

A Clustering Based Approach to Efficient Image Retrieval

Ruofei Zhang, Zhongfei (Mark) Zhang

Department of Computer Science

*Thomas J. Watson School of Engineering and Applied Science
State University of New York at Binghamton, Binghamton, NY 13902*

E-mails: {rzhang, zhongfei}@cs.binghamton.edu

Abstract

This paper addresses the issue of effective and efficient content based image retrieval by presenting a novel indexing and retrieval methodology that integrates color, texture, and shape information for the indexing and retrieval, and applies these features in regions obtained through unsupervised segmentation, as opposed to applying them to the whole image domain. In order to address the typical color feature “inaccuracy” problem in the literature, fuzzy logic is applied to the traditional color histogram to solve for the problem to a certain degree. The similarity is defined through a balanced combination between global and regional similarity measures incorporating all the features. In order to further improve the retrieval efficiency, a secondary clustering technique is developed and employed to significantly save query processing time without compromising the retrieval precision. An implemented prototype system has demonstrated a promising retrieval performance for a test database containing 2000 general-purpose color images, as compared with its peer systems in the literature.

1. Introduction

Content-based image retrieval (CBIR) concerns automatic or semi-automatic retrieval of image data from an imagery database based on *semantic* similarity between the imagery content. The semantic similarity is typically defined through a set of imagery features. These features are extracted from shape, texture, or color properties defined in the imagery domain. The relevance between a query image and images in the database is ranked according to the similarity measure computed from the features. Due to its wide application potential, CBIR research has received intensive attention over the last few years.

In this paper we present a novel approach to addressing the general-purpose CBIR problem. This approach

integrates semantics-intensive clustering-based segmentation with fuzzy color histogram as well as texture and shape features to index imagery data, and consequently, is called FUZZYCLUB. A computationally efficient distance metric is proposed in FUZZYCLUB to reduce the query processing time. The response time is further improved by imposing a secondary clustering technique to achieve the high scalability in the case of very large image databases.

The paper is organized as follows. We begin with a brief review of the related work. Then we introduce to the unsupervised segmentation technique employed in FUZZYCLUB, which is followed by the development of all the indexing features defined in FUZZYCLUB, with a focus on the fuzzy color histogram. The distance metric and the overall similarity issues between two images are subsequently discussed, followed by the introduction to the secondary clustering technique in the region feature vector space to improve retrieval efficiency. Finally, the retrieval performance of FUZZYCLUB is evaluated with a comparison with its two peer systems in the literature, and the paper is concluded.

2. Related Work and Significance

A broad range of research efforts and commercial products [1,2,3] is reported to address the general-purpose CBIR problem. Almost all of the approaches proposed are based on indexing imagery in a feature space. Typical features are color, texture, shape, region, and appearance [4,6,7,9-16]. The most popularly used features are color histogram and its variants. They are used in systems such as IBM QBIC [7] and Berkeley Chabot [8]. Color histogram is computationally efficient, and generally insensitive to small changes in camera position. However, a color histogram provides only a very coarse characterization of an image, resulting a very coarse indexing; images with similar histograms may have dramatically different semantics. The “coarseness” of the color histogram approach is due to the total loss of spatial information of pixels in images. To retain the spatial

information of a color histogram, many research efforts are made in the literature. Pass and Zabih [4] described a split histogram called color coherence vector (CCV). Each of its buckets j contains pixels having a given color j and two classes based on the pixels spatial coherence. The feature is also extended by successive refinement, with buckets of a CCV further subdivided based on additional features. Huang et al [10] proposed color correlograms to integrate color and spatial information. Given n inter-pixel distances, a correlogram is defined as a set of n matrices $\gamma^{(k)}$, where an element $\gamma_{c_i, c_j}^{(k)}$ is the probability that a pixel with color c_i is at a distance k away from a pixel with color c_j . Rao et al [11] generalized the color spatial distribution by computing the color histogram with specific geometric relationships between pixels of each color histogram bucket. Cinque et al [12] proposed another color histogram refinement method, called Spatial-Chromatic Histogram, in which the average position of each color histogram and its standard deviation are recorded to add the spatial information into the traditional histogram. All of the refinement efforts failed to reflect the fuzzy nature of the color features inherently exhibited in the color histogram itself.

