

A Novel Vector-Based Approach to Color Image Retrieval Using a Vector Angular-Based Distance Measure

D. Androutsos,^{*,1} K. N. Plataniotis,[†] and A. N. Venetsanopoulos^{*}

^{*}Digital Signal & Image Processing Lab., Department of Electrical and Computer Engineering, University of Toronto, Toronto, Ontario, Canada M5S 3G4; [†]School of Computer Science, Ryerson Polytechnic University, Toronto, Ontario, Canada M5B 2K3

E-mail: zeus@dsp.toronto.edu, anv@dsp.toronto.edu, kplatani@acs.ryerson.ca

Color is the characteristic which is most used for image indexing and retrieval. Due to its simplicity, the color histogram remains the most commonly used method for this task. However, the lack of good perceptual histogram similarity measures, the global color content of histograms, and the erroneous retrieval results due to gamma nonlinearity, call for improved methods. We present a new scheme which implements a recursive HSV-space segmentation technique to identify perceptually prominent color areas. The average color vector of these extracted areas are then used to build the image indices, requiring very little storage. Our retrieval is performed by implementing a combination distance measure, based on the vector angle between two vectors. Our system provides accurate retrieval results and high retrieval rate. It allows for queries based on single or multiple colors and, in addition, it allows for certain colors to be excluded in the query. This flexibility is due to our distance measure and the multidimensional query space in which the retrieval ranking of the database images is determined. Furthermore, our scheme proves to be very resistant to gamma nonlinearity providing robust retrieval results for a wide range of gamma nonlinearity values, which proves to be of great importance since, in general, the image acquisition source is unknown. © 1999 Academic Press

I. INTRODUCTION

Content-based image retrieval (CBIR) is a research area dedicated to the image database problem [1]. The number of image database systems which have recently been developed, and others that are currently under development [2–5], is strong evidence to this area's importance.

A key aspect of image databases is the creation of robust and efficient indices which are used for retrieval of image information. In particular, color remains the most important low-level feature which is used to build indices for database images. Specifically, the color histogram remains the most popular index, due primarily to its simplicity [6, 7].

However, using the color histogram for indexing has a number of drawbacks:

- histograms require quantization to reduce the dimensionality. A typical 24-bit color image generates a histogram with 2^{24} bins, which requires at least 2 Mbytes of storage space, depending on the resolution. However, with quantization comes loss of color information and there is no set rule as to how much quantization should be done.

- the color space which is being histogrammed can have a profound effect on the retrieval results and also governs the type and amount of quantization. For instance, for the RGB space, uniform quantization can be used since the distribution of each color plane is essentially uniform. However, in spaces such as the Munsell and LUV, uniform quantization may not suffice [6].

- color exclusion is difficult using histogram techniques. For example, a simple way that exclusion could be achieved would be to simply omit histograms which contain nonempty bins corresponding to the exclusion color. Unfortunately, it would be required to specify a priori which bins represent the exclusion colors. If the degree of quantization is also factored in, it is easy to see that determination of these bins is an ill-posed problem. If too few bins are selected, then a perceptually similar bin may be included in the similarity calculation. If too many bins are specified, then a bin corresponding to a dissimilar color may be omitted, which could possibly misclassify an otherwise valid image. A more desirable approach would be to allow certain colors to be excluded from a user-defined query right from the start, without requiring an additional level of analysis. In addition, a similarity measure should be used to determine if indexed colors match an exclusion color, and their level of similarity should affect the overall image similarity calculation accordingly.

- histograms can provide erroneous retrieval results in the presence of gamma nonlinearity. In general, an image database can contain images acquired from many unknown sources and can pass through a number of stages from the moment it is captured to the moment it is displayed. This poses a problem. For example, a scene can be captured on photographic film, transferred to paper and then scanned to digital format where it can be displayed on any computer monitor. These stages introduce a multiplicative nonlinearity due to the gamma nonlinearity of the various equipment [8]. For image retrieval this can cause very poor performance. It can cause false retrievals and render

¹ Corresponding author. URL: <http://www.dsp.toronto.edu>.

comparisons and similarity measures between pixel values, and ultimately images, erroneous [9, 10].

- the histogram captures *global* color activity; no spatial information is available. To include spatial information requires each image to be partitioned into n regions and a histogram built for each region [11], which consequently requires n times more storage.

In this paper we present a novel scheme for extracting, indexing, and retrieving color image data. We use color segmentation to extract regions of prominent and perceptually relevant color and use representative vectors from these extracted regions in the image indices. We do away with histogram indexing techniques and instead implement vector techniques. This way we end up with a much smaller index which does not have the over-completeness or *granularity* of a color histogram, yet performs better and robustly. Our similarity measure for retrieval is based on the angular distance between query color vectors and the indexed representative vectors.

The outline of the paper is as follows: Section II discusses our feature extraction method and how we build our image indices. Section III introduces our vector approach to retrieval and the vector angular-based measure which we implement. Section IV introduces and discusses what we call the *multidimensional distance space* and how we implement it for multiple color query and color exclusion. Section V discusses the problems with gamma nonlinearity and image retrieval and shows how our scheme proves more robust to gamma variation over other methods. Finally, Section VI concludes the paper.

II. FEATURE EXTRACTION AND INDEXING

Recently, some image retrieval systems have begun to move away from histogram techniques and begun to make use of segmentation to extract and index features [12–16]. Color image segmentation is an area which has received a lot of attention and research. Its popularity and effectiveness lies in the fact that the task of color segmentation is an inherent component of human visual processing. At the earliest stages of human vision, low-level processing naturally and automatically partitions a perceived scene, without any recourse to information regarding content or context. These extracted color features are then used in later stages of human vision to build objects which are then identified and classified by the brain.

To extract color features and build indices into our image database we take into consideration factors such as human color perception and recall. For example, as humans it is very difficult, if not impossible, for us to visually discern the difference between two very close *RGB* values, e.g., (255, 48, 32) and (254, 48, 32). Furthermore, if we were to describe the color content of an image, we would use terms such as *red*, *dark yellow*, or *bright green*, not *RGB* values. In addition, we tend to focus on and remember bright saturated regions and large color regions present in an image. Film manufacturers exploit

the former of these characteristics by increasing the saturation of their color film to make colors appear more vivid to better mimic our recall of a photographed scene [17].

When we wish to describe an image or to find a desired image, we essentially build a low-level model of the image in question in our mind and compare candidate images to this model. The color *granularity* provided by histogram indexing is, in most cases, not necessary, especially when the final observer is a human. Thus, it is natural to segment an image into regions of perceptually prominent color and retrieve candidate images based on the similarity to the color of each of these regions.

A. Segmentation

Our method of color indexing implements *recursive* HSV-space segmentation to extract regions within the image which contain perceptually similar color. We chose this space, due to its proven segmentation performance and for the fact that it allows for fast and efficient automation. It is not dependent on variables, such as seed pixels or a number of extracted colors, as in clustering and region growing techniques [18, 19], and this is of great significance if database population is to be independent of human intervention.

