

# An Integrated Color-Spatial Approach to Content-based Image Retrieval

Wynne Hsu, Chua T. S., and Pung H. K.

Department of IS and CS

National University of Singapore

Singapore

(65)-772-7944

{whsu, chuats, punghk}@iscs.nus.sg

## ABSTRACT

*The use of color information for image retrieval has been used widely in many content-based retrieval systems with some success. However, histogram-based color retrieval techniques suffer from a lack of important spatial knowledge. We discuss a technique of integrating color information with spatial knowledge to obtain an overall impression of the image. The technique involves three steps: the selection of a set of representative colors, the analysis of spatial information of the selected colors, and the retrieval process based on the integrated color-spatial information. Two color histograms are used to aid in the process of color selection. After deriving the set of representative colors, spatial knowledge of the selected colors is obtained using a maximum entropy discretization with event covering method. A retrieval process is formulated to make use of the spatial knowledge to retrieve relevant images. A prototype image retrieval system has been implemented on the Unix system. It is tested on an image database consisting of 260 images. The result shows substantial improvement over the histogram-based color retrieval methods.*

**KEYWORDS:** *content-based retrieval, color retrieval, spatial retrieval, image segmentation*

## 1 INTRODUCTION

Many techniques have been proposed to perform image retrieval. They can roughly be classified into three categories: text-based retrieval, content-based retrieval, and semantic-based retrieval. Each method has its own strengths and weaknesses. In the text-based retrieval technique, each image has a number of keywords describing the image itself. The retrieval technique performs keyword matching in order to retrieve relevant images. In the content-based retrieval technique, an image is analyzed based on some

generic attributes such as colors, textures, and shapes. This information is used as a basis for retrieval. In the semantic-based retrieval technique, semantic meanings are used to retrieve relevant images. Typically, some form of knowledge base is required in the semantic-based retrieval systems. Table 1 gives a summary of the advantages and disadvantages of the three classes of retrieval techniques.

For large image databases, unless the images have accompanying text captions, it is often too much to expect a user to input the text descriptions manually. Besides, the manual entry of text descriptions is likely to be error-prone with incomplete or inconsistent text descriptions. In addition, some of the image databases are not confined to specific domains. In situations such as these, content-based retrieval techniques are preferred.

Existing content-based retrieval techniques can typically be classified as retrieval by colors, retrieval by textures, or retrieval by shapes. Shape information retrieval is, by and large, limited to specialized retrieval systems; whereas colors and textures retrieval techniques have been used in a number of general and fully automated retrieval systems. In 1992, Nagasaka & Tanaka [1] developed the color-pair matching technique to model distinct object boundaries. Chua et al. [18] further extended the color-pair technique to perform retrieval of segmented images. Shortcomings in many of the existing color-based retrieval techniques [19, 1, 18] include: the inability to recognize similar objects with different colors, the extreme sensitivity of the techniques to the scaling of an object,

Methods	Advantages	Disadvantages
Text-based	easy to specify abstract query	performance only as good as text descriptions
Content-based	automatic extraction of attributes	hard to specify abstract queries
Semantic-based	concept-based queries	domain-specific

Table 1: Comparisons of Image Retrieval Methods

and the poor tolerance of noisy images.

In this paper, an integrated color-spatial retrieval technique is proposed. This technique aims to overcome the above-mentioned shortcomings through a three-step process. In the first step, some heuristic rules are formulated to find the set of representative colors. Then, this set of representative colors is used as the basis for obtaining relevant spatial information through a maximum entropy discretization process. Finally, the information obtained in step two is used to retrieve relevant images from an image database.

The outline of this paper is as follows. Section 2 reviews the related works on image content retrieval. Section 3 describes the integrated color-spatial retrieval approach. Section 4 details the implementation of a prototype system tested on an image database of over 260 images. Lastly, section 5 contains the conclusions.

## 2 RELATED WORK

With the increasing emphasis on multimedia applications, the production of image and video information has resulted in large volumes of images and video clips that need to be properly indexed for future use. As a result, several techniques have been developed to query image databases by their image content. For example, the system in [8] retrieves images based on a sketch. [19] retrieves images based on their color content. [10] retrieves images based on the use of image shapes (or icons). There is another technique called the pair-wise comparison technique that compares differences in corresponding pixel positions in two images. This technique considers spatial information of the images. However, the technique only works for equal size images and has not been shown to be better than the histogram method.

Specifically, content-based image retrieval can be classified into three categories: template matching, global features matching, and local features matching.