Ravela and Manmatha [13] proposed an appearance based image indexing technique using Gaussian derivative filters at several scales to compute low order 2D differential invariants as indexing features. Recently, region based features are developed to address the partial matching capability for robust CBIR. A region-based retrieval system segments images into regions (objects), and retrieves images based on the similarity between regions. Typical region based CBIR systems include Berkeley Blobworld[15], UCSB Netra[16], Columbia VisualSEEK[9], and Stanford IRM[6], of which [9,15,16] are the classic region based CBIR systems which require significant user interaction in defining or selecting region features, preventing from a friendly interface to users, especially to non-professional users. Another problem in the classic region based CBIR systems is that they focus too much on the region-based similarity as opposed to the similarity with a balanced focus between regions and global images. Wang et al [6] proposed an integrated regional matching scheme for CBIR, which allows for matching a region in one image against several regions from another image. As a result, the similarity between two images is defined as the weighed sum of the distances in a feature space between all regions from different images. Compared with the classic region-based CBIR systems, this scheme decreases the impact of inaccurate region segmentation by smoothing over the “inaccuracy” in distance. Nevertheless, the color representation of each region is simplistic such that much of the rich color information in a region is lost, as it fails to explicitly

express the “inaccuracy” of the color feature exhibited by the fuzzy nature in the feature extraction and human perception of color.

When we design FUZZYCLUB, we keep the following three principles in mind. First, we intend to apply pattern recognition techniques to connect low level features to high level semantics. Consequently, FUZZYCLUB is also region-based methodology, as opposed to indexing images in the whole image domain. Second, we intend to address the color “inaccuracy” issue typically existing in color based image retrieval in the literature. With this consideration, we apply fuzzy logic to the system. Third, we intend to improve the query processing time to avoid the typical linear search problem in the literature; this drives us to develop the secondary clustering technique currently employed in FUZZYCLUB. As a result, comparing with the existing techniques and systems, FUZZYCLUB exhibits the following distinctive advantages: (i) it solves for the color “inaccuracy” problem typically existing in color based CBIR systems to a certain degree (ii) it develops a balanced scheme in similarity measure between regional and global matching in order to capture as much semantic information as possible without sacrificing the efficiency (iii) it “pre-organizes” image databases to further improve retrieval efficiency without compromising retrieval effectiveness. The novelty of FUZZYCLUB is its improvement of the existing techniques and its incorporation and combination of these techniques together in a single system.

3. Image Segmentation

The very first step of FUZZYCLUB is to segment an image into different regions based on color and spatial variation features using a modified version of the k-means algorithm [17] due to its unsupervised learning nature such that we can adaptively updates the number of regions as an iterative process to accommodate the fact that the number of regions in an image is unknown before the segmentation. Image indexing is then taken based on the color, texture, and shape features in each region, as well as the global, overall combination of the regional features.

To segment an image into regions, FUZZYCLUB first partitions an image into 4 by 4 blocks to compromise between texture granularity and computation time. To apply the k-means algorithm, a feature vector consisting of six features from each block is defined as follows. Three of the features are the average color components in a 4 by 4 block. The LAB color space is used due to its desired property of the perceptual color difference proportional to the numerical difference in the LAB space. These features are denoted as $\{C_1, C_2, C_3\}$.

The other three features are used to capture the texture information of the image, represented by the energy in the

high frequency bands of the Haar wavelet transform [18], i.e., the square roots of the second order moments of wavelet coefficients in high frequency bands. To obtain these moments, a Haar wavelet transform is applied to the L component of the image. After a one-level wavelet transform, a 4 by 4 block is decomposed into four frequency bands; each band contains 2*2 coefficients. Without loss of generality, suppose the coefficients in the HL band are $\{C_{k,l}, C_{k,l+1}, C_{k+1,l}, C_{k+1,l+1}\}$. Then the texture feature of this block in the HL band is computed as:

$$f = \left(\frac{1}{4} \sum_{i=0}^1 \sum_{j=0}^1 c_{k+i,l+j}^2 \right)^{1/2} \quad (1)$$

The other two features are computed similarly in the LH and HH bands. These three features of the block are denoted as $\{T_1, T_2, T_3\}$. They can be used to discern texture by showing variations in different directions.