The HSV space [20] classifies similar colors under similar *hue* orientations. *Hue* is particularly important, since it represents color in a manner which is proven to imitate human color recognition and recall. The conversion from *RGB* to *HSV* is performed with the equations

$$H_1 = \cos^{-1} \left(\frac{\frac{1}{2}[(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right), \quad (1)$$

where $H = H_1$ if $B \leq G$; otherwise $H = 360^\circ - H_1$;

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)}; \quad (2)$$

$$V = \frac{\max(R, G, B)}{255}, \quad (3)$$

where R , G , and B are the red, green, and blue component values which exist in the range [0, 255].

In our method, we threshold the peaks of the *hue* histogram, which is known to contain most of the color information, while also taking into account *saturation* and *value* information.

The first step is to build a *hue* histogram for all the *bright chromatic* pixels. We have found experimentally that these tend to be colors that have *value* > 75% and *saturation* \geq 20%. Once the pixels which satisfy this criterion are identified, the *hue* histogram is built and thresholded into m bright colors, where m is an image-dependent quantity determined by the number of peaks which the *hue* histogram exhibits.

From the remaining image pixels, *saturation* and *value* are used to determine which regions of the image are achromatic.

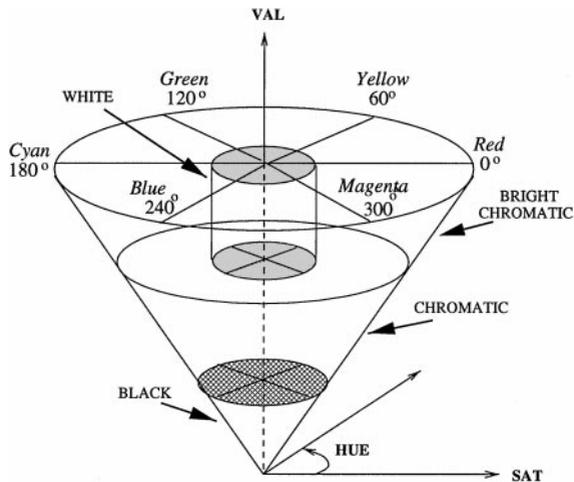


FIG. 1. HSV cone depicting BLACK, WHITE, BRIGHT CHROMATIC, and CHROMATIC regions.

Specifically, it has been found, in the literature and experimentally [21, 22] that colors with $value < 25\%$ can be classified as **black**, i.e. at the bottom of the HSV cone, and that colors with $saturation < 20\%$ and $value > 75\%$ can be classified as **white**, as shown in Fig. 1.

All remaining pixels fall in the *chromatic* region of the HSV cone. However, there may be a wide range of *saturation* values. To account for this, we calculate the *saturation* histogram of all these remaining chromatic pixels. The *saturation* histogram is, in general, multimodal and we take this fact into account. Many segmentation researchers have classified the *saturation* histogram to be bimodal; however, this is not true and we have found that more accurate color segmentation can be obtained by taking into account its multimodal nature. Assume that a *saturation* histogram exhibits p peaks, we threshold each of these peaks and calculate the *hue* histogram for the pixels contained in each given peak. Each resulting *hue* histogram, which exhibits n peaks, is thresholded accordingly to obtain n colors. The process is then repeated again for each of the p *saturation* peaks. The entire process is shown in the flowchart in Fig. 2.

Thus for each image, the segmentation algorithm extracts c colors, where $c = \sum_p n_p + m$, which is clearly an image-dependent quantity. Finally, we calculate the average color of each of the c colors and use that *RGB* value as each region's representative vector. The reason for using *RGB* vectors is primarily due to the fact that there is no established method for similarity calculation in the HSV space. In addition, as will be seen in the next section, by using *RGB* vectors we are able to exploit powerful angular-based distance measures for similarity calculation.

Figure 3 shows a typical image, *jellybeans*, and the segmented result after our recursive segmentation procedure, where 12 colors were identified and extracted, and each region was filled with its average color. As can be seen, the segmented result pro-

vides an accurate low-level representation of the color content of the original image with all dominant and bright colors correctly identified.

The above segmentation technique was performed on our test database of 2000 24-bit images of 512×512 resolution of general image content, including natural scenes, people, architecture, animals, and plants. Each image index was built using the c extracted color vectors, along with the percentage of each extracted color present in the image and the number of regions which contain each color. A statistical analysis of our entire 2000 image database revealed that the average number of colors

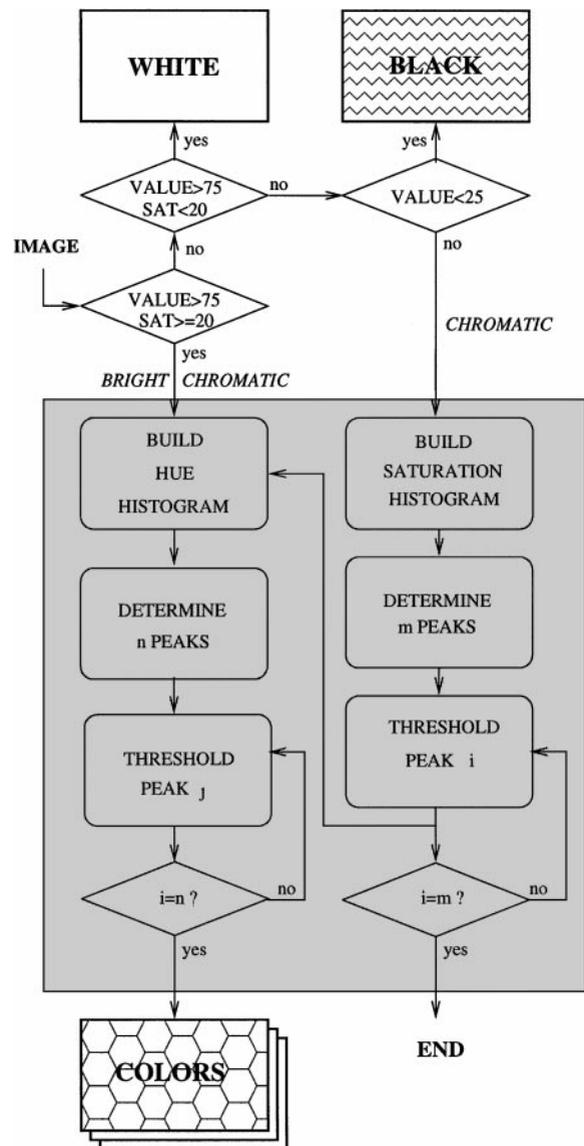


FIG. 2. Flowchart of segmentation procedure. Each image is examined to classify the pixels into one of the four categories: BLACK, WHITE, BRIGHT CHROMATIC, and CHROMATIC regions. For the latter two, *hue* and *saturation* histograms are built and the corresponding peaks are thresholded to segment the colors.

extracted was 4.68, including black and white, and the maximum and minimum number of extracted colors were 13 and 1. This is a surprisingly small number of colors; however, as we will see, retrieval proves to be very effective.

