An Efficient Multi-filter Retrieval Framework For Large Image Databases

Xiuqi Li[1], *Shu-Ching Chen*[2*], *Mei-Ling Shyu*[3], *Borko Furht*[1]
[1] NSF/FAU Multimedia Laboratory, Florida Atlantic University, Boca Raton, FL 33431
[2]Distributed Multimedia Information System Laboratory
School of Computer Science, Florida International University, Miami, FL 33199
[3]Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL 33124

Keywords: Content-Based Image Retrieval, Multimedia Systems, Multimedia Databases

Abstract

An efficient multi-filter retrieval framework for image retrieval in large image databases is proposed. Multiple filters are used to reduce the search ranges at different stages and thus save the time spent on unnecessary similarity comparison. First, a color label histogram filter uses a color label histogram with only thirteen bins to eliminate those images in the image database that are dissimilar to a query image in colors. Next, a wavelet texture filter discards the images that are dissimilar to the query image in texture from the query results of the color filter. A texture distance measure that considers the relationship between the coefficient value ranges and the decomposition levels is proposed. Finally, a spatial segmentation filter removes images dissimilar to the query image in spatial information from the query results of the texture filter. A unique unsupervised segmentation algorithm together with the wavelet technique produces the spatial features of an image automatically. All images passing the three filters are ranked based on the total normalized distance in color, texture, and spatial information. The top N images are displayed in the user interface. The experimental results demonstrate that the proposed framework dramatically reduces the search range.

INTRODUCTION

More and more large-scale image collections become available due to the recent advances in hardware and image compression techniques. As a result of this, the traditional way of retrieving images by manually annotated keywords (text-based) became more and more problematic in large-scale image databases. This is because of two main reasons. First, a large amount of human labor effort has to be devoted to manually annotate all images in a large-scale image database before the retrieval, which is time-consuming and expensive. The second reason is perception subjectivity, which means that the rich meaning of an image is difficult to be precisely described and different people may describe the same image in different ways [1][2]. To overcome the drawbacks of the text-based approach, the *content-based image retrieval* (CBIR) approach [1][2] that retrieves images directly and automatically based on their visual contents such as color, texture, and shape has emerged. In a typical CBIR system, the query pattern is *query by example*, which searches the top N images similar to an example image. In general, the visual features are first extracted offline from all images in an image database before the retrieval. During the retrieval, the visual features of the example image are compared to those of all images in the image database and the top N images are displayed in the user interface [1][2].

Many CBIR systems have been built for commercial or research purposes [1]. In the QBIC system [3], images are queried using color, texture, or shape information. Two color vectors are used. One is the 3D average color vector of an image in RGB, YIQ, Lab, and Munsell color spaces. The other one is a 256-bin color histogram in RGB color space. The average color vector acts as a filter to limit the expensive computations required in the color histogram computation. An improved Tamura texture representation is used as texture information. Shape information consists of area, circularity, eccentricity, major axis orientation, and moment invariants [1]. The VisualSEEK system [4] adopts the following visual features: colors represented by a color set, texture described by the energies of wavelet coefficients in all subbands, and spatial relationships between image regions. During the retrieval, queries based on individual features are first performed independently. Then the results are combined using a weighted sum of the distances [1][4]. The system in [5] employs an icon filter, a graph-photo detector, a color histogram filter, a texture filter and a shape filter to screen the objectionable images.

Our framework is unique in several aspects. First, the color label histogram with only thirteen bins is adopted to extract global color information effectively and efficiently. Secondly, three texture feature vectors and a novel distance measure based on the wavelet decomposition technique are proposed. The distance measure takes into account the relationship between the coefficient value scale and the

[*] This research was supported in part by NSF CDA-9711582.

decomposition level. Thirdly, the spatial information of an image is automatically extracted using a special unsupervised segmentation algorithm. Fourthly, the wavelet technique is utilized to generate the initial partition for the segmentation algorithm, which improves the segmentation performance twice.

The rest of the paper is organized as follows. The multi-filter retrieval framework is presented in Section 2. This section first gives an overview of the entire framework, and then discusses the color filter, the wavelet texture filter, the spatial filter, and the final query presentation in order. Section 3 shows the experimental results. The conclusion and future work are given in Section 4.