After obtaining the feature vectors for all the blocks, we perform the normalization on both color and texture features to whiten them such that the effects of different feature ranges are eliminated. Consequently the k-means algorithm [17] is used to cluster the feature vectors into several classes with every class in the feature space corresponding to one spatial region in the image space. Since clustering is performed in the feature space, blocks in each cluster do not necessarily form a connected region in the image. This way, we preserve the natural clustering of objects in general-purpose images. The k-means algorithm does not specify how many clusters to choose. We adaptively select the number of clusters C by gradually increasing C until a stop constraint is satisfied. The average number of clusters for all images in the database varies in accordance with the adjustment of the stop constraint. In the k-means algorithm we use a color-texture weighted L2 distance metric as

$$\sqrt{w_c \sum_{i=1}^3 (c_{1i} - c_{2i})^2 + w_t \sum_{i=4}^6 (t_{1i} - t_{2i})^2} \quad (2)$$

to describe the distance between blocks. In our implemented prototype system of FUZZYCLUB, we set $w_c = 0.65$ and $w_t = 0.35$.

After the segmentation, FUZZYCLUB is ready for image indexing. In other words, image indexing is based on the features defined in the regions obtained from the image segmentation.

4. Region-Based Features

Within each region, we define three types of features: color, texture, and shape, along with the conventional

geometric information as the feature vector for image indexing.

4.1 Color Features

Color is the most popularly used type of features in image indexing. On the other hand, due to its inherent nature of ‘‘inaccuracy’’ in description of the same semantic content by different the color quantization and/or by the uncertainty of human perception, it is important to capture this ‘‘inaccuracy’’ when define the features. We apply fuzzy logic [22][5] to the traditional color histogram to help capture this uncertainty in color indexing.

We assume that any color is a fuzzy set [20]. That means we will associate any color to a fuzzy function, $\mu_c : \mu \rightarrow [0,1]$ and for any color c' of the color universe, $\mu_c(c')$ is the resemblance degree of the color c' to the color c . The fuzzy model we define should follow the property that the resemblance degree decreases as the inter-color distance increases. The natural choice, according to the typical soft computing literature [20], is to impose a smooth decay of the resemblance function when the inter-color distance increases. Since the LAB color space is of the equivalence between the perceptual inter-color distance and the actual Euclidean distance between the color space coordinates, we define a Gaussian operator to be the fuzzy logic function:

$$\mu_c(c') = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{d^2(c, c')}{2\sigma^2}\right\} \quad (3)$$

where d is the Euclidean distance between color c and c' in LAB space, and σ is the average distance between color c and c' defined as

$$\sigma = \frac{2}{B(B-1)} \sum_{i=1}^{B-1} \sum_{k=i+1}^B d(c, c') \quad (4)$$

where B is the number of buckets in the color histogram.

This fuzzy color model enables to enlarge the influence of a given color to its neighboring colors, according to the uncertainty principle and the perceptual similarity. This means that each time a color c is found in the image, it will influence all the quantized colors according to their resemblance to the color c . Numerically, this is expressed as:

$$h(c) = \sum_{c' \in \mu} h(c') \mu_c(c') \quad (5)$$

where μ is the color universe in the image and $h(c')$ is the normalized, traditional color histogram. This fuzzy histogram operation in fact is the linear convolution between the traditional color histogram and the fuzzy color model. This convolution expresses the histogram smoothing, provided that the color model is indeed a

smoothing, low-pass filtering kernel. The use of Gaussian function as the color model helps generate such a smooth histogram, which subsequently helps reduce the quantization errors [21].

In the implementation of the prototype system of FUZZYCLUB, the LAB color space is quantized into 96 buckets by using uniform quantization (L by 6, A by 4, B by 4). Then Eq. 5 is applied to obtain the fuzzy histogram for each region. $\mu_c(c')$ for each bucket in the histogram is pre-computed based on Eqs. 3 and 4, and is implemented as a lookup table to reduce the online computation.