Template matching [16] is used to detect an object within an image. It is typically subjected to the problems of image noise and is not very useful if the search object changes its orientation and/or direction.

Due to the inherent problems in template matching, global features matching is used to retrieve images. Besides using one global attribute such as color as the sole basis for retrieval [4], techniques for combining multiple attributes have been investigated. Chua et. al. [17] studied combined queries involving color and texture whereas Niblack [5] developed a system that uses colors, texture and/or shapes to retrieve images.

Unfortunately, the global features image retrieval techniques are limited because they cannot take advantage of the local features in an image. Nagasaka & Tanaka [1] developed the color-pairs matching technique to model distinct boundaries of the objects in an image. This technique was further enhanced by Chua et. al. [18]. A related field to modeling objects in an image is the study of image segmentation tech-

niques. Image segmentation is used to identify homogeneous regions within an image. Various techniques are available to perform image segmentation. They include: histogram mode seeking [14] region growing [2, 6, 15], spatial clustering [7, 13] and split and merge [9]. A complete survey of the image segmentation techniques can be found in [11].

## 3 THE INTEGRATED COLOR-SPATIAL RETRIEVAL TECHNIQUE

In spite of the great effort to discover efficient and effective image retrieval techniques, much research is still needed in this area, particularly the content-based image retrieval process. It is clear that color plays a very important role in defining an image. It is also equally clear that color alone is not sufficient to characterize an image. Figure 1 shows two images with similar color compositions but appearing entirely different to a human observer. This is be-



Figure 1: Two Images with Similar Color Compositions

cause though the color composition of the two images are similar, the clustering of the colors and the positions of the color clusters are very different from each other. This leads us to the conclusion that in retrieving or recognizing an image, it is important to combine selected color attributes with relevant spatial information as the basis for retrieval.

Combining selected color attributes with spatial information raises several interesting research issues:

- the basis for the selection of a representative set of colors.
- the algorithm for obtaining the relevant spatial information for the selected colors.
- the process for retrieval using the integrated color-spatial information.

### 3.1 Selection of Color Attributes

Humans are particularly sensitive to large patches of colors [3, 20]. Two images appear to be similar to one another if they both have patches of similar color at roughly the same locations in the images. Thus, the two images in Figure 2 appear similar because they both have the same patches of colors at roughly the same positions. In addition, it was found that human eyes are naturally drawn towards the center of an image [12]. A small patch of red color will most likely be noted if it is located near the center of an image than if it is located near the edges. In view of



Figure 2: Two Similar Images

this, our color selection process must reflect these two human tendencies. A color is selected if (1) it occupies a large percentage of the pixels in the image; and/or (2) it occupies a significant percentage of the pixels within a pre-defined “window” located at the center of the image.

An image is represented as a 2-dimensional array of pixel values of the form  $w \times h \times n$ , where  $w$  and  $h$  are the number of pixels in the x- and y-axis respectively, and  $n$  is the number of bits per pixel. To compute the color content of an image, a color histogram is used.

The color histogram  $H(g, i)$  of image  $g$  can be obtained by counting the number of pixels with color intensity  $i$ . In our color selection process, two color histograms are computed. The first color histogram,  $H_g$ , represents the color composition of the entire image. The second color histogram,  $H_w$ , represents the color composition of the predefined central “window”. The second histogram is used in order to identify colors near the central region of the image. Such a region is likely to contain the main objects depicted in the image. Both color histograms are then sorted according to the number of pixels per color.

Two sets of colors are selected, one from each histogram. The first set forms the background colors, denoted by *Background* while the second set forms the object colors, denoted by *Object*. The heuristic used to determine the color selection is as follows:

1. Let  $c_{H_g}$  denote the sorted histogram of  $H_g$  in decreasing order of the number of pixels per color. Let  $c_{H_w}$  denote the sorted histogram of  $H_w$  in decreasing order of the number of pixels per color. Let  $N_c$  denote the total number of colors to be selected.
2. Assume we have the *procedure* next-non-similar-color( $H, S$ ). This procedure takes two input parameters:  $H$  is a sorted color histogram, and  $S$  is a set of colors. The procedure returns the first color in  $H$  which is not similar to any color in  $S$ .
3. Initialization  
 $Background[0] \leftarrow c_{H_g}[0]$   
 $Object[0] \leftarrow \text{next-non-similar-color}(c_{H_w}, Background)$
4. Iterate to find the set of *Background* and *Object* colors.  
For  $i = 1$  to  $N_c/2$   
 $Background[i] = \text{next-non-similar-color}(c_{H_g},$

$$Object[i] = \begin{matrix} Background \cup Object \\ \text{next-non-similar-color}(c_{H_w}, \\ Background \cup Object) \end{matrix}$$

