# A Novel Hierarchical Approach to Image Retrieval Using Color and Spatial Information

Xiuqi Li[1], Shu-Ching Chen[2]*, Mei-Ling Shyu[3], Sheng-Tun Li[4], and Borko Furht[1]

[1] Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431, USA
[2] Distributed Multimedia Information System Laboratory
School of Computer Science
Florida International University
Miami, FL 33199, USA
[3] Department of Electrical and Computer Engineering
University of Miami
Coral Gables, FL 33124, USA
[4] Department of Information Management
National Kaohsiung First University of Science and Technology
Juoyue Rd. Nantz District
Kaohsiung 811, Taiwan, R.O.C.

**Abstract.** A novel hierarchical approach to image retrieval is proposed. First, a color label histogram is used to effectively filter out the images that are not similar to the query image in color. The proposed color label histogram built by categorizing the pixel colors is computationally much more efficient compared to other approaches. Next, the class parameters of those images passing the first filter are used to identify the images similar to the query image in spatial layout. These class parameters are obtained automatically from the proposed unsupervised segmentation algorithm. Moreover, the wavelet decomposition coefficients are used to generate the initial partition for the segmentation algorithm. It doubles the segmentation performance. At the last stage, all images passing two filters are ranked based on the total normalized distance in color and spatial layout. The experiments show the effectiveness and efficiency of our approach.

## 1 Introduction

Owing to the recent advances in hardware, managing large number of images has become ordinary. This leads to a growing interest in querying images based on the content of the images. Traditionally, images are retrieved using a text-based approach. In this approach, each image is manually annotated and then the retrieval process is converted into retrieval of the keywords in text descriptions

* This research was supported in part by NSF CDA-9711582 and NSF EIA-0220562.

of images. There are several inherent problems in such systems. First, manual annotation is often subjective, inaccurate, and incomplete [1]. Secondly, some of the image properties cannot be described using keywords. Because of these reasons, the content-based approach was developed to query images directly based on their visual attributes such as color, texture, shape, layout, object location, etc. without resort to human annotation.

In most of the content-based image retrieval systems, the goal is to find the top N images that are similar to the user query image [1][3]. Because each image has many visual features, similarity comparison based on a single feature is not enough. There have been some hierarchical approaches to content-based image retrieval combining multiple visual features. In [2], a color histogram filter and wavelet-based shape matching were utilized to query and screen objectionable images.

Our approach is different from the previous approaches in three aspects. First, a novel color label histogram is proposed. By categorizing the pixel colors into a set of thirteen colors and by labeling each pixel based on the color category ID, a histogram with only thirteen bins is obtained. It effectively and efficiently captures the global color information. Secondly, a unique unsupervised segmentation algorithm is applied to images to extract the information about the relationship between the pixel intensities and their spatial layout. Thirdly, wavelet decomposition is used to improve the performance of the segmentation algorithm.

The rest of the paper is organized as follows. In section 2, first, an overview of our query framework is given. Then, the color label histogram filter is presented. Next, the unsupervised segmentation parameter filter and the initial partition generation are presented. Finally, the query ranking is described. Section 3 shows the experimental results. Concluding remarks are given in Section 4.

## 2 Hierarchical Query Framework

Figure 1 illustrates our hierarchical query framework. Before the query, the color label histogram of each image in the image database is extracted offline. The results are stored for later filtering. Each image in the database is also segmented by the SPCPE (Simultaneous Partition and Class Parameter Estimation) algorithm [5][6] offline. The class parameters are generated and stored for later filtering. The query image is processed in the same way as any other image in the database. During the query, the color label histogram and the class parameters of the query image are compared to those of each image in the database. The comparison is performed in two stages. First, a color label histogram filter is used to eliminate all images that are not similar to the query image in color. Those images that passed the color filter are further compared to the query image using the class parameter filter. The second filter uses the class parameters obtained from the SPCPE algorithm to filter out those images that are not similar to the query image in spatial layout. At the end, all images that pass the

