. We apply a fuzzy metric to assign a weight to each additional block to indicate the predicted accuracy of its relevancy label. These additional blocks, together with their labels and weights, enlarge the training set for FSVM learning and therefore improve the retrieval accuracy of short-term learning. We then apply the long-term learning technique to remember the feedback information in semantic clusters. Specifically, we apply a semantic clustering technique to group the user labeled images into appropriate semantic clusters. We apply a merging method to compactly store the memorized feedback information and use a predictive algorithm to find images most semantically related to the query image. We finally combine the retrieval results from both learning techniques to improve the retrieval accuracy. The rest of the paper is as follows: Section 2 presents our proposed learning approach. Section 3 compares our system with peer systems. Section 4 draws conclusions.

2. THE FUZZY COMBINED LEARNING APPROACH

The retrieval process of our system is as follows: The user first supplies a query image q. The system then returns top

n images, which are classified by the user as relevant or irrelevant to q. This process continues for a few iterations, or until the user is satisfied with the retrieval results. For each iteration, the system returns top n images based on the similarity scores $S(q, D_i)$ between q and an arbitrary image D_i in the database. Specifically, $S(q, D_i)$ is computed by:

$$S(q, D_i) = w_{short} \cdot S_{short}(q, D_i) + w_{long} \cdot S_{long}(q, D_i) \quad (1)$$

where $S_{short}(q, D_i)$ and $S_{long}(q, D_i)$ respectively measure the short-term and long-term-based similarity scores between q and D_i; w_{short} and w_{long} are their corresponding weights.

2.1. Short-term-based Low-level Retrieval

For the initial retrieval, we use the expanded MPEG-7 150-bin edge histogram descriptor (EHD) and the 64-bin HSV-based scaled color descriptor (SCD) to extract global low-level features. The Euclidean distances between EHDs and SCDs of q and D_i are computed to measure $S_{short}(q, D_i)$.

In the following iterations, we search for additional blocks from non-returned database images and compute their labels and weights to enlarge the training set for improving FSVM learning. Here, the Gaussian radial basis function (RBF) kernel is used. We use the blocking scheme in [11] to divide each image into five blocks. That is, each image is horizontally divided into three blocks and the middle block is further evenly divided into three sub-blocks. The algorithmic overview of our short-term framework is:

1. Compute $S_{short}(q, D_i)$, which measures the distance between q and D_i in the global low-level feature space.
2. Return top n images similar to q based on $S_{short}(q, D_i)$.
3. The user labels n images as relevant or irrelevant to q.
4. Train an SVM classifier using five blocks of each of the user labeled n images, where each block retains the same label (relevant or irrelevant) of its parent image.
5. Apply the k-means algorithm to separately cluster relevant and irrelevant blocks, where k is experimentally set to be 8.
6. For each of the eight clusters of relevant blocks, choose five additional unlabeled blocks, which are closest to the cluster center, from non-returned database images to expand the positive training set. The negative training set is expanded similarly.
7. Assign each additional unlabeled block x_a a label, which is the same as its associated cluster.
8. Evaluate the relevance membership (i.e., the predicted accuracy of the relevancy label) of each block x_a as:
   a. Calculate $r(x_a)$, the ratio between the distance of x_a to the nearest cluster center of the same label and the distance of x_a to the nearest cluster center of the opposite label.
   b. Set $w_1(x_a)$, a measure of the correctness of the assigned label, to 0 if $r(x_a) \geq 1$. Otherwise, set $w_1(x_p)$ to $e^{-(a_1 r(x_a))}$, where a_1=1 is a scaling factor.
   c. Set $w_2(x_p)$, a measure of how well the assigned label agrees with the label determined by the trained SVM, to $1/(1+e^{-a_2 y})$ if the two labels match. Otherwise, set $w_2(x_p)$ to $1/(1+e^{a_2 y})$, where a_2 = 1 is a scaling factor and y is the distance from x_a to the SVM boundary.
   d. Compute the final predicted weight of x_a as
   $$g(x_a) = w_1(x_a) w_2(x_a) \quad (2)$$
9. Train a FSVM using user labeled and additional blocks, where the weights of user positively and negatively labeled blocks are respectively 1 and -1, and the weight of each block x_a is computed by (2).
10. Add the directed distance from each block of an image D_i to the trained FSVM boundary and normalize the total directed distance to [0, 1]. This normalized total directed distance is used as the similarity score $S_{short}(q, D_i)$ for selecting images most related to the query.
11. Repeat steps 2-10 until the user is satisfied with the retrieval results.