THE MULTI-FILTER IMAGE RETRIEVAL FRAMEWORK

An overview of our framework is shown in Figure 1. It consists of six steps. First, the color label histogram computation algorithm, wavelet texture computation algorithm and unsupervised segmentation algorithm are applied to all images in the image database to extract the visual content information off-line. These three algorithms capture color, texture and spatial information, respectively. The visual feature information of the images is stored in another database (called the image feature DB) for later comparison. The second step is to apply the three algorithms to the query image to extract the query image features. Then the color label histogram filter compares the query image to all images in the image database by the color information and filters out those images in the image database whose color information is much different from that of the query image. Next, the wavelet texture filter takes the query result from the color filter, compares the images by the texture information and removes the images whose color information is similar to that of the query image but the texture information is much dissimilar to that of the query image. The query result in this stage is fed to the spatial segmentation filter. This third filter compares the query image to the images in its search range by the spatial information and eliminates the images that are similar to the query image in color and texture but much different from the query image in spatial information. At the last stage, all images passing three filters are ranked based on the total normalized distance in color, texture, and spatial information.

Color Label Histogram Filter

Color is the most dominant and distinguishing visual feature in content-based image retrieval [2]. A color space must be chosen before extracting the color information. Among all color spaces, only H (Hue) S (Saturation) V (Value) and its variants are perceptual, which makes HSV a proven color space particularly amenable to color image analysis. Therefore, we chose the HSV color space [6][12][13]. There exist many schemes to describe the color information in an image. Color histogram is chosen in our framework because it is most commonly used and computationally fast, and performs well in identifying the global color information [2] [12] [13].

To quantize the color space, we categorize colors based on their H, S and V value ranges. We adopted the categorization method in [6] to color histogram computation. We disregard the difference between the bright chromatic pixels and the chromatic pixels for reducing the total number of bins. Each transition color slice is treated as a separate category. A new category "gray" is added to count all possible value ranges. Therefore, totally thirteen color categories are used in our framework [12][13].

The color filter compares the color label histogram of the query image to that of all images in the database using the L_1-Distance [2]:

$$D_{color}(q,d) = \sum_{k=1}^{B} \left| CH_k^{(q)} - CH_k^{(d)} \right|,$$

where CH_k is the kth bin, B is the total number of bins, and q and d denote the query image and the dth image in the database.

The color filter effectively removes around eighty-five percent of the images in the database from the search range by a color distance threshold value. This greatly saves the time that would otherwise be taken for feature comparison in later stages.

Wavelet Texture Filter

In this subsection, we first discuss how the texture information from an image is extracted using wavelet transform and then how the wavelet texture filter removes images from its search range.

Texture Extraction Using Wavelet Transform

A wavelet is a function that has the shape of a waveform, limited duration, and average value of zero. Wavelet transform converts a function to a combination of two families of basis functions called wavelets, denoted by $\psi_{mn}(x)$ and $\varphi_{mn}(x)$. $\psi_{mn}(x)$ and $\varphi_{mn}(x)$ are generated through translation and dilation of a mother wavelet $\psi_{mn}(x)$ and a father wavelet $\varphi_{mn}(x)$ respectively, as shown in the following formulas [8] [9] [10]:

$$\psi_{mn}(x) = 2^{-m/2}\psi(2^{-m}x - n),$$

$$\varphi_{mn}(x) = 2^{-m/2}\varphi(2^{-m}x - n),$$

where m and n are integers, m refers to the scale, and n refers to the position. A signal f(x) can be represented by $f(x) = \sum_m \sum_n \left(a_{mn}\varphi_{mn}(x) + c_{mn}\psi_{mn}(x)\right)$, where a_{mn} and c_{mn} are the coefficients after wavelet transformation. In a one-level wavelet transformation, a signal is passed through a low-pass filter (LPF) and a high-pass filter (HPF) and the outputs of the filers are downsampled by two.