The reasoning behind the heuristic rule is straightforward: the background color of an image is likely to occupy a large portion of an image space. Hence, the color with the largest number of pixels in  $H_g$  is likely to be the representative color for the background of an image. In addition, most image objects tend to cluster around the center of an image. Thus, chances are that the color with the most number of pixels in  $H_w$  is likely to be a representative color for the object in the image. By having two color histograms, the problem of having a relevant object color being hidden/distorted by the background color is avoided. This is because for a small object, the proportion of the object color is relatively small with respect to the background colors. Thus, for the one histogram method, this object color may be regarded as noise. On the other hand, by having a second histogram for the pre-defined window, the relevant object color is given the rightful proportion. Furthermore, in separating the set of chosen colors into two classes: *Background* and *Object*, it paves the way for retrieving images with similar objects but of different colors. The retrieval process is discussed in Section 3.3.

### 3.2 Obtaining Spatial Information

A matching/retrieving process that is solely based on the color attributes is found to be insufficient. Thus in this work, we try to find the cluster regions of the selected colors and use that as the basis for similarity computation.

#### 3.2.1 The Maximum Entropy Discretization Method

In this section, we discuss an algorithm for deriving the spatial information with respect to a chosen color attribute. The algorithm is based on the maximum entropy discretization method [21]. Consider the data of interest represented by  $n$  variables  $(X_1, X_2, \dots, X_n)$ . The  $k^n$ -discretization process is defined as one that finds a set of  $k$  points on each axis of the  $n$ -dimensional space. These  $n$  sets of points will partition the  $n$ -dimensional space into  $k^n$  cells,  $R = \{R_i | i = 1, 2 \dots, k^n\}$ , where the projection of  $R_i$  on the  $j$ th axis defines an interval  $I_{i(j)}$  on  $X_j$ , ( $1 \leq j \leq n$ ). Let  $P$  be the probability distribution on the  $n$ -dimensional space. The process quantizes  $P$  into  $P(R_i)$ ,  $i = 1, 2 \dots, k^n$  whose probability distribution is denoted by

$$\tau = \left( \begin{matrix} R_i \\ P(R_i) \end{matrix} \right), i = 1, 2, \dots, k^n$$

The Shannon entropy  $H$  associated with  $\tau$  is defined as

$$H(R) = - \sum_1^{k^n} P(R_i) \log P(R_i)$$

The entropy discretization process is to find the sets of boundary points that partition the outcome space into cells subject to the maximization of the Shannon entropy.

To illustrate maximum entropy discretization, we compare it with equal-width-interval discretization. Consider the example of a single variable X having the following 20 observed values: 0.1, 0.9, 1.5, 2.0, 2.8, 3.2, 3.3, 3.5, 3.7, 3.8, 4.0, 4.5, 4.9, 5.5, 6.0, 7.3, 8.5, 8.8, 9.1, 9.5. The histograms generated using the two methods are shown in Figure 3 and Figure 4 respectively. It is clear from Figures 3-4 that the maximum entropy discretization method shows the peak more clearly, thus permitting distinct events to be identified more easily.

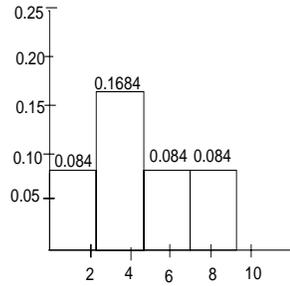


Figure 3: Histogram using Equal-Width-Interval Discretization

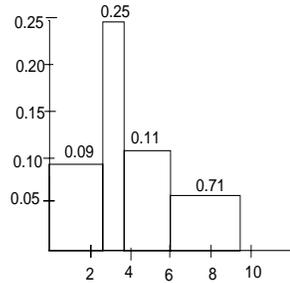


Figure 4: Histogram using Maximum Entropy Discretization.

### 3.2.2 Extracting Spatial Information for Images

We apply the maximum entropy algorithm to extract the clusters of selected colors in our images. For each selected color, the algorithm is applied to the image space in both x- and y-dimension. The results of the application of the algorithm is a set of representative regions for each selected color. For simplicity and without loss of generality, the region  $R_i$  is represented as a rectangular space  $(x_{min}^i, y_{min}^i, x_{max}^i, y_{max}^i)$ . After discretization, an event covering process is used to detect relevant spatial information. The algorithm works as follows. Initially, the entire image is regarded as one whole region. During the first pass, the image is partitioned into four regions based on

the maximum entropy discretization criterion. For each region, an evaluation criterion is used to determine whether further partitioning is needed. If the observed samples that fall within the region deviate significantly from the expected frequency, the region needs to undergo further partitioning. The expected frequency can be computed using three different methods: the uniformity assumption method, the independence assumption method, and finally the domain expert estimation.

