Approximating Content-based Object-level Image Retrieval

Wynne Hsu, Chua T. S., and *Pung H. K.*
Department of IS and CS
National University of Singapore
Lower Kent Ridge Road,
Singapore 119260
Republic of Singapore

Abstract

Object-level image retrieval is an active area of research. Given an image, a human observer does not see random dots of colors. Rather, he/she observes familiar objects in the image. Therefore, to make image retrieval more user-friendly and more effective and efficient, object-level image retrieval technique is necessary. Unfortunately, images today are mostly represented as 2D arrays of pixels values. The object-level semantics of the images are not captured. Researchers try to overcome this problem by attempting to deduce the object-level semantics through additional information such as the motion vectors in the case of video clips. Some success stories have been reported. However, deducing object-level semantics from still images is still a difficult problem. In this paper, we propose a "principle-components" approach to approximate object-level image retrieval. The color and spatial information of the prinicple components of an object are estimated. The technique involves three steps: the selection of the prinicple component colors, the analysis of spatial information of the selected colors, and the retrieval process based on the 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 tool has been implemented on the Unix system. It is tested on an image database consisting of 260 images. The result shows that the "principle-components" approach are able to retrieve similar objects with much better precision than the sole color-based retrieval methods.

1 Introduction

Image retrieval techniques can roughly be classified into three categories: text-based retrieval, content-based retrieval, and semantic-based retrieval. In the text-based retrieval technique, each image is associated with a set of keywords that describe the image itself. Keywords matching are then performed to to retrieve the relevant images. In the content-based retrieval technique, an image is analyzed based on some generic attributes such as colors, texture, 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.

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

In spite of the progress made, image retrieval techniques are still far from being satisfactory. In this paper, we would like to look at the problem in terms of human perception understanding. Many studies have been conducted to understand how a human perceives an image. It is pointed out that human perception exhibits the following characteristics:

1. lightness/color constancy – When a human views an object under different levels of light/color, the perception of lightness/color tends to remain constant despite small variations. In other words, human is able to recognize an object even when the level of illumination or the range of color changes.

2. integration of local features – human has a strong tendency to group bits and pieces of information into a meaningful form, even when that form is only vaguely suggested by the bits and pieces. Consider Figure 2, we see the scene and the central object because of some global comparison of local contour information.

3. principles of proximity and similarity – All things being equal, if several simuli are presented together, there is a tendency to see the form in such a way that similar and close-by items are grouped together (see Figure 3).

Figure 1: Integration of Local Features

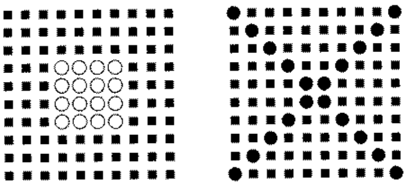

Figure 2: Principle of proximity and similarity

4. principle of closure – Closure is the tendency to unite contours that are very close to one another. Refer to Figure 4, instead of seeing a mirror reflection of the handwritten word "men", we see a series of hearts.

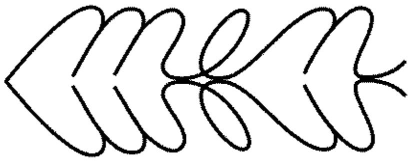

Figure 3: The Closure Effect

5. the scaling effect – The structures in Figure 5 captures different aspect of a city image. Drawing A emphasizes the skyline, Drawing B highlights the individual buildings, and Drawing C spotlights the architectural details. Yet, all three drawings represent the same city. In other words, the human visual sytem is capable of implementing such multiple-scale representation.