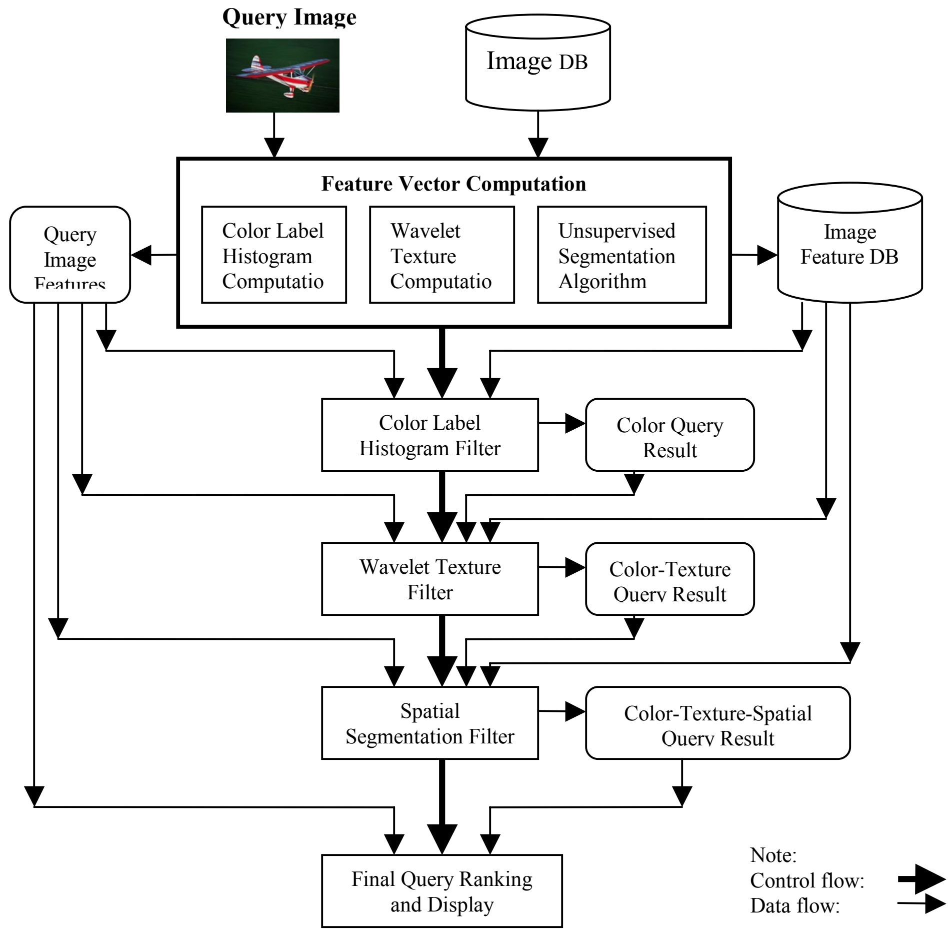

Figure 1. The architecture of the multi-filter retrieval framework.

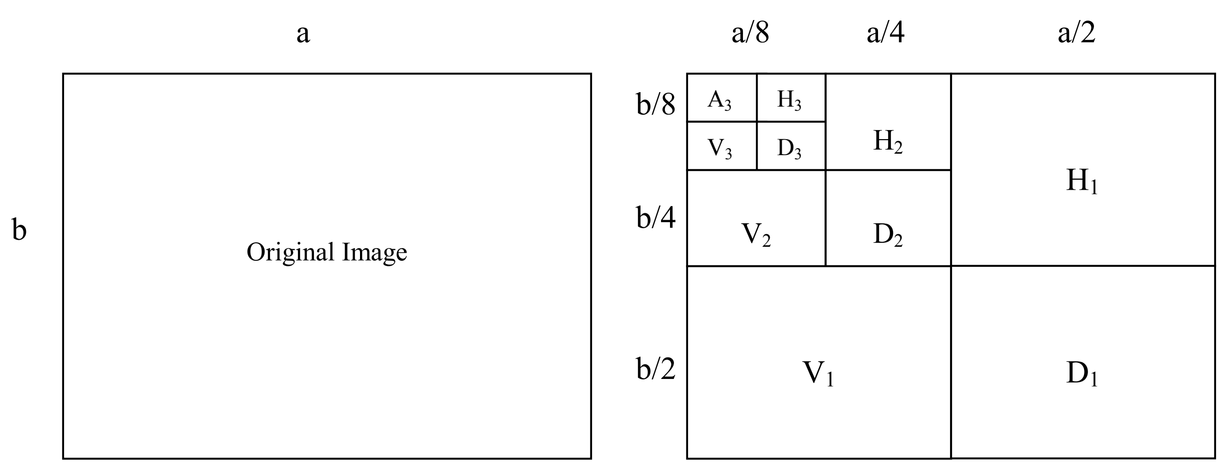

Figure 2. A three-level wavelet decomposition of an image.