4.2 Texture Features

The texture features are defined as the centroid vector of all the three-components Haar wavelet moment vectors defined in Section 3 for each blocks of a region.

4.3 Shape Features

The shape features are defined as a vector containing three components for the normalized inertia [19] of order 1 to 3 of a region, respectively. For a region H in 2-dimensional Euclidean space \mathfrak{R}^2 (i.e., an image space), the normalized inertia of order p is

$$l(H, p) = \frac{\sum_{(x,y):(x,y) \in H} [(x - \hat{x})^2 + (y - \hat{y})^2]^{p/2}}{[V(H)]^{1+p/2}} \quad (6)$$

where $V(H)$ is the number of pixels in the region H , and (\hat{x}, \hat{y}) is the centroid of H . The minimum normalized inertia is achieved by spheres. Denote the p^{th} order normalized inertia of spheres as L_p . The following features are used to describe the shape of a region:

$$S_1 = l(H, 1) / L_1, S_2 = l(H, 2) / L_2, S_3 = l(H, 3) / L_3 \quad (7)$$

Now the indexing vector of a region consists of the three fuzzy color features, the three texture feature components, and the three shape feature components defined in Eq. 7. In addition, the location of the region, represented as the centroid coordinates of the region, and the area of the region, represented as the total number of pixels, are also captured as part of the indexing feature vector, resulting in a complete feature vector for each region.

5. Region Matching and Image Similarity

To compute the distance between two regions, we apply L2 distance metric to fuzzy color histogram, texture

vector, and shape vector, respectively. For the fuzzy histogram, we define the distance metric as:

$$d_C^{pq} = \sqrt{\sum_{i=0}^{B-1} [(h_p(i) - h_q(i))]^2} \quad (8)$$

where B is the number of buckets in the histogram, and $h_p(i)$ and $h_q(i)$ are the fuzzy histograms for region p and q , respectively. Similarly, L2 distance metric is applied to the texture vector and the shape vector, respectively:

$$d_T^{pq} = \|T_p - T_q\| \quad (9)$$

$$d_S^{pq} = \|S_p - S_q\| \quad (10)$$

where T_p, S_p are the texture feature vector and the shape feature vector for region p , respectively, and T_q, S_q are for region q , respectively.

To measure the distance between two regions, we separate the contribution from the color and texture features from that from the shape features, as the former is considered more reliable in image indexing than the latter. Consequently, given two regions p and q , the inter-region distance on color and texture is defined as

$$D_{pq} = \sqrt{d_C^{pq^2} + d_T^{pq^2}} \quad (11)$$

The intra-region distance (i.e., the deviation) for region p on color and texture is defined as

$$D_{pp} = \left[\frac{1}{N_p} \sum_{j=1}^{N_p} \|Z_p^j - Z_p\|^2 \right]^{1/2} \quad (12)$$

where N_p is the number of blocks in region p , Z_p^j is the color-texture vector of block j in region p defined in Section 3, and Z_p is the centroid vector in region p of all the color-texture vectors for all the blocks in this region.

Conceptually, the overall distance between two regions should increase when the inter-region distance increases and when the intra-region distance decreases. Hence, we define the overall distance between two regions p and q as follows:

$$DIST_{pq} = w \frac{D_{pq}}{D_{pp} + D_{qq}} + (1-w) d_S^{pq} \quad (13)$$

where w is a weight. In the prototype system of FUZZYCLUB, we set w as 0.7. Since all components are normalized, this overall distance between two regions is also normalized. The separation between the contribution from the color and texture features and that from the shape features allows us to adjust the weights for these different contributions. The favor to the former in weight in the prototype system reflects our belief that the former is more

reliable than the latter, which is verified by the experimental results.

Given the definition of the distance between two regions, we are ready to compute the global similarity between two images. Suppose we have M regions in image 1 and N regions in image 2, the following algorithm computes the global similarity between image 1 and image 2:

Step 1: compute the distance between one region in image 1 and all regions in image 2. For each region i in image 1, the distance between this region and image 2 is defined as:

$$R_{i\text{Image}2} = \text{Min}(\text{DIST}_{ij}, j) \quad (14)$$

where j enumerates each regions in image 2. This definition captures the minimum distance between a region and all the regions in an image, which maximizes the potential similarity between the region and the image.