What is the implication of these human perception characteristics to the research in image retrieval techniques? Certainly, a good and effective image retrieval technique should, to some extent, model these human perception charactersitics. A survey of existing image retrieval techniques reveals the following: Text-based retrieval is based on the assumptions that it is possible to describe a visual image, completely and without ambiguity, using words. However, text-based description is generated by a human. From our list of human perception characteristics, we see

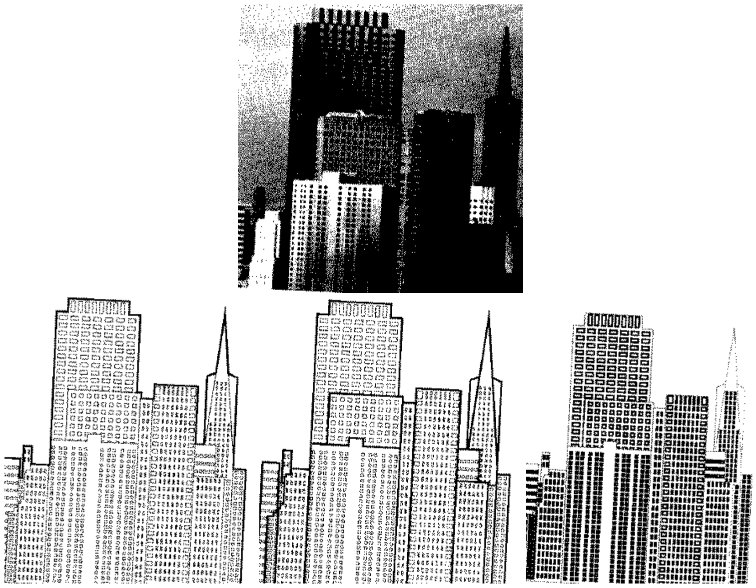

Figure 4: The Scaling Effect

that it is entirely possible for one human to look at an image and yet perceive a totally different objects/scenes from a second observer (see Figure 1). This strong

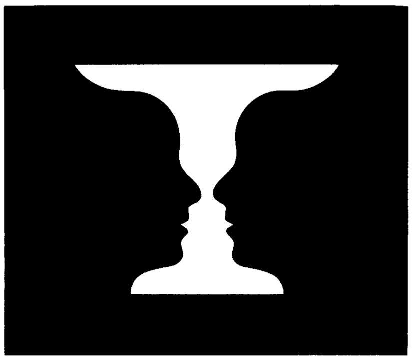

Figure 5: Two Different Interpretations of the Same Image. You can see the white vase against a dark ground or you can see the black faces against a white ground.

dependence on human's subjectivity is the primary limitations of text-based retrieval techniques. On the other hand, existing content-based retrieval techniques focus on describing a visual image using low-level attributes such as color, shape, textual information. Color-based retrieval techniques retrieve images with similar color composition. The histogram and the color-pairs method are the two most common techniques used for color-based retrieval. Both these methods are very sensitive to the scaling effect, and the lighting conditions. In addition, the histogram method completely ignored the effects of proximity and similarity in describing an image. Though the color-pairs method attempts to estimate such effects by dividing the image into groups of cells and analyzing the color-pairs content of the cells, it fails to truly catpure the proximity and similarity effects. The texture-based retrieval techniques emphasize on the needs to catpure sensitivity/resolution information, and proximity/ similarity information. Measures such as directionality, coarseness, and others are defined in an attempt to capture such information. Unfortunately, this approach ignores one important aspect of human's perception – the ability to integrate of local features to derive high-level meaning such as object recognition. Because of this, texture-based approach, by itself, fails to become an effective image retrieval tool. The semantic-based retrieval techniques look at the problem from another angle. Instead of attempting to capturing low-level perception information such as color, resolution, proximity, and others, the focus is in representing object-level perception information. For example, a concert scene usually composed of the stage, the audience, and the musicians. To retrieve images similar to a concert scene, a semantic-based retrieval tool will look for images that fit the descriptions of the stage object, the audience object, and the musicians object. This technique assumes the availability of an automatic object features extractor and object recognizer. However, this is not a realistic assumption because the general problem of object recognition is still largely unresolved. In this paper, rather than trying to overcome the general problem of feature extraction and object regonition, we propose a retrieval technique that "guesses" the color and proximity attributes of the principle components of the objects and uses these information to approximate object-level image retrieval. An image retrieval tool based on the proposed technique has been developed at the National University of Singapore.