It is also important to note that, by virtue of using color segmentation, we can also incorporate spatial color information into our indices quite easily and efficiently, without having to dramatically increase the index size. However, we do not address spatial color retrieval in this paper.

III. RETRIEVAL

A. Vector Approach

By virtue of the fact that our color indices are three-dimensional vectors which span the *RGB* space, we have at our disposal a number of vector distance measures that can be implemented for retrieval. However, studies have shown that measures based on the angle of a color vector produce perceptually accurate retrieval results in the *RGB* domain [23]. Furthermore, angular measures are *chromaticity*-based, which means that they operate primarily on the orientation of the color vector in the *RGB* space and, therefore, are more resistant to intensity changes.

As further evidence to the validity of angular-based measures, we find that they exhibit excellent performance in the area of image filtering [24–26]. At first, retrieval and filtering seem unrelated. However, the fact is that both use distance measures to determine candidacy. In particular, *order statistics* filtering implements distance measures to group similar vectors together and discard outliers, whereas retrieval *ranks* the similarity between candidates.

Figure 4 depicts the color variation of a given color vector in the *RGB* space as the angle is varied at eight points around the central vector. From the figure we can see that, as the angle increases further away from the central vector, the perceived color also changes. However, for the small angles of 0.05 *rad* and 0.10 *rad* the color is perceptually the same as the central color; thus, a small neighborhood around a given vector in the *RGB* space contains colors that can be considered equivalent.

B. Distance Measure

In our system we implement a distance measure based on the *angular distance* between two vectors. Specifically it is a *combination* distance measure which is composed of an angle

and magnitude component [27]:

$$\delta(\mathbf{x}_i, \mathbf{x}_j) = 1 - \underbrace{\left[1 - \frac{2}{\pi} \cos^{-1} \left(\frac{\mathbf{x}_i \cdot \mathbf{x}_j}{|\mathbf{x}_i| |\mathbf{x}_j|} \right) \right]}_{\text{angle}} \underbrace{\left[1 - \frac{|\mathbf{x}_i - \mathbf{x}_j|}{\sqrt{3 \cdot 255^2}} \right]}_{\text{magnitude}}, \quad (4)$$

where \mathbf{x}_i and \mathbf{x}_j are three-dimensional color vectors. Since we deal with *RGB* vectors, we are constrained to one quadrant of the Cartesian space. Thus, the normalization factor of $2/\pi$ in the *angle* portion is attributed to the fact that the maximum angle which can possibly be attained is $\pi/2$. Also, the $\sqrt{3 \cdot 255^2}$ normalization factor, in the *magnitude* part of (5), is due to the fact that the maximum difference vector which can exist is (255, 255, 255) and its magnitude is $\sqrt{3 \cdot 255^2}$. Both normalization factors contribute so that δ takes on possible values in the range [0, 1].

This distance measure takes into consideration both the angle between two vectors and the magnitude of the vector difference. However, when two vectors under consideration are collinear, only magnitude difference is used, which is required since two vectors can be collinear but perceptually quite different due to magnitude differences.

For comparison purposes, we have also investigated a number of other common vector distance measures, specifically:

- the *angular distance* between two vectors,

$$\theta(i, j) = 1 - \frac{2}{\pi} \cos^{-1} \left(\frac{\mathbf{x}_i \cdot \mathbf{x}_j}{|\mathbf{x}_i| |\mathbf{x}_j|} \right), \quad (5)$$

which is the *angle* component of (5) above;

- the generalized *Minkowski* metric (L_M norm) [19],

$$d_M(i, j) = \left(\sum_{k=1}^p |x_i^k - x_j^k|^M \right)^{\frac{1}{M}}, \quad (6)$$

where p is the dimension of the vector \mathbf{x}_i and x_i^k , is the k th element of \mathbf{x}_i . Three special cases of the L_M metric are of particular interest, namely, $M = 1, 2, \infty$;

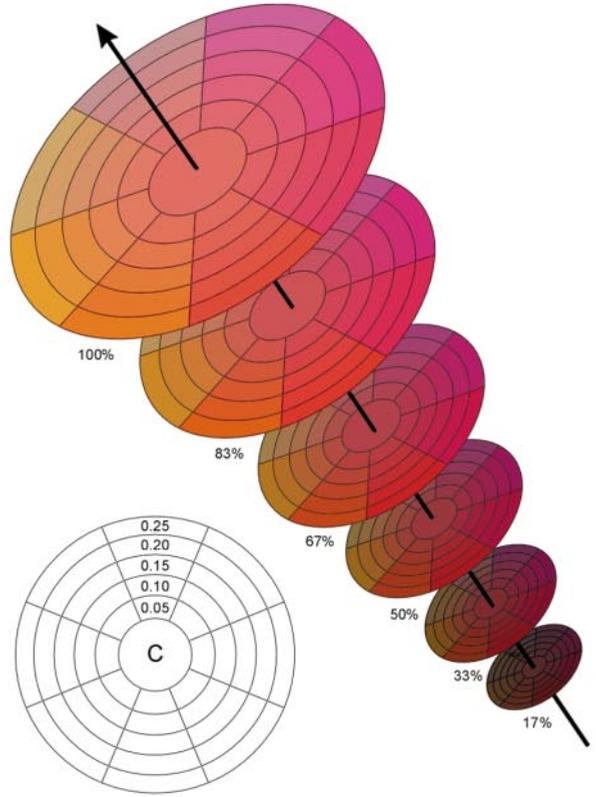
- and the *Canberra* distance defined as

$$d_c(i, j) = \sum_{i=1}^p \frac{|x_i^k - x_j^k|}{|x_i^k + x_j^k|}, \quad (7)$$

FIG. 3. True-color 24-bit *jellybean* image and its segmented image of only 12 colors.

FIG. 4. Six swatches depicting color changes around a central color vector \mathbf{C} , with *RGB* values of (187, 83, 78), due to changes in the angular distance with respect to that vector. The angle is calculated for angles of 0.05, 0.1, 0.15, 0.20, 0.25, and 0.5 *rad* for eight points around \mathbf{C} . Each swatch shows the effect at a different point along \mathbf{C} , specifically at 125%, 100%, 75%, 50%, and 25% of $|\mathbf{C}|$.

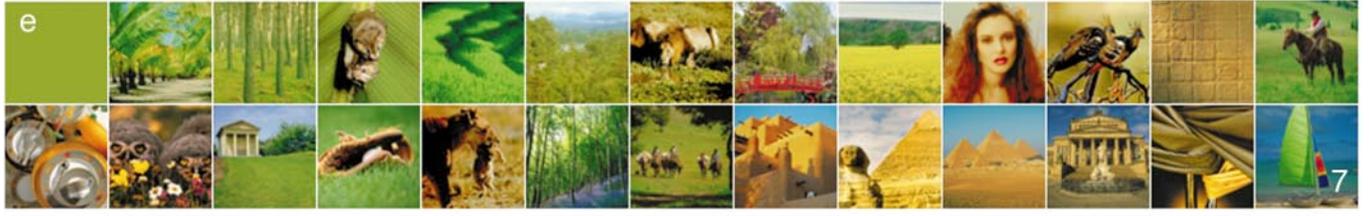
FIG. 6. *Bat* image and the color *sea green*.