Step 2: similarly, the distance between a region j in image 2 and image 1 is defined as

$$R_{j\text{Image}1} = \text{Min}(\text{DIST}_{ji}, i) \quad (15)$$

where i enumerates each regions in image 1.

Step 3: now we have $M+N$ distances. We define the distance between two images (1 and 2) as follows:

$$\text{Dist Image}(1,2) = \frac{\sum_{i=1}^M w_{1i} R_{i\text{Image}2} + \sum_{j=1}^N w_{2j} R_{j\text{Image}1}}{2} \quad (16)$$

where w_{1i} is the weight for region i in image 1, and w_{2j} is the weight for region j in image 2. Since we think a region with a larger area plays a more significant role in contributing to the overall similarity value between two images than a region with a smaller area, we define

$$w_{1i} = \frac{N_{1i}}{N_1}, \text{ where } N_{1i} \text{ is the number of blocks in region } i \text{ and } N_1 \text{ is the total number of blocks in image 1, and define } w_{2j} \text{ similarly for image 2.}$$

This definition of the overall similarity between two images captured by the overall distance between the images is a balanced scheme in similarity measure between regional and global matching. As compared with many existing similarity measures in the literature, this definition strives to incorporate as much semantic information as possible and at the same time also achieves a computational efficiency. Given this definition, for each query image q , it is straightforward to compute $\text{Dist Image}(q, d)$ for every image d in the database in the retrieval.

6. Secondary Clustering and Image Retrieval

The time of image retrieval depends in a large degree on the number of images in the database in almost all CBIR systems. Many existing systems attempt to compare the query image with every target images in the database to find the top matching images, resulting in an essentially linear search, which is prohibitive when the database is large. We believe that it is not necessary to conduct a whole database comparison. In fact, it is possible to exploit a priori information regarding the ‘‘organization’’ of the images in the database in the feature space before a query is posed, such that when a query is received, only a part of the database needs to be searched while a large portion of the database may be eliminated in the search. This certainly saves significant query processing time without compromising the retrieval precision.

To achieve this goal, in FUZZYCLUB we add a pre-retrieval screening phase to the feature space after a database is indexed by applying a secondary k-means algorithm to the distance DIST_{pq} in the region feature vector space to cluster all the regions in the database into classes. The philosophy is that regions with similar {color, texture, shape} features should be grouped together in the same class. This secondary clustering is performed offline, and each region’s indexing data along with its associated class ID is recorded in the index files. Consequently, in the prototype implementation of FUZZYCLUB, the image database is indexed in terms of a three level tree structure, one for the region level, one for the class level, and one for the image level.

Assuming an image database is indexed based on the features defined in Sections 4 and 5, and is ‘‘organized’’ based on the secondary clustering, given a query image, FUZZYCLUB processes the query as follows:

Step 1: Perform the query image segmentation to obtain all the regions, say we have regions $Q_i, i \in [0, N-1]$ in the query image.

Step 2: Compute the distances between each region Q_i and all class centroids in the database to determine which class Q_i belongs to by the minimum-distance-win principle. Assume that region Q_i belongs to class $C_j, j \in [0, K-1]$.

Step 3: Retrieve all the regions in the database which belongs to the class $C_j, j \in [0, K-1]$. These regions comprise a region set T_{jd} . The images containing any regions in the set T_{jd} is subsequently retrieved from the index structure. These images comprise an image set I_d

Step 4: Compare the query image with the images in the image set I_d . The distance $Dist Image$ is used for each pair and the top-least-distance images are returned in the retrieval.

It is shown theoretically and experimentally that FUZZYCLUB saves query processing time significantly without compromising the retrieval precision by avoiding searching the whole database.

7. Experimental Evaluations

We have implemented FUZZYCLUB as a prototype system in a platform of Pentium III 800 MHZ CPU with 256M memory, and have evaluated this prototype system in a general-purpose image database containing 2000 images from COREL. These images are manually divided into 10 well-defined groups such as people, nature scenery, building, and vehicles. No pre-restrictions such as different camera models or lighting conditions are specified in the image database for this evaluation. These images are all in JPEG format

The whole indexing time, including running the secondary clustering after indexing each image, for the 2000 image database takes 60-70 minutes, corresponding to about 2 seconds per image. The average query time for returning top 20 images per query is less than 1 second.