The outline of this paper is as follows. Section 2 gives the related work. Section 3 describes the process of "guessing" the colors of principle components of objects. The estimation of the proximity information is detailed in Section 4. The testing and results are given in Section 5. Finally, Section 6 summarizes the findings.

2 Related Work

With the increased 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 image and also the technique 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 some pre-defined objects within an image. It is typically subjected to the problems of image noise and is not very useful if the search objects change its orientation and/or direction.

Due to the inherent problems in template matching, global features matching is used to retrieve images. One popular global feature matching technique is to perform matching based on the color attribute. Swain and Ballard citeSWAIN have proposed a histogram intersection technique. Their core idea is to compute:

$$H(I,M) = \frac{\sum_{j=1}^{n} min(I_j, M_j)}{\sum_{j=1}^{n} M_j}$$

where $H(I,M)$ is the match value, I and M, are the color histograms of the query image and a model image in the database respectively. Each histogram contains n bins. The closer the match value to unity, the more similar is the model image to the query image. Another method to perform matching based on color attribute is the distance method [12]. In this method, the color information is represented by capturing the mean value, μ, of the 1D histogram of each of the three color components of the image (μ_R, μ_G, μ_B). The distance between they query image and a model image can be computed using the Manhattan distance,$D^M_{q,i}$ or the Euclidean distance $D^E_{q,i}$

$$D^M_{q,i} = \sum_{R,G,B} |\mu_q, \mu_i|$$
$$D^E_{q,i} = \sqrt{\sum_{R,G,B} (\mu_q - \mu_i)^2}$$

In addition, we also have the reference color table method [12] whereby a set of reference colors is first defined. For every image, each pixel in the color image is assigned the nearest color from the table. Then the histogram of the pixels with the newly assigned colors is computed. This is then used for matching between a query image and a model image in the database. 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 the combined query 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] develops 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 techniques. 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 scheme [2, 6, 15], spatial clustering scheme [7, 13] and split and merge scheme [9]. A compute survey of the image segmentation techniques can be found in [11].

3 The Selection of Principle Component Colors

In general, object-level retrieval is difficult because of the lack of semantic meanings encoded in the representation of an image. An image is generally 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. Using image segmentation technique, we are able to isolate regions of homogenous colors. For example, given the image of a car, we are able to find the regions corresponding to the car windows, the car doors, and the car body. However, to automatically deduce that these regions can be merged to form an object (car object) is still beyond the grasps of researchers today. In this paper, instead of attempting to automatically extract features of an object and to recognize an object, we propose a method to "guess" the colors and spatial information of the principle components of the object. Given the colors and the spatial information of the principle components of the object, we then propose a retrieval technique that utilizes this information to retrieve images with similar objects. The motivation for this approach is based on two observations: (1) Human is able to recognize an object just by the presence of two or three major components [3]. An experiment was performed to determine whether the first few components that are available from an unoccluded view of a complete object would be sufficient for identification of the object. Figure 6 shows the partial and complete versions of some objects. It was found that, in general, human subjects are able to correctly identify an object with the presence of just 2 to 3 principle components of the object. (2) Human's perception tends to be focused. Consider Figure 7, the small black box is more likely to be noticed in (a) than in (b) because the black box is nearer to the center of focus in (a).

In view of these two observations, the color of the principle component of an object is selected as follows. The user first uses the cursor to select a focus window such that the object of interest lies inside this focus window. Two color histograms are computed. The first color histogram, H_g, represents the color composition of the entire image, g. The second color histogram, H_w, represents the color composition of the focus-window, w. Both color histograms are then sorted according to the number of pixels per color. In general, the color of the principle component of the object is one that occupies a significant number of pixels in the focus-window that is different from the background colors. To distinguish between object colors and background colors, a heuristic algorithm is used. The inputs to the algorithm are the two color histograms, H_g and H_w. The set of background colors and the set of object colors are determined 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