4



6



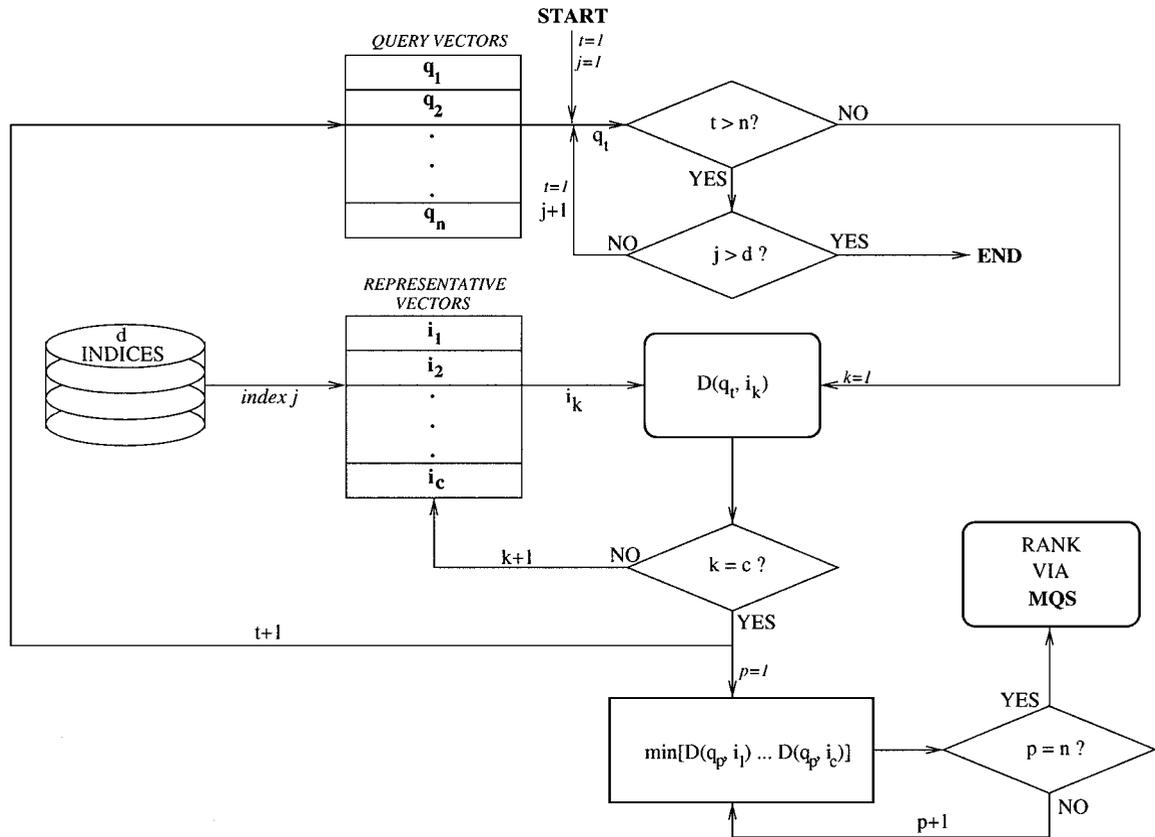


FIG. 5. Flowchart of the query procedure. A distance D is calculated between each pairing of the c representative vectors of each image index i , with each of the n query colors. For each of the d database indices, this produces a set of $c \times n$ distance values. For each index, the n smallest of these values are passed on to the MQS to determine the overall ranking of the image.

where p is the dimension of the vector \mathbf{x}_i and x_i^k is the k th element of \mathbf{x}_i . The *Canberra* metric applies only to nonnegative multivariate data, which is the case when color vectors described in the RGB reference system are considered.

C. Single Color Query

Our first test of the system deals with the single color query, i.e. where only one color is specified. For each image index, the system takes a specified query color and calculates the given distance measure to *each* of the n representative vectors and retains the minimum. The result is the closest match of the set of n representative color vectors, of a given database image, to the query color. The process is repeated for every database index and the entire set of distances are then ranked to retain the

top 25 images. Figure 5 depicts the steps involved in the query process.

For our results, we chose to look for images which contained at least 50% of the query color *sea green* with *RGB* values (130, 164, 53), which is shown in Fig. 6, along with the database image *bat* from which it was extracted. Figure 7 shows the retrieval result using our *combination* distance, along with the retrieval result of the other vector distance measures discussed in Section III.B. In addition, we have also included results using histogram techniques to compare against our technique in Fig. 8; specifically, we tested RGB and HSV-space histogram techniques. Unavoidably, quantization was performed in the two color spaces. Due to the relatively uniform distribution of the RGB bands, we chose eight uniform quantization bins for each of the RGB bands. For the HSV color space histogram, we

FIG. 7. Retrieval result using (a) *combination* distance, (b) *angular distance*, (c) L_1 norm, (d) L_2 norm, and (e) L_∞ norm. Top left image of each result is the query color *sea green*.

FIG. 8. Retrieval result using (a) RGB histograms with (8, 8, 8) quantization bins and (b) HSV histograms with (12, 5, 5) quantization bins. Top left image is the query color *sea green*.

FIG. 9. Query set Q obtained from 25 test subjects who were asked to find images which contained at least 50% of the query color *sea green* (top left image).

took the statistical nature of the HSV space into consideration, along with human sensitivity to *hue* and *saturation*, and uniformly quantized the *hue*, *saturation*, and *value* histograms into 12, 5, 5 bins, respectively [6]. The similarity metric that we implemented for the actual retrieval was the *histogram intersection* [28].

Figure 7a depicts the top 25 retrieval results for our *combination* measure. Figures 7b–d show the results for the other vector distance measures discussed. In addition, Fig. 8 shows the results for the RGB and HSV-space histogram retrieval techniques. In each of these figures, the *top left* image is the *sea green* query color and the similarity ranking is from highest to lowest from left to right, top to bottom.

All methods returned images that contained colors similar to the query color; however, the *combination* measure and the *angle* measure returned more images which were perceptually more accurate. This was established by comparing the retrieved results with a *query set Q* which contains the images which most humans would consider to fit the given query. In our case, The query set *Q* was obtained from 25 volunteers who were asked to manually search through our 2000-image database and list the images which were considered to contain at least 50% of the query color (*sea green*). The results were then tabulated and the top 25 images which were chosen the most often comprised the query set *Q*, depicted in Fig. 9.