To evaluate the retrieval effectiveness of FUZZYCLUB, we randomly select 4 query images with different semantics, namely flower, dinosaur, vehicle, and African people. For each query, we examine the precision of the retrieval based on the relevance of the image semantics. The semantic relevance is determined by manual truthing the query image and each of the retrieved images in the retrieval. The four query retrievals are shown in Figure 1.



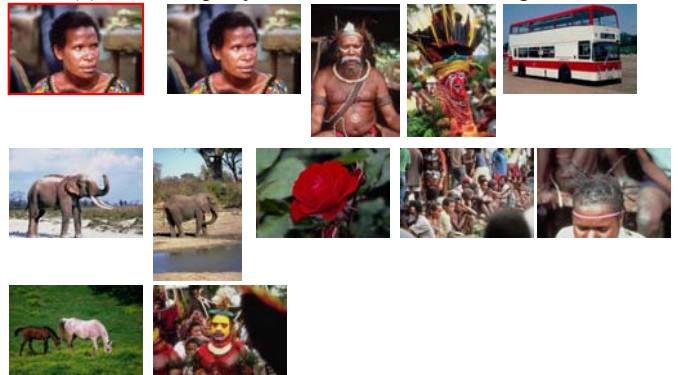
(a) Flower query; 10 matches out of the top 11.



(b) Dinosaur query; 11 matches out of the top 11.



(c) Vehicle query; 8 matches out of the top 11.



(d) African people query; 6 matches out of the top 11.

Figure 1. Retrieval results of the four queries evaluated; the top-left corner image is the query, and the ranking goes rightward and downward.

To perform a more quantitative evaluation, we compare the performance of FUZZYCLUB with those of IRM [6] and the Geometric Histogram [11]. For each group images in the 2000 database images, we randomly select 30 images as queries to each of the three systems, respectively. The average precisions for each groups based on the returned top 21 images are recorded. Since the number of relevant images in the database for each query image is the same, the recall values are not computed as they are proportional to the precision values in this case. The precision comparison among these three systems is shown in Figure 2. Figure 3 shows the average precision-recalls of FUZZYCLUB taken over 100 queries randomly selected over all the ten groups.

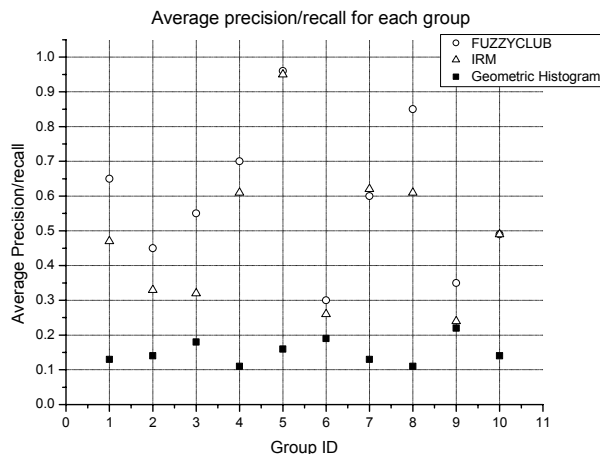


Figure 2. Average precision/recall comparison between FUZZYCLUB, IRM, and Geometric Histogram systems.

These evaluation results clearly show that the performance of FUZZYCLUB is significantly better than those of Geometric Histogram [11] because Geometric Histogram approach only focuses on the color features, which is not explicitly semantic-related for many images. And even for the color features it does not capture the inherent “inaccuracy” of color features in semantic content representation. The evaluation results also show that FUZZYCLUB noticeably outperforms IRM [6] system in most groups. This is in part due to the more detailed color representation and application of fuzzy logic in the feature matching in FUZZYCLUB.