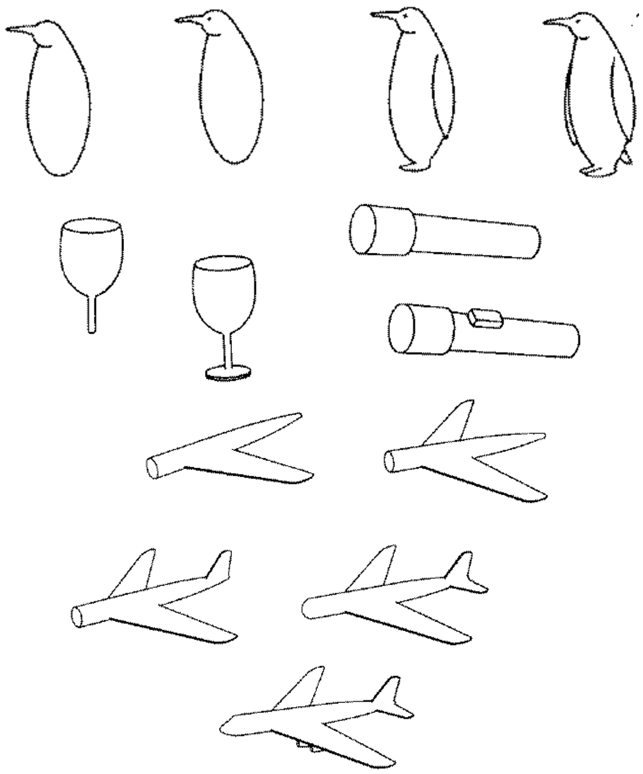

Figure 6: Partial and Complete versions of three-component objects (wine glass and flashlight) and nine-component objects (penguin and airplane

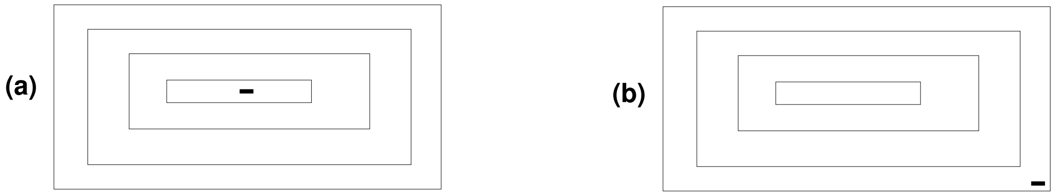

Figure 7: Illustration of human's perception tends to be focused

colors. The procedure returns the first color in H which is not similar to any color in S.

3. Initialization
   $Background[0] \leftarrow c_{H_g}[0]$
   $Object[0] \leftarrow$ next-non-similar-color(c_{H_w},
   $Background$)

4. Iterate to find the set of $Background$ and $Object$ colors.
   For i = 1 to $N/2$
   $Background[i]$ = next-non-similar-color(c_{H_g},
   $Background \cup Object$)
   $Object[i]$ = next-non-similar-color(c_{H_w},
   $Background \cup Object$)

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. On the other hand, the image object lies within the focus-window. Thus, chances are the colors that occupy the majority portion of pixels in H_w are the colors of the principle components of the object in the image. By using two color histograms, we are able to distinguish between the background colors and the object colors. This is important because usually for a small object, the proportion of the object colors is relatively small with respect to the background colors. Thus, if we just have one histogram for the entire image, important object colors may be regarded as noise. On the other hand, by having a second histogram for the focus-window, the relevant object colors are given the rightful proportion. With the colors of the principle components selected, the next step is to obtain the spatial information of the principle components.

3.1 Obtaining the Spatial Information of the Principle Components

For retrieval of images at the object-level, it is important to compute the similarity of the objects in the two images. A matching/retrieving process that is solely based on the color attributes is found to be insufficient for this purpose. Instead, we would like to compute the similarity of two images based on the amount of match between the principle components found in the two images.

A number of techniques are available to perform image segmentation. In this paper, we look at using the maximum entropy discretization method [20] to estimate the spatial information of the principle components. The basic idea is to estimate the spatial regions of the principle components by a series of rectangles.