Method 1 (uniformity assumption).

If no outcome is expected to occur more than the others, the probability distribution is uniform. The expected frequency of observations in a cell is estimated by

$$M \times \frac{\mu(I_{x_i})}{\mu(I_x)} \times \frac{\mu(I_{y_i})}{\mu(I_y)}$$

where M is the total number of data;  $I_{x_i}$  and  $I_{y_i}$  are the projected intervals of the cell i on X and Y respectively; and  $I_x$  and  $I_y$  are the extended intervals that enclose  $I_{x_i}$  and  $I_{y_i}$ , and  $\mu(I_{x_i})$  is the average or the expected value of  $I_{x_i}$ . Figure 5 shows the definitions of  $I_{x_i}$ ,  $I_{y_i}$ ,  $I_x$ , and  $I_y$ , respectively.

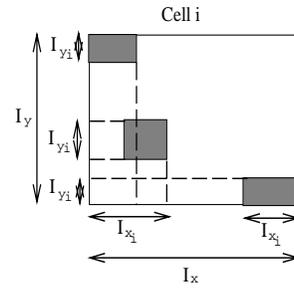


Figure 5: Figure showing  $I_{x_i}$ ,  $I_{y_i}$ ,  $I_x$ , and  $I_y$ .

Method 2 (The independence assumption). If the outcomes from  $I_{x_i}$  and  $I_{y_i}$  are independent, the expected frequency on cell i is

$$\frac{M_x(i) \times M_y(i)}{M}$$

where  $M_x(i)$  and  $M_y(i)$  denote the marginal frequency of the outcomes on  $I_{x_i}$  and  $I_{y_i}$ , respectively.

Method 3 (expert estimation assumption). An expert will supply the estimated expected frequencies.

Once the expected frequencies are calculated, the deviation can be computed by

$$D = \frac{(obs(i) - exp(i))}{\sqrt{(exp(i))}}$$

If D exceeds a certain threshold, then the cell is covered and no further partitioning is needed. Otherwise, it is placed in a stack for subsequent passes. The partitioning process stops when either all the cells are

covered or the number of samples within a cell falls below the given limit.

The algorithm is outlined as follows:

1. Set R to the entire image space.
2. Push R onto stack.
3. Repeat
  - Pop  $R_i$  from stack.
  - If number of pixels in  $R_i > \text{SIZE}$  then
    - Partition R into 4 subregions  $\{R_j\}, j = 1, \dots, 4$ , subject to the SIZE constraint.
    - For each  $R_j$  do
      - \* Compute the value of D.
      - \* If  $D < \text{TOLERANCE}$
      - \* Then Push  $R_j$  onto stack
      - \* Else Output  $R_j$ .
  - Else discard  $R_i$ .
4. Until stack is empty.

This algorithm has two parameters: SIZE and TOLERANCE. SIZE specifies the minimum area of a region. If the area of a region falls below the SIZE limit, no further partitioning will be carried out. TOLERANCE specifies the allowable level of noise permitted in each region. By decreasing the TOLERANCE level, the retrieval technique is found to be tolerant towards noisy images. The output of the algorithm is the color-spatial information of the image. The color-spatial information of an image is represented as a list of two-tuples  $\{(c_1, r_1), (c_2, r_2), \dots, (c_n, r_n)\}$  where  $c_i$  denotes a chosen color, and  $r_i$  is a list of four-tuples in the form of  $\{(x_{min}^1, y_{min}^1, x_{max}^1, y_{max}^1), \dots, (x_{min}^n, y_{min}^n, x_{max}^n, y_{max}^n)\}$ . Tuple  $(x_{min}^i, y_{min}^i, x_{max}^i, y_{max}^i)$  denotes a rectangular region with  $(x_{min}^i, y_{min}^i)$  as the lower left-hand corner and  $(x_{max}^i, y_{max}^i)$  as the upper right-hand corner.

### 3.3 The Retrieval Process

Having found the color-spatial information, our next step is to use this information in the image retrieval process. However, first we must define quantitatively the degree of similarity between two given images. For this purpose, a computational model for the integrated color-spatial information of the image is derived.