Here, we use a set of compact low-level features [12] (i.e., 9-D color moments, 18-D edges, and 9-D textures) to represent each block to speed up the training and learning process. These features also complement global low-level features to represent an image from different perspectives and achieve better low-level feature-based retrieval results.

2.2. Long-term-based High-level Retrieval

Our system constructs semantic clusters based on users' relevance feedback. Since humans tend to classify objects into semantic categories (e.g., mountain, beach, vehicles, etc.) and remember how well each object belongs to each category, we build our long-term learning framework in the similar manner. If two images are jointly labeled as positive examples in a search session, it is likely that they contain similar semantic content and belong to the same semantic categories. We then construct a cluster to represent such a semantic category and estimate the query semantics by evaluating its relationship to each cluster. We finally compute the semantic similarity between the query image and each database image.

The basic framework of the long-term-based high-level retrieval is as follows: Each query session generates two sets of images: a set relevant to the query and a set irrelevant to the query. These two sets are treated as a new semantic cluster which represents the semantics shared by the relevant images and does not represent the semantics of the irrelevant images. A merging algorithm is then applied to decide whether this new cluster represents the same semantic concept as any existing clusters. If so, the semantic information from the new cluster is added to the existing clusters and the new cluster is removed. Otherwise, the new cluster is added to expand the semantic categories. A predictive algorithm is finally applied to find images most semantically related to the query. We explain both merging and predictive algorithms in detail.

We first measure the fuzzy membership of an image D_i

to a semantic cluster J by:

$$M(D_i, J) = \frac{\text{relevant}(D_i, J)}{\text{occurred}(D_i, J)} \quad (3)$$

where occurred(D_i, J) is the number of times that image D_i has been returned together with the images from cluster J, and relevant(D_i, J) is the number of times that image D_i has been labeled as relevant to the images from cluster J in all the relevance feedback iterations.

We then use this membership to design the merging algorithm. Specifically, we calculate the similarity of two clusters by finding the images that are relevant to both clusters and calculate the dissimilarity of two clusters by finding the images that are relevant to one cluster, but irrelevant to the other cluster. If the cluster's similarity outweighs its dissimilarity, the clusters should be merged. The algorithmic view of this merging method is as follows:

1. Compute the similarity between clusters J and K:

$$Sim(J, K) = \frac{\sum_{i=1}^{N} M(D_i, J) * M(D_i, K)}{N} \quad (4)$$

   where N is the total number of images in the database. This similarity measures the percentage of the database images being in both clusters J and K.

2. Compute the dissimilarity between clusters J and K:

$$DisSim(J, K) = \max\left(\frac{\sum_{i=1}^{N} I(D_i, J) * M(D_i, K)}{N}, \frac{\sum_{i=1}^{N} I(D_i, K) * M(D_i, J)}{N} \right) \quad (5)$$

   where $I(D_i, J)$ defines the irrelevance of image D_i to cluster J. $I(D_i, J)$ is computed as 1-$M(D_i, J)$ if relevant(D_i, J) >0. Otherwise, it is set to be 0. This irrelevance value measures the fuzzy membership of image D_i not being in a semantic cluster J.

3. If $Sim(J, K)$ outweighs $DisSim(J, K)$ by 0.25, clusters J and K are merged by summing occurred(D_i, J) and occurred(D_i, K), and summing relevant(D_i, J) and relevant(D_i, K).