3.1.1 The Maximum Entropy Discretization Method

Consider the data of interest represented by n variables $(X_1, X_2, \cdots, X_n)$. 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 = $\{R_i | i = 1, 2 \cdots, k^n\}$, where the projection of R_i on the jth axis define 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), $i = 1, 2 \cdots, k^n$ whose probability distribution is denoted by

$$\tau = \begin{pmatrix} R_i \\ P(R_i) \end{pmatrix}, i = 1, 2, \cdots, k^n$$

The Shannon entropy H associated with τ 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 histogram generated using the two methods are shown in Figure 8 and Figure 9 respectively. It is clear from Figures 8-9 that the maximum entropy discretization method shows the peak more clearly, thus permitting distinct events to be identified more easily.

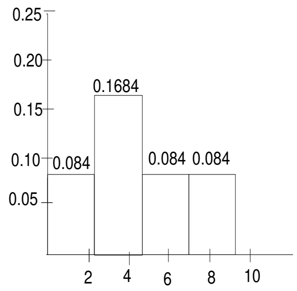

Figure 8: Histogram using Equal-Width-Interval Discretization

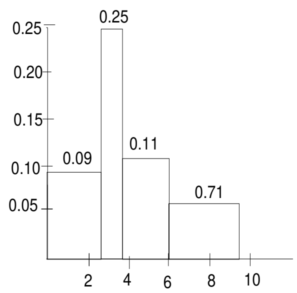

Figure 9: Histogram using Maximum Entropy Discretization.

3.1.2 Extracting Spatial Information

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 $(x^i_{min}, y^i_{min}, x^i_{max}, y^i_{max})$. 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 than 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}.

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 > SIZE then
     - Partition R into 4 subregions $\{R_j\}$, $j = 1, \cdots, 4$, subject to the SIZE constraint.
     - For each R_j do
       * Compute the value of D.
       * If D > 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 adjusting 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 $\{(c_1, r_1),(c_2, r_2), \cdots, (c_n, r_n)\}$ where c_i denotes a chosen color, and r_i is a list of cluster regions of color c_i. Each r_i is a list of four-tuples in the form of $\{(x^1_{min},y^1_{min},x^1_{max},y^1_{max}), \cdots, (x^n_{min},y^n_{min},x^n_{max},y^n_{max})\}$. Tuple $(x^i_{min},y^i_{min},x^i_{max}, y^i_{max})$ denotes a rectangular region with (x^i_{min},y^i_{min}) as the lower left-hand corner and (x^i_{max},y^i_{max}) as the upper right-hand corner.

3.2 The Retrieval Process

Having found the color-spatial information of the principle components, our next step is to use this information in the image retrieval process. A similarity measure is computed based on the computational models for color-spatial information.

3.2.1 Computational Models for Color-Spatial Information

Given the color-spatial information, let $C(i, k)$ denotes the ith 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:

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

where $\tilde{\max}\ x = \max\ \tilde{x}$ where $\tilde{x} \in \{\ x \mid x$ satisfies the following criteria: the proportion of x with respect to the total cluster area is greater than a pre-defined threshold. Obj_{g_1} is the set of representative object colors in image g_1, Obj_{g_2} is the set of representative object colors in image g_2, N_i and N_k is the number of clusters of color i and color k respectively, and $C_{g_1}(j,i) \cap C_{g_2}(m,k)$ 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 10.