The LPF generates coefficients a_{mn}, while the HPF produces coefficients c_{mn}. a_{mn} and c_{mn} correspond to the trend and fluctuation at a particular scale m and a particular position n, respectively. By recursively applying the above one-level wavelet transform to a signal and its sub-signals with lowest frequency at each level, we can extract approximation and detail information from the signal at different scales. Therefore, wavelet transform provides a multi-resolution analysis of a signal. To reconstruct the original signal, the wavelet coefficients are recursively upsampled and passed through a different set of low-pass and high-pass filters [8] [9][10].

For an image, the first-level wavelet transform is done by first applying one-level wavelet transform to each row the original image and then applying one-level transform to each new column of the intermediary result matrix. This generates four sub-images (subbands) of the original image, which are the approximation sub-image A_1, the horizontal detail sub-image H_1, the vertical detail sub-image V_1, and the diagonal detail sub-image D_1. Applying the same procedure to the sub-image A_1 generates the second level wavelet transform, which consists of four sub-images of A_1: A_2, H_2, V_2, and D_2. Figure 2 shows a three-level wavelet transform of an image. A_3 is the thumbnail sub-image of the original image.

There are many kinds of wavelets such as Haar, Daubechies, and Coif. Daubechies wavelets are proven to be good for image analysis and synthesis because of their compact support, more continuous derivatives, and zero integral of mother wavelets [10]. Therefore, Daubechies wavelets are chosen in our framework. The maximum decomposition level is three.

The wavelet texture computation algorithm decomposes the gray-scale versions of a query image and all images in the image database and stores three feature vectors for each image. The three vectors are mean vector, variance vector, and energy vector representing the mean, variance, and energy of each sub-image (subband), respectively.

Wavelet Texture Filter

The wavelet texture filter compares the mean, variance, energy vectors of a query image to those of an image in the query result obtained from the color filter using the following scheme. First, for each feature vector, a Euclidean distance L_2 between the corresponding components at each decomposition level is computed. This generates three distance values for each feature vector, totally nine distance values. Given a query image *q* and the *ith* image in the search range, the distance between the component c_j in two vectors of texture feature *k* (mean, variance, or energy) at level *l* is $D(q,i)^{(k,l)} = \sqrt{\sum_j \left(c_j^{(k,l)}(q) - c_j^{(k,l)}(i)\right)^2}$, where *j* refers to the subband *j* at level *l*.

Secondly, all nine distance values are normalized by dividing them by their maximum distance values. The normalization is necessary because the coefficients at higher decomposition level are usually much larger than those at the lower level. A total distance is computed for each feature (mean, variance, or energy) using the mean of the three normalized distance values at three levels. Given a query image *q* and the *ith* image in the search range, the distance between the two vectors of their texture feature *k* (mean, variance, or energy) is $D(q,i)^{(k)} = \frac{1}{3}\sum_{l=1}^{3}\left(D(q,i)^{(k,l)} / \left(\underset{i}{Max}\, D(q,i)^{(k,l)}\right)\right)$, where *l* refers to the decomposition level *l*.

Thirdly, a total texture distance between a query image q and the ith image in the search range is computed using the mean of the three texture feature distances as shown below:

$$D_{texture}(q,i) = \frac{1}{3}\sum_{k=1}^{3} D(q,i)^{(k)},$$

where *k* represents the *kth* texture feature.

The wavelet texture filter removes images that are dissimilar to a query image in texture by means of thresholding the total texture distance. Around half of the images in the query result obtained from the color filter are eliminated. All images passing the wavelet texture filter are sent to the spatial segmentation filter for further comparison.

Spatial Segmentation Filter

We use the class parameters generated by our Simultaneous Partition and Class Parameter Estimation (SPCPE) algorithm to represent the spatial information because the algorithm assumes that the intensity of a pixel is a polynomial function of its spatial location with its class parameters being the coefficients of the function. The SPCPE algorithm partitions a gray-scale image into b segments that belong to s classes. With an initial partition, the estimation of the partition and class parameters is done iteratively and simultaneously using the Bayesian learning approach. [7][11] [12] [13].