We finally deploy the predictive algorithm to find images most semantically related to the query image. To this end, our system first decides the probability of each semantic cluster representing the query semantics. It next calculates the probability of each database image being a member of each semantic cluster. It finally computes the semantic similarity between the query and each database image as the total of the product of the above two probabilities. Given each database image D_i, a semantic cluster J_k, a positive set *Pos* which contains positively labeled images in a feedback iteration, and a negative set *Neg* which contains negatively labeled images in a feedback iteration, the algorithmic view of this predictive method is:

1. Compute the degree of J_k representing all images in the positive feedback set *Pos* by:

$$related(J_k) = \sum_{D_i \in Pos} relevant(D_i, J_k) \quad (6)$$

2. Compute the degree of J_k representing all images in the negative feedback set *Neg* by:

$$unrelated(J_k) = \sum_{D_i \in Neg} relevant(D_i, J_k) \quad (7)$$

3. Determine the degree of J_k representing the semantic concept of query q by:

$$difference(J_k) = related(J_k) - unrelated(J_k) \quad (8)$$

4. Compute the probability of J_k representing the semantic concept of query q by:

$$P(J_k) = \frac{difference(J_k)}{\max(difference(J_1), \ldots, difference(J_T))} \quad (9)$$

   where T is the total number of semantic clusters.

5. Compute the semantic similarity between query q and each database image D_i as:

$$S_{long}(q, D_i) = \sum_{k=1}^{T} (M(D_i, J_k) * P(J_k)) \quad (10)$$

That is, we assign the highest value to the image which is the most semantically similar to the query.

3. EXPERIMENTAL RESULTS

We tested our CBIR system on a 6000-image Corel database, with 100 images in each of 60 distinct semantic categories such as vehicles, mountain, beach, etc. To facilitate the evaluation process, our system automatically selects query images and performs the relevance feedback process. A retrieved image is automatically classified as relevant if it is in the same semantic category as the query. Two sets of experiments were designed to evaluate the retrieval performance of short-term learning and combined learning by incorporating correct and 5% erroneous feedback, respectively. To introduce the noise, we let the simulated "user" misclassify some relevant images as irrelevant and some irrelevant images as relevant. For the second experimental set, we randomly chose 10% of the database as queries and constructed long-term semantic clusters by performing a query session for each chosen query. After the training, the system was tested using the remaining 5,400 images as queries. No additional semantic clusters were stored during the testing. In each experiment, we performed four feedback iterations with top 25 images returned in each iteration using (1) with w_{Long}=w_{Short}.

3.1. Effectiveness of Short-term Learning

Fig. 1 compares the proposed block-based FSVM learning method with the global SVM learning method, the global FSVM learning method [5], and the block-based SVM learning method, which uses the same blocking scheme as ours. The algorithms are compared in terms of the retrieval precision when correct and erroneous feedback is involved, respectively. Fig. 1(a) shows that our block-based FSVM method consistently achieves the best accuracy, while the global SVM method achieves the worst accuracy in all iterations. The block-based SVM method performs better than the global SVM method in all iterations. It also achieves better retrieval accuracy than the global FSVM method in later iterations. Fig. 1(b) shows our block-based FSVM method is the most resilient against the erroneous feedback due to the small decrease in retrieval precision.

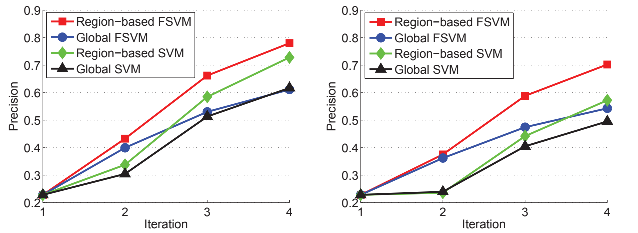

Fig. 1: Comparison of four SVM-based methods in terms of average retrieval precision. Left: with correct relevance feedback; Right: with 5% relevance feedback errors.

3.2. Effectiveness of Fuzzy Combined Learning

We compare the proposed fuzzy combined learning system with the memory learning system [10] and a variant of the proposed system which incorporates the global SVM method as the short-term learning scheme. The three systems were respectively tested with and without a 5% chance of the user mislabeling each returned image.