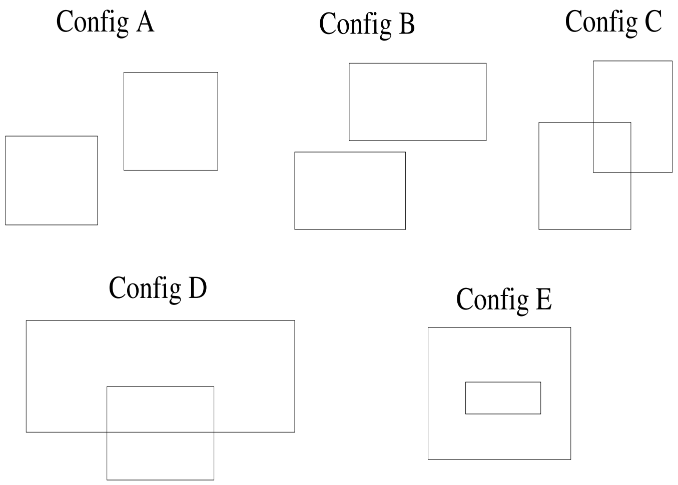

Figure 10: The Possible Configurations of Two Clusters.

In configuration A and configuration B, the two clusters do not overlap. These two cases correspond to the conditions: ($x_2 < x_3$ or $x_4 < x_1$), and ($y_2 < y_3$ 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 $\tilde{y}_1, \tilde{y}_2, \tilde{y}_3, \tilde{y}_4$ denotes the sorted lists of x and y coordinates respectively. The intersection region is given by

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

3.3 Making the Comparison Orientation-Independent

With this clustering representation, we can extend our similarity comparison to images with different sizes, different orientation, 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 is given below:

Scaling

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

Rotation anticlockwise about the origin

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

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 applying the appropriate matrix operation to model the desired transformation, two images with different sizes, orientation or position can be compared.

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 11. It consists of two main modules: the preprocessing module and the comparison module. Two databases are used. One for storing the 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 retrieving techniques: the color histogram technique and the principle-components technique. Each query image is tested using the two techniques. For completeness's sake, we now outline briefly the computational model used in the color histogram technique.

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.