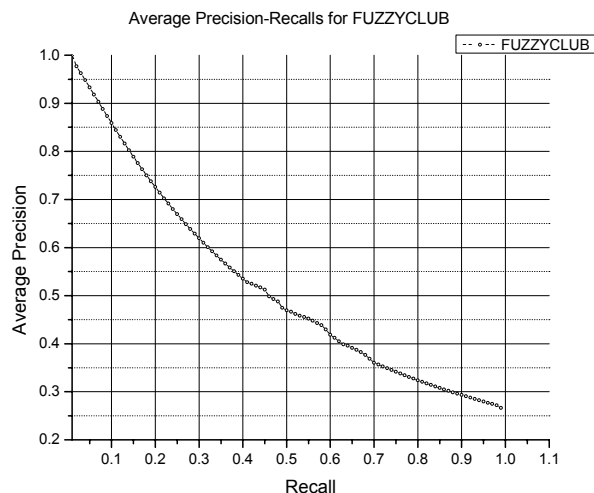


Figure 3. Average precision-recall of FUZZYCLUB over randomly selected 100 queries.

In order to evaluate the query processing speed, we compare the query processing time of FUZZYCLUB with that of IRM [6]. Figure 4 shows the average response time of the two systems to return top 21 images for 30 queries in each of the 10 groups.

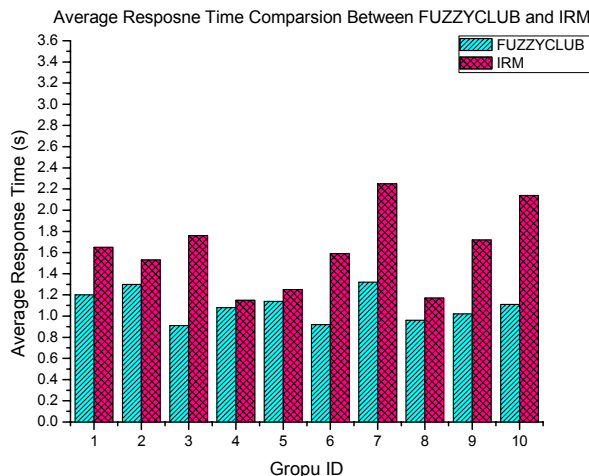


Figure 4. Comparison of the query processing time between FUZZYCLUB and IRM.

From Figure 4, the average response time reduction of FUZZYCLUB to IRM [6] is 29.56%. With the size of the database increases, the retrieval efficiency of FUZZYCLUB increases proportionally. This is due to the secondary clustering technique employed in FUZZYCLUB.

8. Conclusion

A novel image indexing and retrieval methodology, called FUZZYCLUB, is presented in this paper. FUZZYCLUB integrates color, texture, and shape information along with the conventional geometric information together as an indexing vector, and applies the indexing vector to regions as opposed to a whole image. Consequently, the overall image similarity is developed through regional similarity based on all the feature components. In order to address the color feature ‘inaccuracy’ problem, FUZZYCLUB applies fuzzy logic to color histograms. In order to improve the query processing time, FUZZYCLUB incorporates a secondary clustering technique to “pre-organize” the database to significantly save the search time. Experimental evaluation based on a 2000 COREL color image database shows that FUZZYCLUB promisingly outperforms the peer image retrieval systems from the literature. As compared with the existing techniques and systems, FUZZYCLUB exhibits the following distinctive advantages: (i) it solves for the color “inaccuracy” problem typically existing in color

based CBIR systems to a certain degree (ii) it develops a balanced scheme in similarity measure between regional and global matching in order to capture as much semantic information as possible without sacrificing the efficiency (iii) it “pre-organizes” image databases to further improve retrieval efficiency without compromising retrieval effectiveness. The novelty of FUZZYCLUB is its improvement of the existing techniques and its incorporation and combination of these techniques together in a single system.