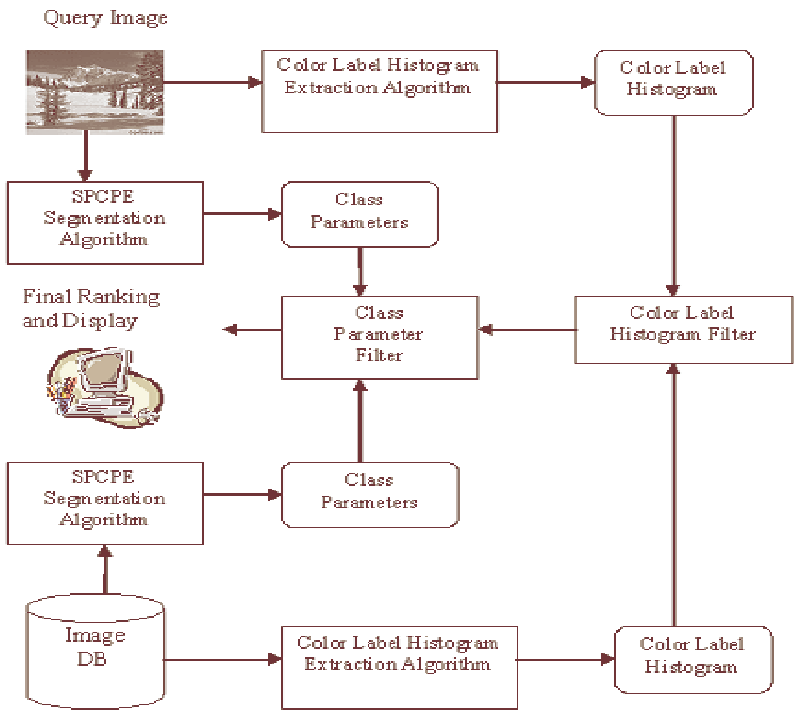


**Fig. 1.** The hierarchical framework

two filters are ranked based on the total normalized color and class parameter distance and the top six (or fewer) images are displayed in the user interface.

Next, we discuss in more details the color label histogram filter, the unsupervised segmentation parameter filter, and the final query ranking.

### 2.1 Color Label Histogram Filter

The image color is represented in a 3-channel color space. There are many color spaces, such as RGB, HSV, YUV, YIQ, CIE LAB, and CIE LUV. No color space is dominant in all applications.

In [4], the author used twelve color categories for the representative colors of color regions in an image. All categories are obtained from the experimental results based on the H, S and V value ranges. In our approach, the above categories are modified for our color histogram computation. To reduce the total number of histogram bins, the difference between the bright chromatic pixels and chromatic pixels is ignored. Therefore, bright blue and blue are considered to be in the same color category (BLUE). In addition, each transition slice is counted as a separate bin because each pixel should be counted only once in a color histogram. A new category "gray" is added because a color histogram takes into account of all pixels of all possible color values. After these modifications, the resulting color label histogram contains only thirteen bins (color categories). Compared to the color histogram with 512 bins used in [2], the color label histogram in our approach is computationally much more efficient without much loss of retrieval precision. Table 1 lists each color category and the corresponding H, S, V value ranges.

**Table 1.** Color category and HSV value ranges

| Color category ID | Color category | Hue range | Saturation range | Value range |
|---|---|---|---|---|
| 1 | White | Any | $< 20$ | $\geq 85$ |
| 2 | Black | Any | Any | $< 25$ |
| 3 | Gray | Any | $< 20$ | (25,85] |
| 4 | Red | [350°,25°) | | |
| 5 | Red-Yellow | [25°,45°) | | |
| 6 | Yellow | [45°,65°) | | |
| 7 | Yellow-Green | [65°,85°) | | |
| 8 | Green | [85°,160°) | $\geq 20$ | $\geq 25$ |
| 9 | Green-Blue | [160°,180°) | | |
| 10 | Blue | [180°,270°) | | |
| 11 | Blue-Purple | [270°,290°) | | |
| 12 | Purple | [290°,330°) | | |
| 13 | Purple-Red | [330°,350°) | | |

Histogram comparison between the query image $q$ and the $jth$ image in the database is based on the $L_1$-*Distance* [1], which is defined as follows:

$$D^{(q,j)}_{colorlabelhist} = \sum_{i=1}^{N} \left| X_i^{(q)} - X_i^{(j)} \right| \tag{1}$$

where $X_i$ is the $ith$ bin and $N$ is the total number of bins.

A threshold value is used by the color label histogram filter to eliminate the images in the database that are not similar to the query image in color.