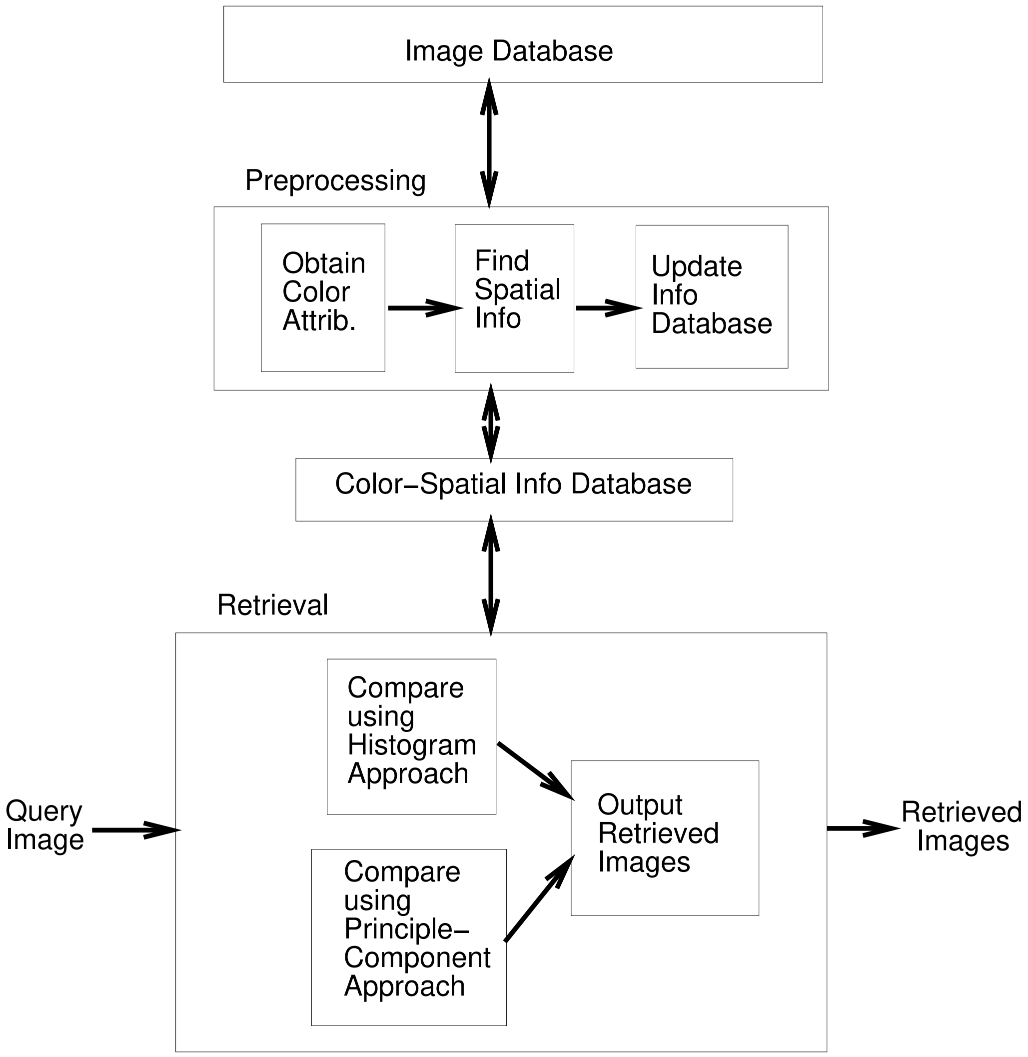

Figure 11: The Overall System Architecture.

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

$$SIM_{c1}(g_1, g_2) = \frac{\sum_{j \in S1_N}(1 - \frac{|NH(g_1,j) - NH(g_2,j)|}{max\{NH(g_1,j), NH(g_2,j)\}})}{N}$$

where$S1_N$ is the set of all non-zero color intensities of image g_1, and N is the total number of elements in $S1_N$. $SIM_{C1}(g_1, g_2)$ takes value between 0 and 1. If g_1 and g_2 are very similar to one another, $SIM_{C1}(g_1, g_2)$ will be very close to 1. If g_1 has little relationship to g_2, $SIM_{C1}(g_1, g_2)$ will be very close to 0.

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 desktop 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 principle-components technique.

Before presenting the results, let us first define the parameters used in the experiments. Recall that in the color selection process, a focus window is used to capture the principle components colors. In our experiments, the size of this window is fixed 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 principle-components approach using indirect matching criteria is 0.6.

Six queries are formulated to test the effectiveness of the retrieval techniques. For each query, three experiments are performed. In the first experiment, the principle-components 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.

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

Category	Relevant Images
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.

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

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

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

Table 3: Results of Image Retrieval Experiments

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)

$$\begin{aligned} \text{recall} &= \frac{\text{relevant retrieved}}{\text{all relevant}} = \frac{A}{A+C} \\ \text{precision} &= \frac{\text{relevant retrieved}}{\text{all retrieved}} = \frac{A}{A+B} \end{aligned}$$

Table 3 gives the summary of the results. From the table, we see that the principle-components 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 principle-components method. This is because the query category in Query2 is Painting. In this category, colors, 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 then the principle-components approach. However, as the recall level increases, the principle-components approach finds all the relevant images with better precision. Next, we compare the performance of the prinicple-components approach using the direct matching criteria versus the indirect matching criteria. In general, there is no significant difference in the performances between the two. In fact, the direct matching criteria seems to perform much better in Query4. This is to be expected because the Query4's category is Farming Field. Using direct matching criteria, the

Figure 12: Query4 Retrieval Samples

images in Figure 12 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 object-level image retrieval using the color and spatial information of the principle components in the image. Comparisons of the results using the color-histogram method and our proposed approach have been presented. Beside a better average precision for retrieval of images with similar objects, our proposed approach is more tolerant to noise in the image and it is flexible in dealing with 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] Irving Biederman, "Human image understanding: Recent research and a theory," *Computer Vision, Graphics, and Image Processing*, 32:29–73, 1985.

[4] Binaghi E., Gaglardi 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 derivatie," *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] B. M. Mehtre, M. S. Kankanhalli, A. D. Narasimhalu, and G. C. Man, "Color matching for image retrieval," *Pattern Recognition Letters*, 16:325–331, 1995.

[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] 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.