Reference

- [1] M. De Marsicoi, L. Cinque, and S. Levialdi, “Indexing pictorial documents by their content: a survey of current techniques”, *Image and Vision Computing*, 1997, 15 pp. 119-141.
- [2] Arnlod W. M. Smeulders, Marcel Worring et al, “Content-Based Image Retrieval at the End of the Early Years”, *IEEE Trans. On PAMI*, Vol. 22, No. 12, Dec. 2000.
- [3] Y. Alp Aslandogan, and Clement T. Yu, “Techniques and Systems for Image and Video Retrieval”, *IEEE Trans. on Knowledge and Data Engineering*, Vol. 11, No.1, Jan. 1999.
- [4] Greg Pass, and Ramin Zabih, “Histogram Refinement for Content-Based Image”, *Proceedings of the IEEE Workshop on Applications of Computer Vision*, Sarasota, Florida, Dec. 1996.
- [5] Constantin Vertan, and Nozha Boujemaa, “Embedding Fuzzy Logic in Content Based Image Retrieval”, *Proceedings of the 19th International Meeting of the North America Fuzzy Information Processing Society*, Atlanta, Jul. 2000.
- [6] James Z. Wang, Jia Li and Gio Wiederhold, “SIMPLiCity: Semantics-Sensitive Integrated Matching for Picture Libraries”, *IEEE Trans. On PAMI*, Vol. 23, No.9, Sep. 2001.
- [7] M. Flickner et al, “Query by Image and Video content: The QBIC system”, *IEEE Computer*, Vol. 28, No. 9, Sep. 1995, pp. 23-32.
- [8] Virginia Ogle, and Michael Stonebraker, “Chabot: Retrieval from a relational database of images”, *IEEE Computer*, Vol. 28, No. 9, Sep. 1995, pp. 40-48.
- [9] John R. Smith, and Shih-Fu Chang, “VisualSEEk: a fully automated content-based image query system”, *ACM Multimedia 96*, No. 11, Nov. 1996.
- [10] Jing Huang, S Ravi Kumar et al, “Image Indexing Using Color Correlograms”, *Proceedings of the IEEE Conf. Computer Vision and Pattern Recognition*, Puerto Rico, Jun. 1997.
- [11] Aibing Rao, Rohini K. Srihari, and Zhongfei Zhang, “Geometric Histogram: A Distribution of Geometric Configuration of Color Subsets”, *Internet Imaging, Proceedings of SPIE*, Vol. 3964, San Jose, Jan. 2000, pp. 91-101.
- [12] L. Cinque, G. Ciocca et al, “Color-based Image Retrieval using Spatial-chromatic histograms”, *Image and Vision Computing*, Vol. 19, 2001, pp. 979-986.
- [13] S. Ravela, and R. Manmatha, “Retrieving Image by Appearance”, *Proceedings of the International Conf. on Computer Vision (ICCV)*, Bombay, India, Jan. 1998.
- [14] H. Wang, F. Guo, and D.D. Feng, “A Signature for Content-based Image Retrieval Using Geometrical Transform”, *ACM Multimedia 98*, 1998
- [15] C. Carson, M. Thomas, S. Belongie, et al, “Blobworld: A System for Region-Based Image Indexing and Retrieval”, *Proceedings of the third international conf. on Visual Information System*, Amsterdam, Netherlands, Jun. 1999, pp. 509-516.
- [16] W.Y. Ma, and B. Manjunath, “NaTra: A Toolbox for Navigating Large Image Databases”, *Proceedings of the IEEE International Conf. on Image Processing*, Santa Barbara, Oct. 1997, pp. 568-571.
- [17] J.A. Hartigan, and M.A. Wong, “Algorithm AS136: A k-means Clustering Algorithm”, *Applied Statistics*, Vol. 28, 1979, pp. 100-108.
- [18] I. Daubechies, *Ten Lectures on Wavelets*, Capital City Press, 1992.
- [19] A.Gersho, “Asymptotically Optimum Block Quantization”, *IEEE Trans. On Information Theory*, Vol. 25, No. 4, Apr. 1979, pp. 373-380.
- [20] Zheru Chi, Hong Yan and Tuan Pham, *Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition*, World Scientific, 1996.
- [21] J.Kautsky, N.K.Nichols, and D.L.B.Jupp, “Smoothed histogram modification for image processing”, *CVGIP: Image Understanding*, Vol. 26, No.3, Jun. 1984, pp. 271-291.
- [22] Sankar K. Pal, Ashish Ghosh, and Malay K. Kundu, *Soft Computing for Image Processing*, Physica-Verlag, 2000.