Comparing Figs. 7 and 8 with Fig. 9, we see that our *combination* distance retrieves more images that belong to *Q* than the other investigated measures and techniques. Specifically, our measure returned 16 images from the query set *Q*. This is of utmost benefit and importance since the perceived color content of an image is, in most cases, of greater significance than the actual pixel values. In addition, the *combination* measure returned less images that were erroneous, i.e., that contained colors that were totally irrelevant with respect to the query color.

Quantitative performance was evaluated by calculating the retrieval rate, defined as [11]:

$$R_{i,j} = \frac{N_j}{N_i} \times 100, \tag{8}$$

where N_i are the total images in a given query set *Q* (i.e., all images in the database which match the query) and N_j are the number of images which appear in the top N_i retrieval positions which are part of *Q*. Table 1 lists the retrieval rates for the above mentioned distance measures. Clearly, it can be seen that the angular-based measures exhibit much higher retrieval rate,

TABLE 1
Retrieval Rate for Six Different Vector Distance Measures and Two Histogram Techniques

Measure	$R_{i,j}$
L_1	44
L_2	48
L_∞	36
Canberra	40
Angle	60
Combo	64
RGB _{8,8,8}	48
HSV _{12,5,5}	28

and in particular our *combination* measure provided the highest retrieval rate. It even surpassed the two histogram indexing techniques.

IV. MULTIDIMENSIONAL QUERY SPACE

During the query process, for each user-specified query color, a distance is calculated, using (5), to *each* representative vector in a given database index. For multiple color queries, a distance to each representative vector is calculated for each query color. We take the minimum of these distances and form a *multidimensional query distance vector D*:

$$\mathbf{D}(d_1, \dots, d_n) = (\min(\delta(\mathbf{q}_1, \mathbf{i}_1), \dots, \delta(\mathbf{q}_1, \mathbf{i}_c)), \dots, \min(\delta(\mathbf{q}_n, \mathbf{i}_1), \dots, \delta(\mathbf{q}_n, \mathbf{i}_c))), \tag{9}$$

where \mathbf{q}_n are the *n* query colors and \mathbf{i}_c are the *c* indexed representative colors of each database image.

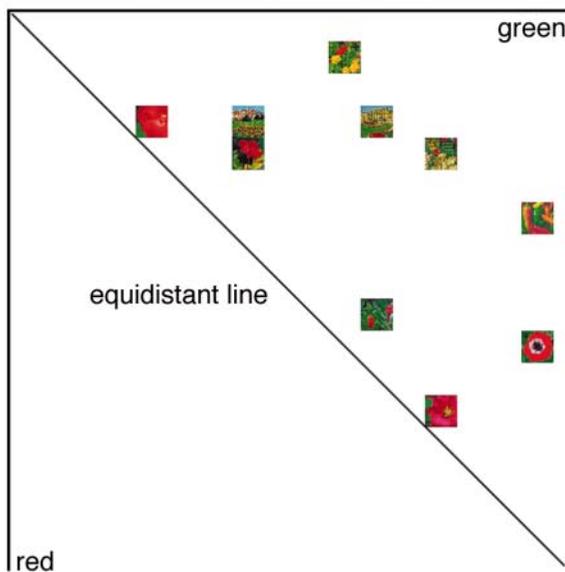
For example, let us assume that a query consists of two query colors, \mathbf{q}_1 and \mathbf{q}_2 , and a given index contains representative color vectors $\mathbf{i}_1, \mathbf{i}_2$ and \mathbf{i}_3 . The minimum distance d_1 between \mathbf{q}_1 and $\mathbf{i}_{1\dots3}$ is taken, along with the minimum distance d_2 between \mathbf{q}_2 and $\mathbf{i}_{1\dots3}$, to build $\mathbf{D}(d_1, d_2)$. The process is repeated for each database index and we obtain the result with *N* distance vectors \mathbf{D} , where *N* is the size of the image database (2000 in our case).²

² Consequently, this is a linear operation and computation time can grow as the database size increases. To help alleviate this, when each representative vector is created it is possible to perform ordering or clustering to group similar representative vectors together and compare query vectors *only* with representative vectors that are in close proximity to each other. One aspect of our current research focuses on this issue.

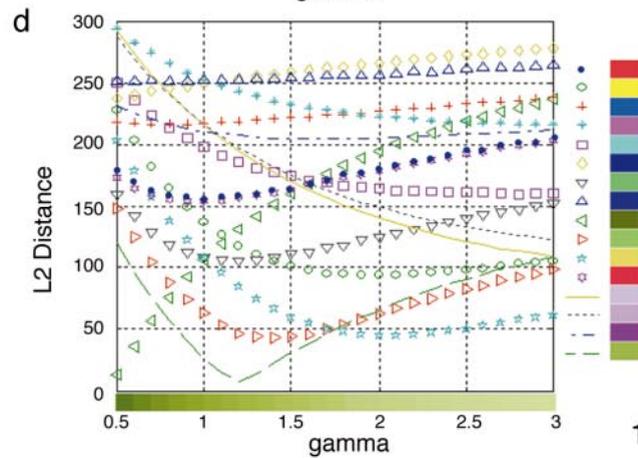
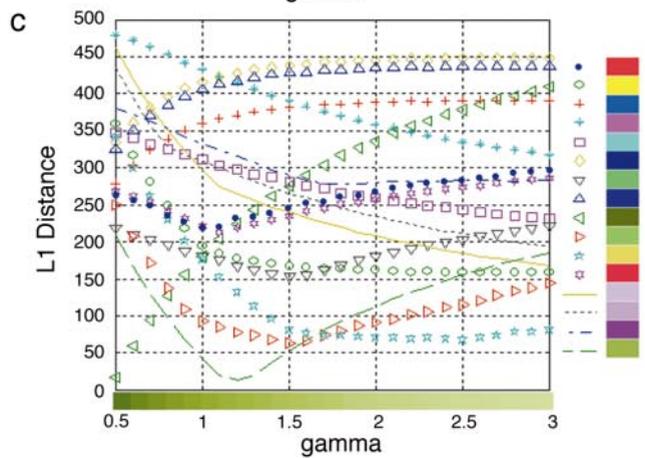
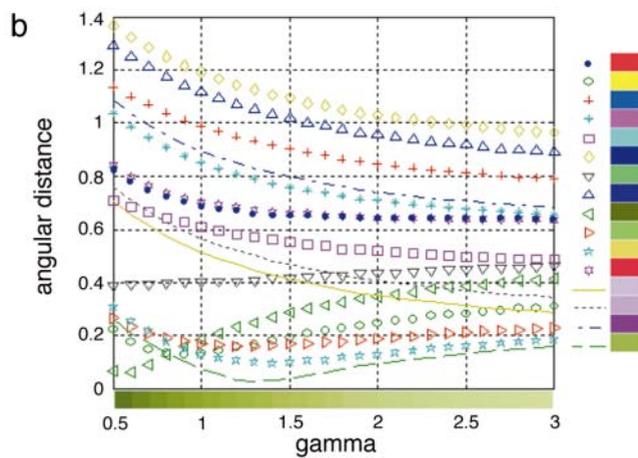
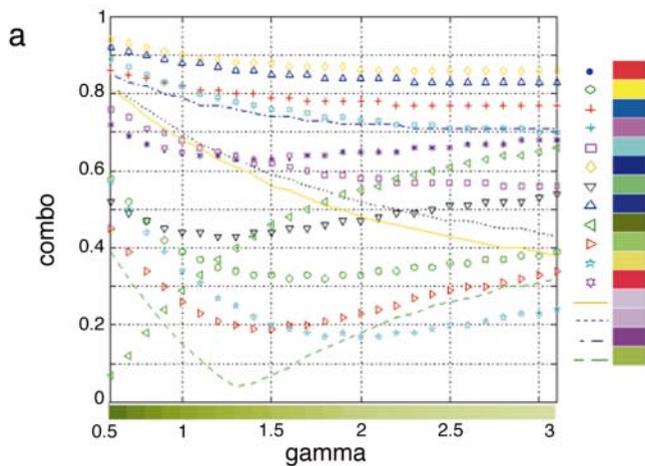
FIG. 10. Query result for (a) images with *red* and *green*, (b) images with *red* and *green* and excluding *yellow*.