## 2.2 Unsupervised Segmentation Class Parameter Filter

Given a gray-scale image, the SPCPE algorithm [5][6] partitions it into $s$ *regions (classes)* that are mutually exclusive and totally inclusive. Each class consists of one or more *segments* that are similar to each other in some sense and may not be spatially contiguous. Therefore, each image is partitioned into $s$ classes and $b$ segments. In the SPCPE algorithm, both the *class parameters* $\theta$ and the *partitions* $C$ are considered as random variables. The algorithm estimates $C$ and $\theta$ to be that which *maximizes* the *a-posterior probability (MAP)* of the partition variable and class parameter variable given the image data Y. Specifically, the algorithm begins with an initial partition, estimates $C$ and $\theta$ iteratively and simultaneously, and stops when the partition cannot be further optimized (the cost function reaches a local minimum).

**Initial Partition Generation.** The SPCPE algorithm starts with an initial partition and optimizes it using the least square technique and re-labeling rule. It is found in our experiment that the initial partition is very important and different initial partitions lead to different segmentation results. To produce a better result, the wavelet decomposition coefficients [7][8] are used in the initial partition generation. Without loss of generality, assume that there are two classes. The algorithm estimates $C$ and $\theta$, which has the least cost $J$ [5][6].

Our idea of using wavelet transform for initial partition generation is to label pixels as different classes based on the wavelet coefficient values. Images are first decomposed using Haar wavelet at level one. Next, salient points in horizontal, vertical and diagonal subbands are extracted by thresholding. For each of the three subbands, all pixels in the original image that correspond to the salient points in that subband are labeled as one class, and the rest of the pixels are labeled as the other class. This generates three candidate initial partitions. The final initial partition is the one with the least cost $J$ among the three candidates. Compared to the random initial partition generation, the segmentation precision is doubled with the help of the wavelet technique.

**Class Parameter Filter.** The unsupervised segmentation filter applies the SPCPE algorithm to the query image and all the images in the database to generate the class parameters. Then the filter compares the class parameters of the query image to those of the images passing the color label histogram filter. It filters out the images in its search range whose class parameters are much different from those of the query image.

Class parameter comparison is based on the sum of the Euclidian Distance of each corresponding class parameter between the query image $q$ and the $jth$ image in the search range.

$$D_{classpar}(q,j) = \sum_{m=1}^{NC} \sqrt{\sum_{i=0}^{3} \left(a_{mi}^{(q)} - a_{mi}^{(j)}\right)^2} \tag{2}$$

where $NC$ indicates the total number of classes and $a_{mi}$ is the $ith$ class parameter for class $m$.

### 2.3 Final Query Ranking

After passing the above two filters, images are sorted in descending order based on the sum of the normalized color label histogram distance and the normalized class parameter distance. The top six (or fewer) images are returned and displayed in the user interface. It is found that the class parameter distance is much larger than the color label histogram distance. Therefore, the two distances need to be normalized before the sum is computed. Normalization is implemented by dividing each color/parameter distance by the maximum color/parameter distance among all color/parameter distances between the query image and all the images that passed the two filters.

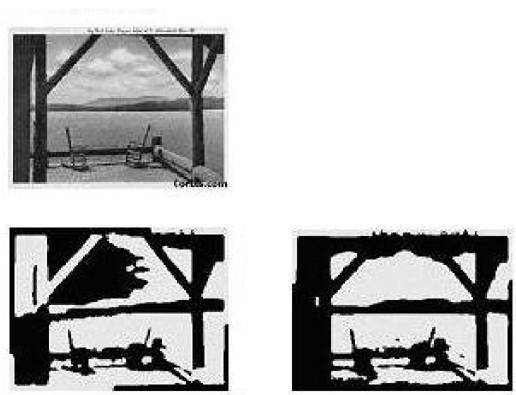

**Fig. 2.** Initial partition generation for Image 388.

## 3 Experimental Results

The experiments were conducted on various natural scene images, which were downloaded from yahoo (*www.yahoo.com*) and corbis (*www.corbis.com*). They vary in color and spatial layout. Their sizes are 256x192.

### 3.1 Experiments on Initial Partition Generation and Filtering Effects

Figure 2 shows the comparison of the initial partition generation effect using the random generation and the wavelet technique for Image 388. The image in the first row is the original image. The left image in the second row is the final segmentation result from a randomly generated initial partition. The right image in the second row is the final segmentation result from an initial partition generated through the wavelet transformation. It can be clearly seen from this figure that the segmentation result using the wavelet initial partition is much better than that of the random initial partition. With the wavelet initial partition, the porch, chairs, and the mountain are identified as the foreground, while the ocean and the sky are identified as the background. However, without using the wavelet initial partition, the top-left section of the porch was wrongly classified as the background, and part of the sky is mistakenly classified as the foreground.