#### 3.3.1 Computational Models for Color-Spatial Information

Given the color-spatial information, let  $C(i, k)$  denote the  $i$ th cluster of color  $k$ . Two computational models are defined for two matching criteria: direct and indirect matching. In direct matching (matching objects of the same color), the similarity measurement, L, between two images  $g_1$  and  $g_2$  is defined as follows:

$$L_{g_1, g_2} = \sum_{k=1}^{N_c} \sum_{i=1}^{N_k^{g_1}} \sum_{j=1}^{N_k^{g_2}} C_{g_1}(i, k) \cap C_{g_2}(j, k)$$

where  $N_c$  is the total number of colors in the representative set,  $N_k^{g_1}$  is the total number of clusters of color  $k$  in image  $g_1$ ,  $N_k^{g_2}$  is the total number of clusters of color  $k$  in image  $g_2$ , and  $C_{g_1}(i, k) \cap C_{g_2}(j, k)$  is the intersection between cluster  $C_{g_1}(i, k)$  and cluster  $C_{g_2}(j, k)$ .

In indirect matching (matching similar objects with different color), the similarity measurement,  $L^s$ , between images  $g_1$  and  $g_2$  is defined as:

$$L_{g_1, g_2}^s = \sum_{i \in Obj_{g_1}} \left( \max_{k \in Obj_{g_2}} \sum_{j=1}^{N_i} \sum_{m=1}^{N_k} C_{g_1}(j, i) \cap C_{g_2}(m, k) \right)$$

where  $\max x = \max \tilde{x}$  where  $\tilde{x} \in \{x \mid x \text{ satisfies the following criteria: the proportion of } x \text{ with respect to the total cluster area is greater than a pre-defined threshold. } Obj_{g_1} \text{ is the set of representative object colors in image } g_1, Obj_{g_2} \text{ is the set of representative object colors in image } g_2, N_i \text{ and } N_k \text{ are the number of clusters of color } i \text{ and color } k \text{ respectively, and } C_{g_1}(j, i) \cap C_{g_2}(m, k) \text{ is the intersection between two clusters.}$

Computing the intersection between two clusters requires two steps. First, we determine whether the two clusters overlap. Let the two clusters be  $C_1 = ((x_1, y_1)(x_2, y_2))$  and  $C_2 = ((x_3, y_3)(x_4, y_4))$ . All the possible configurations of the two clusters are shown in Figure 6.

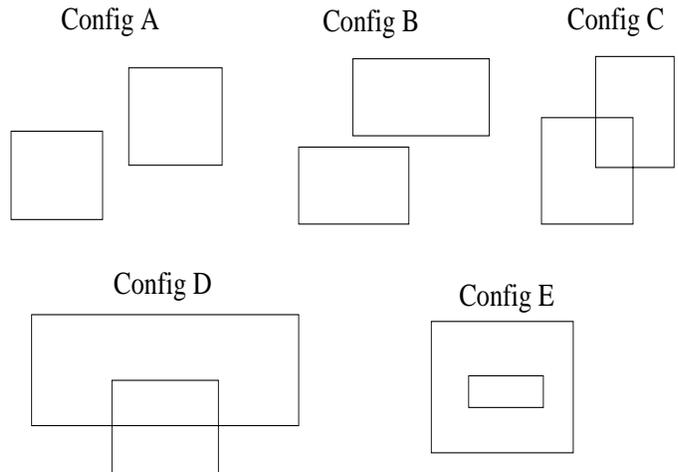


Figure 6: The Possible Configurations of Two Clusters.

In configuration A and configuration B, the two clusters do not overlap. These two cases correspond to the conditions:  $(x_2 < x_3 \text{ or } x_4 < x_1)$ , and  $(y_2 < y_3 \text{ or } y_4 < y_1)$ , respectively. In such cases,  $C_1 \cap C_2 = 0$ . Suppose the two clusters do overlap as in the case of

configuration C, configuration D, and configuration E. The degree of overlap is computed by first sorting the x coordinates and the y coordinates of the two clusters. Let  $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4$  and  $\tilde{y}_1, \tilde{y}_2, \tilde{y}_3, \tilde{y}_4$  denote the sorted lists of x and y coordinates respectively. The intersection region is given by

$$C_1 \cap C_2 = (\tilde{x}_3 - \tilde{x}_2) \times (\tilde{y}_3 - \tilde{y}_2)$$

### 3.4 Making the Comparison Orientation-Independent

With this clustering representation, we can extend our similarity comparison to images with different sizes, different orientations, and different positions. This is because we can transform a cluster of one image into the corresponding cluster of the second image (having different size, orientation, or position) by simply operating on the tuples with a suitable transformation matrix. Some common transformations can be modeled as given below:

1) Scaling

$$\begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} s_x x \\ s_y y \end{pmatrix}$$

2) Rotation anticlockwise about the origin

$$\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x\cos\theta + y\sin\theta \\ -x\sin\theta + y\cos\theta \end{pmatrix}$$

3) Translation

$$\begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} \delta x \\ \delta y \end{pmatrix} = \begin{pmatrix} x - \delta x \\ y - \delta y \end{pmatrix}$$

By premultiplying the first image clusters' coordinates with the desired transformation matrix, the first image clusters are transformed to match the second image clusters. In this way, the two images can be compared meaningfully.

## 4 SYSTEM DESIGN AND IMPLEMENTATION

A prototype system is designed and implemented on the UNIX system using C programming language. The overall system architecture is shown in Figure 7. It consists of two main modules: the preprocessing module and the comparison module. Two databases are used. One is for storing the integrated color-spatial informations of all the images, while the other contains the actual images.

Queries are issued by image example. The user may either enter the image id of the query image or input a new image example. The system implements two retrieval techniques: the color histogram technique and the integrated color-spatial technique. Each query image is tested using the two techniques. The integrated color-spatial technique has been discussed in Section 3. For completeness's sake, we now outline briefly the computational model used in the color histogram technique.

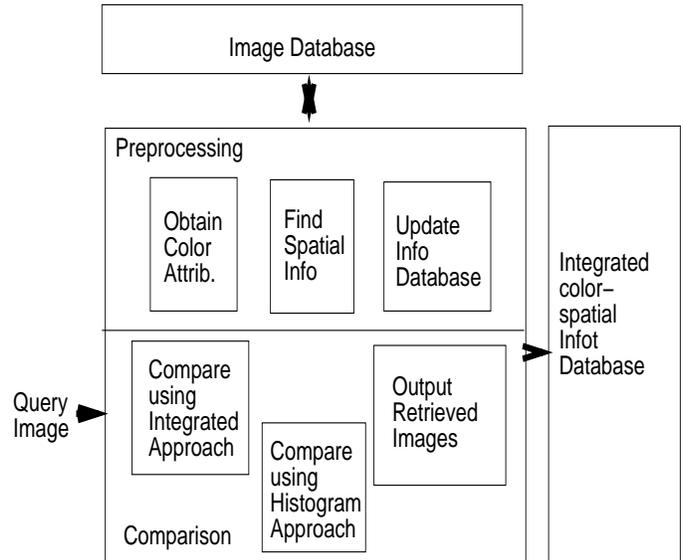


Figure 7: The Overall System Architecture.

Assuming there are  $n$  color intensities per pixel for the images, the difference between the histograms of two images,  $g_1$  and  $g_2$  is defined as follows:

$$E_H(g_1, g_2) = \sum_{j=0}^{n-1} |H(g_1, j) - H(g_2, j)|$$

Chua et al. extends the work to develop the similarity measure for the color model. Let  $NH(g, j)$  denote the normalized color histogram. The normalized color history is obtained by dividing each entry of the histogram by the total number of pixels of the image.

The similarity measurement,  $SIM_{C1}(g_1, g_2)$ , between two images,  $g_1$  and  $g_2$ , is defined as follows:

$$SIM_{C1}(g_1, g_2) = \frac{\sum_{j \in S_{1N}} (1 - \frac{|NH(g_1, j) - NH(g_2, j)|}{\max\{NH(g_1, j), NH(g_2, j)\}})}{N}$$

where  $S_{1N}$  is the set of all non-zero color intensities of image  $g_1$ , and  $N$  is the total number of elements in  $S_{1N}$ .  $SIM_{C1}(g_1, g_2)$  takes values between 0 and 1.

Once the similarity measures are computed, all those images with non-zero similarity measures are ranked and displayed.

## 5 TESTING AND RESULTS

### 5.1 Image Database

The image database selected for testing consists of 260 images from a wide variety of categories. This is to give a more accurate reflection of the results since the retrieval algorithm is for general applications. The major categories of these images are listed in Table 2. These images are scanned into the system using a desk-top color image scanner. The images are stored in 256 colors with sizes of up to 200 by 200 pixels.

## 5.2 Retrieval Results

To test the effectiveness of the retrieval technique developed, six queries are chosen for evaluation purposes. The queries are chosen from different categories ranging from size 18 to size 4. The categories corresponding to Query1 - Query6 are: Swim Suit Model, Painting, Portrait, Farming Field, Jelly Fish and, Water Activities, respectively. Queries are tested using two separate retrieval techniques – the color histogram technique and the integrated color-spatial technique.