FIG. 11. Spatial representation of the query result for images with *red* and *green*. The axes represent the calculated distance δ for each image to the corresponding color. The origin at the top left is the closest match to both query colors.

FIG. 13. Plot of the (a) *combination* distance, (b) *angular* distance, (c) L_1 distance, and (d) L_2 distance of 16 colors against a color L whose gamma is varied from 0.5 to 3.0.



11



13

This set of \mathbf{D} vectors span what we call the *multidimensional query space* (MQS) [29]. For the case of two query colors the space is two-dimensional, for three colors it is three-dimensional, etc. Each database image exists at a point in this space and its location determines its retrieval ranking for the given query. The key to this ranking lies in the origin of the MQS and the *equidistant line*, where all component values of \mathbf{D} are equal (see Fig. 12a). The database image that is the closest match to *all* the given query colors q_1, q_2, \dots, q_n is the one which is closest to the origin of the MQS. This implies that the distance vector \mathbf{D} that is most centrally located, i.e., is collinear with the *equidistant line* of the MQS and at the same time has the smallest magnitude, corresponds to the image which contains the best match to *all* the query colors.

Thus, we need to rank the retrieval results based on the magnitude of \mathbf{D} and the angle, $\angle \mathbf{D}$, that it makes with the *equidistant line*. To do this we combine the two values using a weighted sum

$$\mathcal{R} = w_1|\mathbf{D}| + w_2\angle \mathbf{D}, \tag{10}$$

where lower rank values \mathcal{R} imply images with a closer match to both the query colors. The weights w_1 and w_2 can be adjusted to control which of the two parameters, i.e. magnitude or angle, are to dominate. We have found that values of $w_1 = 0.8$ and $w_2 = 0.2$ give the most robust results. This is to be expected since collinearity with the equidistant line does not necessarily imply a match with *any* query color. It implies that each query color is equally close to the indexed colors. However, $|\mathbf{D}| \rightarrow 0$ implies closer matches to one or more colors. Thus, a greater emphasis must be placed on the magnitude component.

A. Multiple Color Query

Figure 10a depicts a user query for two colors. Specifically, it was desired to find images with at least 10% of the RGB colors 26, 153, 33 (green) and 200, 7, 25 (red). Clearly, the top 10 results displayed exhibit colors with strong similarity to the two query colors.

Figure 11 shows a spatial representation of the retrieval results for the same query colors. Here we can clearly see how close the retrieved images are to each query color and how the vector representation of the two distances determines the retrieved image ranking. Clearly, the origin represents the best match to both colors. Images which lie on the equidistant line have equal distance to each query color; however, these images must also be close to the origin to be *similar* matches to the query colors. This reasoning is easily extended to a higher dimension (i.e., a greater number of query colors) with the origin remaining the best retrieval result.

B. Color Exclusion

Our proposed vector approach provides a framework which easily accepts exclusion in the query process. It allows for image

queries containing any number of colors to be excluded, in addition to including colors in the query results. From the discussion in Section III.A above, we are interested in distance vectors \mathbf{D} which are collinear with the equidistant line and which have small magnitude, i.e., close to the MQS origin. The exclusion of a certain color should affect \mathbf{D} accordingly and its relation to the equidistant line and the origin. For example, if it is found that an image contains an indexed color which is close to an exclusion color, the distance between the two can be used to either pull (or push) \mathbf{D} closer (or further) to the ideal and accordingly affect the retrieval ranking of the given image.

To this end, we determine the minimum distances of each exclusion color with the indexed representative colors, using (5), to quantify how close the indexed colors are to the exclusion colors,

$$\mathbf{X}(x_1, \dots, x_n) = (\min(\delta(\xi_1, \mathbf{i}_1), \dots, \delta(\xi_1, \mathbf{i}_c)), \dots, \min(\delta(\xi_n, \mathbf{i}_1), \dots, \delta(\xi_n, \mathbf{i}_c))), \tag{11}$$

where ξ_n are the n exclusion colors and \mathbf{i}_c are the c indexed representative colors of each database image. Equation (11) quantifies how *close* any indexed colors are to the exclusion colors. Thus, since x_n depicts similarity, a simple transformation of $1 - x_n$ depicts dissimilarity. Thus, we can apply this transform to each of the components of \mathbf{X} and then merge this with \mathbf{D} to give the overall multidimensional vector

$$\Delta = [\mathbf{D} \quad \mathbf{I} - \mathbf{X}], \tag{12}$$

where \mathbf{I} is a vector of size n with all entries of value 1. The dimensionality of Δ is equal to the *number of query colors + number of exclusion colors*.

The final retrieval rankings are then determined from $|\Delta|$ and the angle which Δ in (12) makes with the equidistant line of the *query colors*. Figure 12b graphically depicts the notion of

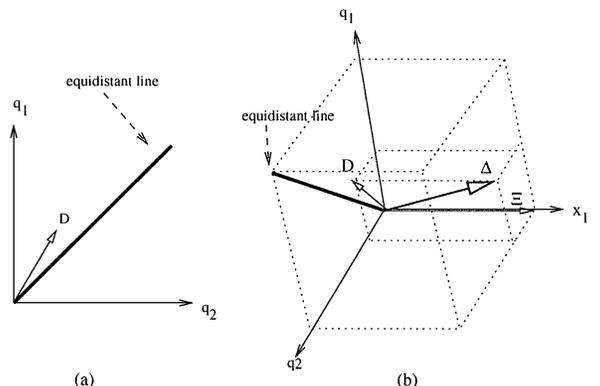


FIG. 12. (a) Vector representation of two query colors q_1 and q_2 , their multidimensional distance vector \mathbf{D} , and the corresponding equidistant line. (b) The same two query colors, one exclusion color x_1 , and the resulting multidimensional distance vector Δ .

color exclusion and shows how Δ affects \mathbf{D} and its relation to the origin and equidistant line.