The initial partition is very important in generating a good segmentation result. The previous version of the SPCPE algorithm generates the initial partition randomly, which is unstable and not very effective. To correct this deficiency, the wavelet decomposition technique is utilized. The idea is to generate a candidate partition for each decomposition subband based on the salient points in that subband. The one with the least cost $J(C_1, C_2, \theta_1, \theta_2)$ is selected as the final result.

$$J(C_1, C_2, \theta_1, \theta_2) = \sum_{y_{ij} \in C_1} -\ln p_1(y_{ij} \mid \theta_1) + \sum_{y_{ij} \in C_2} -\ln p_2(y_{ij} \mid \theta_2),$$

where y_{ij} refers to the intensity at pixel *(i,j)*, $C_1, C_2, \theta_1, \theta_2$ represent the partitions and class parameters for class 1 and 2, and $p_1(y_{ij} \mid \theta_1)$ and $p_2(y_{ij} \mid \theta_2)$ are the probabilities of the pixel *(i,j)* belonging to class 1 and class 2 given class parameters θ_1, θ_2 respectively [12] [13].

The SPCPE algorithm segments a query image and all images in the image database and stores two vectors (one for each class) for each image. Each vector consists of all parameters for a class. The spatial segmentation filter computes the spatial information similarity between a query image q and an image i in the search range using the following Euclidian distance between their corresponding class parameter vectors:

$$D_{spatial}(q,i) = \sum_{n=1}^{2} \sqrt{\sum_{j=0}^{3} \left(a_{nj}^{(q)} - a_{nj}^{(i)}\right)^2},$$

where n refers to the class n and a_{nj} refers to the *jth* class parameter for class n. A small threshold is used by the spatial segmentation filter to eliminate the images that are not similar to the query image in the spatial information. It effectively eliminates approximately half of the images in its search range.

Final Query Result Presentation

At the last stage, a total distance taking into account the color, texture, and spatial information should be used to rank the images in the query result obtained from the spatial filter based on their similarities to the query image. Normalization is necessary before the total distance is computed because the distance values in different visual features may be at different scales. We use the ratio of each visual feature distance to its corresponding maximum distance. The total similarity distance between a query image q and the image i in the final query result is given as:

$$D(q,i) = D_{color}(q,i) / \max_{i \in [1,K]} (D_{color}(q,i))$$
$$+ D_{texture}(q,i) / \max_{i \in [1,K]} (D_{texture}(q,i))$$
$$+ D_{spatial}(q,i) / \max_{i \in [1,K]} (D_{spatial}(q,i)),$$

where K is the total number of images in the image database.

The images in the query result of the spatial filter are sorted based on their total normalized distances. The top six or less images similar to the query image are displayed in the query user interface as the final query result.

EXPERIMENTAL RESULTS

The image database in our current system contains 2,000 images in Corel Gallery downloaded from [14]. There are twenty semantic categories, each of which consists of 100 images. All images have size 384x256.

Figure 3 is the query result of Image 13. There are six images returned. The image in the first row is the query image. The images in the next two rows are listed in descending ranks. It is easy to see that all of them have a blue background (light or dark) and an object as the foreground. The foreground objects are in similar locations in all images. Therefore, all images are similar in color, texture and spatial information.

Figure 4 gives the query result of Image 18. Six images are returned and displayed in a similar way to that in Figure 3. Clearly, all images contain a large amount of green color with some foreground objects located approximately at the center of each image. Moreover, they are similar in a semantic sense because all images contain animals or flowers against green grass.

The query result of Image 621 is shown in Figure 5. Only four images are returned. We can observe that all images contain a large amount of black color and some amount of green color. The Rank 1 and Rank 4 images contain less amount of green color, while the Rank 2 and Rank 3 images contain more amount of green color. Although the blue color in Rank 2 image does not show up in the query image, it contains much green color as the query image does. In addition, the locations of the foreground objects in all images are approximately the same.

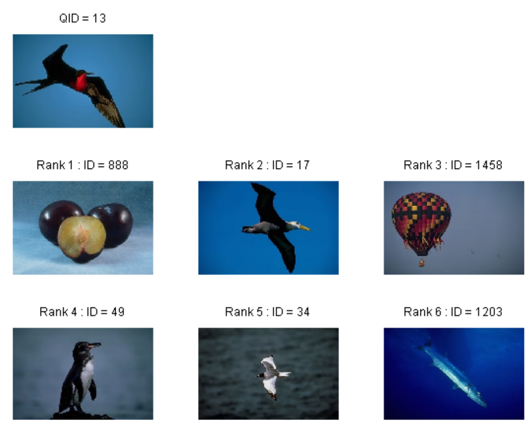

Figure 3. Query result of Image 13.

Figure 4. Query result of Image 18.