Before presenting the results, let us first define the parameters used in the experiments. Recall that in the color selection process, a pre-defined window is used to capture the object colors. In our experiments, the size of this pre-defined window is set to 100 by 100 pixels located at the center of the image. During the discretization process, the value of SIZE is set to 64 and the value of TOLERANCE is set to 0.35. Uniformity assumption is made in the computation of D. In addition, the threshold value for the integrated approach using indirect matching criteria is 0.6.

For each query, Query1 - Query6, three experiments are performed. In the first experiment, the integrated approach using direct matching (DI) criteria is used for the retrieval process. In the second experiment, the indirect matching (II) criteria is used. Finally, as a basis for comparison, the histogram (H) approach is used in the third experiment.

The results of retrieval are presented on the screen one at a time in decreasing order of similarity measure. The results of retrieval are summarized in terms of recall and precision. The definition of recall and precision is given below.

	Relevant	Not Relevant
Retrieved	A (correctly retrieved)	B (incorrectly retrieved)
Not Retrieved	C (missed)	D (correctly rejected)

$$\text{recall} = \frac{\text{relevant retrieved}}{\text{all relevant}} = \frac{A}{A + C}$$

$$\text{precision} = \frac{\text{relevant retrieved}}{\text{all retrieved}} = \frac{A}{A + B}$$

Category	Relevant Images
Portrait	18
Evening Scene	17
Farming Field	5
Forest	10
Painting	9
Water Activities	11
Grass View	7
Tortoise	7
Jelly Fish	7
Mountain View	7
Chinese Opera	6
Night Scene	5
Camera	5
Snow Scene	5
Tram Board	5
Harbour View	4
Chef Show	4
Sea Rock	4
Coral	4
Swim Suit Model	4
Fish Species A	4
Fish Species B	4
Fish Species C	4
Fish Species D	4
Fish Species E	4
Fish Species F	4

Table 2: The Major Categories in the Image Database.

Table 3 gives the summary of the results. From the table, we see that the integrated approach (DI or II) generally performs better than the color histogram method (H). For Query 2, when the recall level is low, the histogram method actually out-performs the integrated method. This is because the query category in Query2 is Painting. In this category, color, and not shape, plays a more important role in determining whether an image is relevant or irrelevant. Hence, when the recall level is low, the color histogram performs better than the integrated approach. However, as the recall level increases, the integrated approach

Recall	Precision					
	Query1			Query2		
	DI	II	H	DI	II	H
0.25	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>	0.33	0.2	<b>0.67</b>
0.5	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>	0.16	0.17	<b>0.5</b>
0.75	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>	0.21	<b>0.23</b>	0.167
0.9	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>	<b>0.25</b>	<b>0.25</b>	0.07

Recall	Precision					
	Query3			Query4		
	DI	II	H	DI	II	H
0.25	<b>0.71</b>	0.5	0.23	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>
0.5	<b>0.45</b>	<b>0.45</b>	0.07	<b>1.0</b>	0.67	0.67
0.75	<b>0.31</b>	0.30	0.07	<b>0.6</b>	0.375	0.143
0.9	0.21	<b>0.28</b>	0.08	<b>0.12</b>	-	0.08

Recall	Precision					
	Query5			Query6		
	DI	II	H	DI	II	H
0.25	<b>1.0</b>	<b>1.0</b>	0.11	<b>1.0</b>	<b>1.0</b>	<b>1.0</b>
0.5	0.375	<b>0.75</b>	0.1	<b>0.86</b>	<b>0.86</b>	0.325
0.75	<b>0.24</b>	0.13	0.05	<b>0.41</b>	<b>0.41</b>	0.086
0.9	<b>0.22</b>	0.11	0.05	<b>0.43</b>	<b>0.43</b>	0.08

Table 3: Results of Image Retrieval Experiments

finds all the relevant images with better precision. Next, we compare the performance of the integrated approach using the direct matching criteria versus the indirect matching criteria. In general, there is no significant difference in performance between the two. In fact, the direct matching criteria seems to perform much better in Query4. This is to be expected because Query4’s category is Farming Field. Using direct matching criteria, the images in Figure