Figure 10b depicts the query result when at least 10% of the *RGB* colors 26, 153, 33 (green) and 200, 7, 25 (red) (same colors as in Section IV.A), were desired and the color 255, 240, 20 (yellow) was excluded. Clearly, images which contained colors close to *yellow* were removed from the top ranking results, as compared to the Fig. 10a, where *yellow* was not excluded.

V. GAMMA

As mentioned in Section I, images are usually acquired from many unknown sources and can pass through a number of stages from the moment they are captured to the moment they are displayed. These stages introduce a multiplicative nonlinearity known as gamma nonlinearity [8] which can make two spectrally identical colors appear quite different and exhibit a low similarity measure. This can have a profound effect on retrieval results, especially when using histogram techniques for indexing and retrieval [9].

Thus, for good image retrieval performance a common image representation space would be ideal. However, most current image formats do not provide important color information, such as gamma, to convert a given image into such a device independent space. Until such a standard is proposed and accepted, techniques have to be created which deal with gamma. We have found that our vector approach, along with our distance measure, exhibits good performance in the presence of gamma nonlinearity.

A. Gamma Resistance

The ideal scenario would be to find a scheme which would be unaffected by gamma nonlinearity. In other words, we would like to find a measure that retains the same retrieval ranking of candidate images, based on color, for a wide range of gamma changes.

To this end, we initially tested our system for gamma resistance on a set of solid colors. Specifically, we used the *sea green* query color and varied the gamma from 0.5 to 3.0 by utilizing the gamma power function

$$y = x^\gamma, \quad (13)$$

where x is the input value and y is the output value. Equation (13) was applied to each of the *RGB* planes uniformly for each gamma value.

These gamma-varied *RGB* values were then used to calculate similarity to a set of 16 randomly selected colors using the angular distance measure. In addition, the *angular* L_1 and L_2 measures were also tested. The results are plotted in Figs. 13a–c, along with a legend depicting the 16 colors and the gamma-varied color.

Each curve in each graph depicts how the distance of the query color to a given color changes as the gamma is varied. The set of curves together can be interpreted as the retrieval order of

the 16 colors at a given gamma value (i.e., lower distance value implies a closer match and, thus, a higher retrieval ranking).

Clearly, the L_1 and L_2 measures give much more erratic behavior than the *combination* and *angular* measures. Furthermore, there are substantially less crossover points evident with the angular measure. These points correspond to a change in the similarity ranking of a given color or image in a database. The *combination* and angular measure plots in Figs. 13a and b exhibit much smoother behavior and low variation and crossovers, resulting in a retrieval ranking order that remains much more stable across the gamma scale, which indicates that image retrieval using angular-based measures should follow the same pattern.

B. Gamma and Retrieval

For further evidence as to the gamma resistance of our system, we tested image retrieval under gamma variation on our entire test database. Each image in our database was indexed using our proposed indexing scheme at varying gamma nonlinearity values of 0.8 to 2.6 at steps of 0.2. For each set of indices created at each of the 10 gamma levels, retrieval was performed using the query color *sea green*, which was obtained from the *bat* image. The system, once again was set to retrieve all the images which had at least 50% of the query color.

In Table 2, the number of appearances that each image retrieved in the top 25 positions made throughout the 10 gamma levels are listed. In other words, the retrieval of a certain image was tracked across the retrieval results of the varying gamma levels. For each gamma level, the top 25 retrieval results were displayed simultaneously. Each image was tracked to see if it remained among the top 25 retrieved images, as the gamma value was varied, and the number of appearances that each image made was tabulated. The entries with an asterisk indicate the number of appearances the *bat* image made, i.e., the image from which the query color was extracted. For the *combination* and *angular* distance measure there were 21 images which appeared 6 or more times. Three images appeared 10 times for the

TABLE 2
Comparison of the Number of Appearances Each Image Makes in Each Retrieval Result at 10 Different Gamma Levels

Appearances	Combo	Angle	L_2	L_1	RGB (8, 8, 8)	HSV (12, 5, 5)
1	10	12	27	27	20	10
2	8	5	8	14	20	12
3	3	3	14	14*	12	6
4	5	8	12*	11	9	3
5	4	3	9	5	4	6
6	5	6	7	8	2	3*
7	4	2	2	4	2	1
8	5	2	2	1	3	1
9	4*	4	0	0	2*	2
10	3	7*	0	0	3	10
Unique	54	52	81	84	77	54

combination measure and seven images appeared 10 times for the *angular* measure. However, it was found that many of the images which the *angular* measure retrieved 10 times were not accurate retrievals and only three of them were members of the query set Q , whereas with the *combination* measure, all four images were members of Q .

In addition, a low number of *unique* images factored into the retrieval results. By *unique* images, we refer to images which appeared *only once* across the varying gamma level retrieval results. Thus, the ideal retrieval result over all the gamma values, would be 25 *unique* images, i.e., the same 25 images retrieved consistently amongst the top 25 retrieval results, at each of the 10 gamma levels. The worst case is 25 different images retrieved at each gamma level, i.e. 250 *unique* images. The *angular* measure had the least *unique* images; however, as mentioned above, many retrieved images were inaccurate retrievals.

For comparison, we also investigated other distance measures, namely the L_1 and L_2 measures, and we have also included results using histogram techniques. Specifically, we varied the gamma nonlinearity value from 0.8 to 2.6 in steps of 0.2, as above, and built an RGB and HSV histogram for each database image at each gamma level, as was done in Section III.C.

These methods provided results which are not nearly as robust to gamma changes as our proposed method and resulted in a high number of *unique* images and much fewer appearances higher than six. The only exception is the HSV histogram, which seems to perform well in the presence of gamma nonlinearity. However, if we look at Fig. 14, where retrieval rate is plotted against gamma, we see that the HSV-histogram technique exhibits the lowest retrieval rates across the gamma values. Furthermore, we find that the highest retrieval rates are obtained by the angular measures, with the widest peak exhibited by the *combination* distance measure. Thus, the *combination* measure provides the highest retrieval rates over a wider range of gamma values, over all other techniques, while retaining a low number of *unique* images.

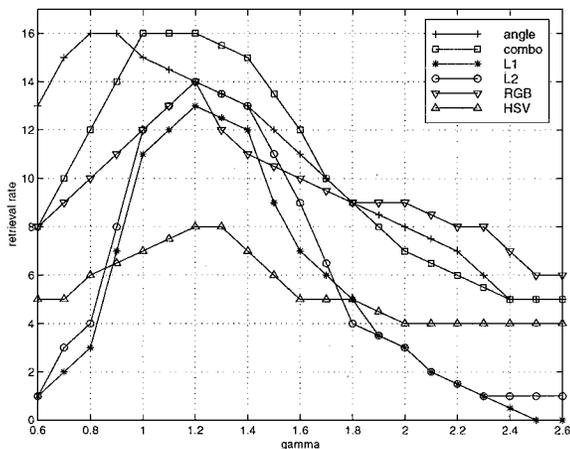


FIG. 14. Plot depicting the effect of gamma nonlinearity on the retrieval rate.