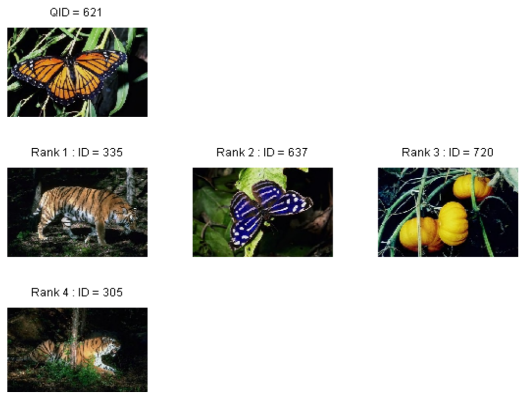

Figure 5. Query result of Image 621.

CONCLUSION AND FUTURE WORK

In this paper, we proposed an efficient multi-filter retrieval framework for large image databases. It employs three filters to dramatically reduce the query search range at different stages and thus save a huge amount of time taken unnecessarily in feature comparison at each stage. Our framework is unique for several reasons. First, a novel color label histogram with only thirteen bins is used. It effectively and efficiently extracts the global color information by classifying the pixel colors into a small number of color categories. Second, a novel distance measure considering the relationship between the wavelet coefficient value ranges and the decomposition levels is proposed. Third, a unique unsupervised segmentation algorithm together with the wavelet technique is adopted to automatically extract the spatial information of the images. Using the wavelet technique in generating the initial partition doubles the performance of the segmentation algorithm.

We are planning to integrate the SPCPE algorithm with more classes into our current framework. Relevance Feedback and supervised learning schemes will also be considered in the future.

REFERENCES

[1] Rui, Yong and Thomas S. Huang, "Image Retrieval: Current Techniques, Promising Directions and Open Issues," *Journal of Visual Communication and Image Representation*, Vol. 10, 39-62, March, 1999.

[2] Ma, Wei-Ying and HongJiang Zhang, "Content-Based Image Indexing and Retrieval", 1999 *Handbook of Multimedia Computing*, CRC Press, 227-254.

[3] Flicker, M., H. Sawhney, etc., "Query by Image and Video Content: The QBIC System," *IEEE Computers,* Sept. 1995.

[4] Smith, John R. and Shih-Fu Chang, "VisualSEEK: A Fully Automated Content-Based Image Query System," *Proc. of 1996 ACM Multimedia*.

[5] Wang, J., G. Wiederhold, and O. Firschein, "System for Screening Objectionable Images Using Daubechies' Wavelets and Color Histograms," *Proc. of Interactive Distributed Multimedia Systems and Telecommunication Services,* 1997.

[6] Androutsos, D., "Efficient Indexing and Retrieval of Color Image Data Using a Vector-Based Approach," Ph.D. Dissertation, University of Toronto, 1999.

[7] Sista, S. and R.L. Kashyap, "Bayesian Estimation For Multiscale Image Segmentation," *Proc. of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing,* Phoenix, AZ, March 1999.

[8] "Wavelet Toolbox User's Guide," The Mathworks Inc. 2000.

[9] Daubechies, Ingrid. *Ten Lectures on Wavelets*. Capital City Press, 1992.

[10] Wang, James-Ze, Gio Wiederhold, Oscar Firschein, Sha-Xin Wei, "Content-based Image Indexing and Searching Using Daubechies' Wavelets," *International Journal of Digital Libraries*, P311-328, 1997.

[11] Chen, Shu-Ching, Mei-Ling Shyu, Chengcui Zhang, and R.L. Kashyap, "Video Scene Change Detection Method Using Unsupervised Segmentation and Object Tracking," *IEEE International Conference on Multimedia and Expo (ICME)*, pp. 57-60, 2001.

[12] Li, Xiuqi, Shu-Ching Chen, Mei-Ling Shu, and Borko Furht, "A Novel Hierarchical Approach For Image Retrieval Using Color and Spatial Information," *submitted to* 2002 *IEEE International Conference on Multimedia and Expo (ICME)*.

[13] Li, Xiuqi, Shu-Ching Chen, Mei-Ling Shu, and Borko Furht, "An Effective Content-Based Visual Image Retrieval System," accepted for publication, 2002 *IEEE Computer Society International Computer Software and Applications Conference (COMPSAC)*.

[14] *http://wang1.ist.psu.edu/docs/related*.