Figure 8: Query4 Retrieval Samples

8 have a low similarity measure because the trees and the grass are of a different shades of green. This is what we would have expected because for a human observer, a grass-field and a tree-forest belong to different categories. However, in the indirect matching criteria, the specific object color is not important. As a result, the two images have a high similarity measure because of the large overlapping regions of the objects in the two images. Hence, precision suffers in this situation. Employing the same reasoning, we find that the indirect matching criteria works better in Query3 when the recall level is high. This is because Query3’s category is Portrait which consists of many human faces with slightly different skin colors,

eye colors or hair colors. In other words, the specific object colors are not critical in determining the relevancy of the image. Thus, depending on the application domain (whether specific object colors are critical in determining the relevancy of the images), either direct or indirect matching criteria is used. If the application domain is such that specific object colors are not critical, then indirect matching criteria is used, otherwise, direct matching criteria should be used.

## 6 CONCLUSIONS

We have shown the results for image retrieval using the integrated color-spatial approach. Comparisons of the results using the color-histogram method and the integrated approach have been discussed. Beside yielding a better average precision for image retrieval, the integrated approach is more tolerant to noise in the image and is flexible in dealing with similar objects of different colors, as well as in dealing with objects with slightly different sizes and orientations.

## References

- [1] Nagasaka A. and Tanaka Y., “Automatic video indexing and full-video search for object appearances,” *Visual Database Systems, II, IFIP*, pages 113–127, 1992.
- [2] T. Asano and N. Yokoya, “Image segmentation schema for low-level computer vision,” *Pattern Recognition*, 14:267–273, 1981.
- [3] J. Beck, “Perceptual grouping produced by line figures,” *Percept. Psychophys.*, 2:491–495, 1967.
- [4] Binaghi E., Gagliardi I., and Schettini R., “Indexing and fuzzy logic-based retrieval of colour images,” *Visual Database Systems, II, IFIP*, pages 79–92, 1992.
- [5] Niblack W. et. al., “The QBIC project: Querying images by content using colour, texture, and shape,” *SPIE*, 1908:173–187, 1993.
- [6] R. M. Haralick, “Digital step edges from zero crossing of second directional derivative,” *IEEE Trans. Pattern Anal. Mach. Intell.*, 6:58–68, 1984.
- [7] R. M. Haralick and G. L. Kelly, “Pattern recognition with measurement space and spatial clustering for multiple image,” *Proc. IEEE*, 57:654–665, 1969.
- [8] T. Kato, T. Kurita, and H. Shimogaki, “Intelligent visual interaction with image database systems – toward the multimedia personal interface,” *Journal of Information Processing*, 14(2):134–143, 1991.
- [9] A. Klinger, “Data structures and pattern recognition,” *Proc. of the First International Joint Conference on Pattern Recognition*, pages 497–498, Oct. 1973.
- [10] Stanchev P. L., Smeulders A. W. M., and Groen F. C. A., “An approach to image indexing of documents,” *Visual Database Systems, II, IFIP*, pages 63–77, 1992.

- [11] Haralick R. M. and Shapiro L. G., "Survey: Image segmentation techniques," *Computer Vision, Graphics, and Image Processing*, 29:100–132, 1984.
- [12] M. W. Matlin and H. J. Foley, *Sensation and Perception*, Allyn and Bacon, 3rd edition, 1992.
- [13] D. L. Milgram, "Region extraction using convergent evidence," *Computer Graphics Image Processing*, 11:1–12, 1979.
- [14] R. Ohlander, *Analysis of Natural Scenes*, PhD thesis, Carnegie-Mellon University, Pittsburgh, 1975.
- [15] T. Pavlidis, "Segmentation of pictures and maps through functional approximation," *Computer Graphics Image Processing*, 1:360–372, 1972.
- [16] W. K. Pratt, *Digital Image Processing*, John Wiley, 2nd edition, 1991.
- [17] Chua T. S., Chan S. K., Pung H. K., and Lu G. J., "A content-based image retrieval system," Technical report, Internal Report, DISCS, NUS, 1994.
- [18] Chua T. S., Lim S. K., and Pung H. K., "Content-based retrieval of segmented images," *ACM Multimedia*, pages 211–218, Oct. 1994.
- [19] Michael J. Swain and Dana H. Ballard, "Color indexing," *International Journal of Computer Vision*, 7(1):11–32, 1991.
- [20] A. Treisman and R. Paterson, "A feature integration theory of attention," *Cognit. Psychol.*, 12:97–136, 1980.
- [21] Chiu D. K. Y. and Kolodziejczak T., "Synthesizing knowledge: A cluster analysis approach using event-covering," *IEEE Transactions on Systems, Man, and Cybernetics*, 16(2):462–467, 1986.