**Table 2.** Experimental result on the filtering effect

| Filter | Avg(%) | Max(%) | Min(%) |
|---|---|---|---|
| Color Filter | 80 | 95 | 70 |
| Class Parameter Filter | 85 | 90 | 75 |

To evaluate the filtering effect, we computed the average, maximum, and minimum percentage of images eliminated from the search range of each filter based on all images in the database. Table 2 shows the experimental result. From Table 2, we can see that the two filters dramatically reduce the number of images that require computation for the following stage. Therefore, the query is speeded up.

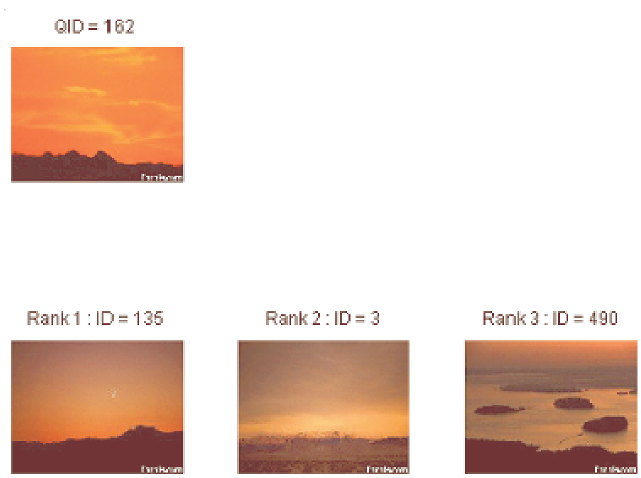


**Fig. 3.** Query result of Image 162

### 3.2 Experiment on the Retrieval Performance

The query result of Image 162 is shown in Figure 3. The image in the first row is the query image. The top three similar images and their ranks and image IDs are displayed in the next two rows. There are only three images returned. As can be seen from this figure, the result is quite good. The query image and the top three images contain two major colors: red and black. As for the spatial layout, the query image is very similar to the images with Rank 1 and Rank 2. All of them consist of a top area and a bottom area. The image with Rank 3 is a little bit different. There are several small dark areas on the top half of the image. However, the major areas are still the top and bottom ones.

## 4 Concluding Remarks

In this paper, a hierarchical framework for content-based image retrieval is proposed. A novel color label histogram, a unique unsupervised segmentation algorithm, and the wavelet technique are integrated in our framework. Before the query process, the color label histogram and the class parameters are extracted from all the images in the database offline.

During the query process, the color label histogram filter and the class parameter filter are used to filter out images that are not similar to the query image in color and spatial layout, respectively. All images passing the two filters are ranked based on the total normalized distance at the final stage. The top six (or fewer) images are returned in the interface. The experimental result demonstrates the effectiveness of our framework.

## References

[1] Ma, W., Zhang, H.J.: Content-Based Image Indexing and Retrieval. Handbook of Multimedia Computing, CRC Press (1999)

[2] Wang, J., Wiederhold, G., Firschein O.: System for Screening Objectionable Images Using Daubechies Wavelets and Color Histograms. Proc. of Interactive Distributed Multimedia Systems and Telecommunication Services (1997)

[3] Flicker, M., Sawhney, H., etc.: Query by Image and Video Content: The QBIC System. IEEE Computers, Vol. 28. (1995)

[4] Androutsos, D.: Efficient Indexing and Retrieval of Color Image Data Using a Vector Based Approach. PhD Dissertation. (1999)

[5] Sista, S., Kashyap, R.: Bayesian Estimation For Multiscale Image Segmentation. IEEE International Conference on Acoustics, Speech, and Signal Processing, Phoenix (1999)

[6] Chen, S.-C., Sista, S., Shyu, M.-L., Kashyap, R.: An Indexing and Searching Structure for Multimedia Database Systems. IST/SPIE Conference on Storage and Retrieval for Media Databases. (2000) 262–270

[7] Wavelet Toolbox Users Guide. The Mathworks Inc. (2000)

[8] Daubechies, I.: Ten Lectures on Wavelets. Capital City Press. (1992)