This can be attributed to the fact that the *combination* measure takes both angle and magnitude difference into consideration. Since gamma nonlinearity primarily affects intensity, angular differences remain relatively unaffected, which explains why the *angular* measure results in a low number of *unique* images. However, intensity difference is still important for optimal color similarity, which explains why the *combination* distance also has a low number of *unique* images, but at the same time exhibits much better retrieval rates.

VI. CONCLUSIONS

In this paper we present a new scheme for color image indexing and retrieval. We perform a recursive HSV-space segmentation technique to identify perceptually prominent color areas and use the average color vector of these areas as image indices in the database. Due to the vector nature of our system, we implement a *combination* distance measure, based on the vector angular distance, in the retrieval process. The query mechanism proves very flexible, allowing single and multiple color queries via a *multidimensional query space*. In addition, color *exclusion* can be implemented where certain colors can be specified to *not* be present in the query results. We have tested our system and our distance measure against other common vector distance measures and also against popular histogram indexing schemes and found that it outperforms them all. The retrieval results exhibit the highest retrieval rate of all the measures and methods investigated. Furthermore, the results comply much more closely with results which our set of 25 human observers considered correct. Last, we have found that our system provides more stable and robust results across a wide range of varying gamma nonlinearity values, which is important when dealing with general image databases which contain images from unknown sources.

REFERENCES

1. V. N. Gudivada and V. V. Raghavan, Content-based image retrieval systems, *Computer* **28**(9), 1995.
2. W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin, The QBIC project: Querying images by content using color, texture and shape, in *Storage and Retrieval for Image and Video Databases*, SPIE-1908, San Jose, 1993.
3. J. R. Smith and S.-F. Chang, VisualSEEK: A fully automated content-based image query system, in *ACM Multimedia 96*, Boston, November 1996.
4. A. Petland, R. W. Picard, and S. Sclaroff, Photobook: Tools for content-based manipulation of image databases, in *Storage and Retrieval for Image and Video Databases II*, SPIE-2185, San Jose, February 1994.
5. I. K. Sethi, I. Coman, B. Day, F. Jiang, D. Li, J. Segovia-Juarez, G. Wei, and B. You, Color-WISE: a system for image similarity retrieval using color, in *Storage and Retrieval for Image and Video Databases VI*, San Jose, 1998, pp. 140–149.
6. X. Wan and C.-C. Jay Kuo, Color distribution analysis and quantization for image retrieval, in *Storage and Retrieval for Image and Video Databases IV*, SPIE-2670, 1995, pp. 8–16.

7. M. Stricker and M. Orengo, Similarity of color images, in *Storage and Retrieval for Image and Video Databases III*, SPIE-2420, 1995, pp. 381–392.
8. C. A. Poynton, *A Technical Introduction to Digital Video*, Wiley, New York, 1996.
9. G. Lu, On image retrieval based on color, in *Storage and Retrieval for Image and Video Databases IV*, SPIE-2670, San Jose, 1996.
10. G. Lu, *Communication and Computing for Distributed Multimedia*, Artech House, Boston, 1996.
11. H. Zhang, Y. Gong, and S. W. Smoliar, Image retrieval based on color features: an evaluation study, in *Digital Image Storage and Archiving Systems*, SPIE-2606, 1995, pp. 212–220.
12. W. Y. Ma and Y. Deng, and B. S. Manjunath, Tools for texture/color based search of images, in *Human Vision and Electronic Imaging II*, SPIE-3016, 1997, pp. 496–505.
13. A. D. Bimbo, M. Mugnaini, P. Pala, F. Turco, and L. Verzuoli, Image retrieval by color regions, in *International Conference on Image Analysis and Processing*, Florence, 1997, pp. 180–187.
14. C. Carson, S. Belongie, H. Greenspan and J. Malik, Region-based image querying, in *CVPR '97 Workshop on Content-Based Access of Image and Video Libraries*, Puerto Rico, 1997.
15. S. Belongie, C. Carson, H. Greenspan and J. Malik, Color- and texture-based image segmentation using the expectation-maximization algorithm and its application to content-based image retrieval, in *ICCV '98, Mumbai, India*, 1998.
16. N. R. Howe, Percentile blobs for smage similarity, in *IEEE Workshop on Content-Based Access of Image and Video Databases*, Bombay, India, January 1998.
17. S. J. Sangwine and R. E. N. Horne, Editors, *The Colour Image Processing Handbook*, Chapman & Hall, London, 1998.
18. M. Celenk, A color clustering technique for image segmentation, *Comput. Vision Graphics Image Process.* **52**, 1990, 145–170..
19. R. O. Duda, and P. E. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973
20. A. R. Smith, Color gamut transform pairs, in *SIGGRAPH 78*, 1978, pp. 12–19.
21. N. Herodotou, K. N. Plataniotis, and A. N. Venetsanopoulos, A content-based storage and retrieval scheme for image and video databases, in *Visual Communications and Image Processing '98*, SPIE-3309, San Jose, January, 1998.
22. Y. Gong and M. Sakauchi, Detection of regions matching specified chromatic features, *Computer Vision and Image Understanding* **61**(2), 1995.
23. D. Androustos, K. N. Plataniotis, and A. N. Venetsanopoulos, Distance measures for color image retrieval, in *International Conference on Image Processing '98, Chicago, USA, October 1998*.
24. P. E. Trahanias and A. N. Venetsanopoulos, Vector directional filters: A new class of multichannel image processing filters, *IEEE Trans. Image Process.* **2**(4), 1993, 528–534.
25. K. N. Plataniotis, D. Androustos, S. Vinayagamoorthy, and A. N. Venetsanopoulos, Color image processing using adaptive multichannel filters, *IEEE Trans. Image Process.* **6**(7), 1996, 933–949.
26. P. E. Trahanias, D. Karakos, and A. N. Venetsanopoulos, Directional processing of color images: Theory and experimental results, *IEEE Trans. Image Process.* **5**(6), 1996, 868–880.
27. D. Androustos, K. N. Plataniotis, and A. N. Venetsanopoulos, A vector angular distance measure for indexing and retrieval of color, in *Storage & Retrieval for Image and Video Databases VII*, SPIE-3656, San Jose, USA, 1999, pp. 604–613.
28. M. J. Swain and D. H. Ballard, Color indexing, *Int. J. Comput. Vision* **7**(1), 1991.
29. D. Androustos, K. N. Plataniotis, and A. N. Venetsanopoulos, Efficient color image indexing and retrieval using a vector-based scheme, in *1998 IEEE Second Workshop on Multimedia Signal Processing, Redondo Beach, CA, December 1998*.