Fig. 2 compares the average retrieval precision of the three learning systems. It shows that our system achieves the best retrieval accuracy in all iterations with correct or erroneous feedback. Fig. 2(a) demonstrates the variant of our system achieves comparable performance as the memory learning technique when the correct feedback is involved. Therefore, we conclude our short-term learning facilitates the long-term learning to boost the retrieval accuracy in all iterations. Fig. 2(b) demonstrates our system and its variant are resilient to the erroneous feedback, while the memory learning technique significantly drops the retrieval accuracy in all iterations. This resilience is mainly due to the merging of similar semantic clusters. This noise resilience feature is important in the real world since it is normal for users to make mistakes due to the inherent subjectivity, user laziness, or maliciousness.

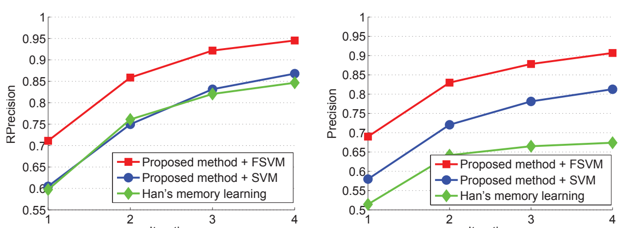

Fig. 2: Comparison of three learning methods in terms of average retrieval precision. Left: with correct relevance feedback; Right: with 5% relevance feedback errors.

Our system performs queries in real time and uses less space to save learned semantic knowledge. It requires O($N \times C$), instead of O($N \times N$) space used in the memory learning technique, where N is the total number of database images and C is the number of semantic clusters ($C << N$).

4. CONCLUSIONS

This paper introduces a noise-resilient fuzzy combined short-term and long-term learning method for CBIR. The proposed system first incorporates additional blocks in FSVM-based short-term learning. It then applies the long-term-based semantic clustering technique to remember the semantic knowledge in clusters and applies the predictive algorithm to compute the high-level semantic similarity between the query and each database image. Our extensive experimental results show that our system achieves the best retrieval accuracy when compared to its variant and its peer (e.g., the memory learning-based system). Our system also requires significantly less space and is more scalable to large databases.

We will explore the potential of applying the proposed technique in image annotation. The semantic clustering technique will be further studied to determine the relations of images. We will also test the proposed technique on a large database, where each image may have multiple semantic meanings, for its effectiveness and scalability.

5. REFERENCES

[1] A. Kushki, P. Androutsos, K. N. Plataniotis, and A. N. Venetsanopoulos, "Query Feedback for Interactive Image Retrieval," *IEEE Trans. CSVT*, Vol. 14, pp. 644-655, 2004.

[2] P. Muneesawang and L. Guan, "An Interactive Approach for CBIR Using a Network of Radial Basis Functions," *IEEE Trans. Multimedia*, Vol. 6, No. 5, pp. 703-716, 2004.

[3] D. H. Widyantoro and J. Yen, "Relevant Data Expansion for Learning Concept Drift from Sparsely Labeled Data," *IEEE Trans. KDE*, Vol. 17, No. 3, pp. 401-412, 2005.

[4] S. Wilson, and G. Stefanou, "Bayesian Approaches to CBIR," *Proc. of Int. Conf. on Bayesian Statistics and its Applications*, 2005.

[5] K. Wu and K. H. Yap, "Fuzzy SVM for CBIR," *IEEE Computational Intelligence Magazine*, Vol. 1, pp. 10-16, 2006.

[6] K. Tieu and P. Viola, "Boosting Image Retrieval," *Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 228-235, 2000.

[7] M. Li, Z. Chen, and H. Zhang, "Statistical Correlation Analysis in Image Retrieval," *Pattern Recognition*, Vol. 35, pp. 2687-2693, 2002.

[8] 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.

[9] 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.

[10] 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.

[11] X. Qi and Y. Han, "Incorporating Multiple SVMs for Automatic Image Annotation," *Pattern Recognition*, Vol. 40, No. 2, pp. 728–741, 2007.

[12] X. Qi and R. Chang, "Image Retrieval Using Transaction-Based and SVM-Based Learning in Relevance Feedback Sessions," *Proc. of ICIAR*, LNCS 4633, pp. 637–649, 2007.