

CLUSTER-BASED COLOR MATCHING FOR IMAGE RETRIEVAL

*Mohan S. Kankanhalli**

Babu M. Mehtre

*Jian Kang Wu**

*Real World Computing Partnership, Novel Function ISS Laboratory

Institute of Systems Science

National University of Singapore

Kent Ridge, Singapore 0511

Email: {mohan,mehtre,jiankang}@iss.nus.sg

Phone: (65) 772-3122

Fax: (65) 774-4990

June 8, 1995

Abstract

Color is an important attribute for image matching and retrieval. We present a new method for color matching based on a clustering algorithm in the 3-D color space. We define a new color feature to characterize the color information and a distance measure to compute the color similarity of images. We have implemented this technique and tested it for a database of about 170 images. The test results show that the ‘Efficiency of Retrieval’ of this new method is very high.

Keywords: Color, Matching, Clustering, Measure, Retrieval.

1 Introduction

Color is an important attribute of visual information. Hence, color is a very useful attribute for image matching and retrieval [1]. Color images have not been used widely due to the large storage and high computational requirements. With advances in technology, both the computing & the storage costs are rapidly decreasing. Therefore, color images are increasingly being used now in many applications.

Our work has been motivated by a need to develop efficient color matching techniques so that image retrieval based on color can be effective and fast. This is required in many multimedia information systems such as trademarks databases, face image databases etc. The problem is defined as follows: assume that there are a large number of color images in the database. Given a query image, we would like to obtain a list of images from the database which are most “similar” in color to the query image. For solving this problem, two concepts need to be developed – first, a feature which represents the color information of the image and second, a similarity measure to compute the similarity between the feature values of two images.

The rest of the paper is organized as follows: Section 2 is on the image retrieval model, section 3 is a brief review of past work, Section 4 describes the new method, test results and discussion are given in section 5. Section 6 presents the conclusions.

2 An Image Retrieval Model

Figure 1 shows a block schematic of a generic image archival and retrieval model. In essence, image matching and retrieval is based on some characteristic features of image class under consideration. The input images are analyzed to extract the features and these features are used to store in the image database, along with the original images. These features could, for example, be shape features, texture features or color features. Whenever an image is submitted for search, it is analyzed and its features are extracted. These extracted features are matched against those in the database. A set of closely matching images are brought out as the result of search output.

An important criterion for testing the efficacy of the search and retrieval is that the output *must* include all the similar images. The list may have other images as well, but that is not very important. The important thing is that the similar ones should not be missed in the search process. If a similar image is not brought out, it would defeat the purpose of having an automated search. Such a criterion is important in many applications like trademark registration, fingerprint identification etc., where the system brings out the short list and the final decision is taken by the human expert in the loop.

In this paper, we present an efficient method for image matching when color is used as the feature for matching purpose.

3 Past Work

Recently there has been an increased interest in color research e.g., classification [2, 3], segmentation [4]. Swain and Ballard [5] have proposed a color matching method in their paper on ‘color indexing’. Their method, called *histogram intersection*, is based on matching of color histograms and the core idea in their technique is to compute:

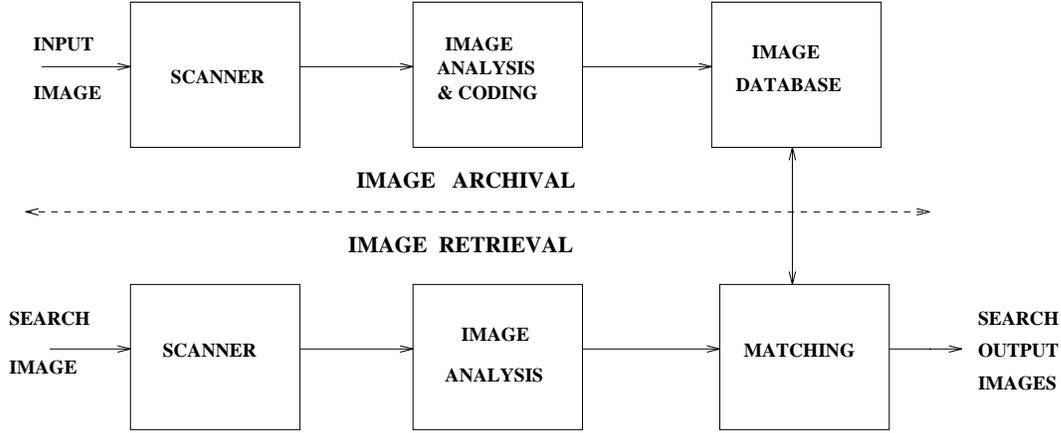


Figure 1: An Image Retrieval Model

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

where $H(I, M)$ is the match value, I and M , are image (query image) and model (an image in the database) histograms respectively, each containing n bins. The match value is computed for every model histogram and the value is closer to unity if the model image is similar to the query image. It is obvious that a match value of unity is obtained for an image compared with itself. In this approach, the feature vector $\bar{\mathbf{f}}$ used to characterize the color information of an image is the 3-D color histogram $h(x, y, z)$ and the similarity measure is given by the match value as shown in eqn. (1). Recent works such as [6, 7] also use some variation of the histogram intersection method.

In the *distance method* [8] the feature used for capturing the color information is the mean value, μ , of the 1-D histograms (normalized by the total number of pixels) of each of the three color components of the image. The feature vector $\bar{\mathbf{f}}$ for characterizing an RGB image is:

$$\bar{\mathbf{f}} = (\mu_R, \mu_G, \mu_B) \quad (2)$$

The following distance measures have been used:

$$D_{q,i}^M = |\overline{\mathbf{f}}_q - \overline{\mathbf{f}}_i| = \sum_{R,G,B} |\mu_q - \mu_i| \quad (3)$$

$$D_{q,i}^E = \sqrt{(\overline{\mathbf{f}}_q - \overline{\mathbf{f}}_i)^2} = \sqrt{\sum_{R,G,B} (\mu_q - \mu_i)^2} \quad (4)$$

where $D_{q,i}^M$ is the Manhattan distance between the query image and a database image, $D_{q,i}^E$ is the Euclidean distance between the query image and a database image, $\overline{\mathbf{f}}_q$ is the color feature vector of the query image and $\overline{\mathbf{f}}_i$ is the color feature vector of the database image.

In the *reference color table method* [8], a set (table) of reference colors is defined. This set of colors is selected such that all the colors in the application are approximately covered perceptually. For every image, the histogram for this smaller set of colors is computed. For this purpose, 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. If the reference color table selected is good, the ‘new image’ obtained by assigning the nearest color from the table, will perceptually be the same as the original image. So, for this method, the color feature chosen is this reduced color histogram based on the colors of the reference table. Thus,

$$\overline{\mathbf{f}} = (\lambda_1, \lambda_2, \dots, \lambda_n) \quad (5)$$

where λ_i is the relative pixel frequency (with respect to the total number of pixels) for the i^{th} reference table color in the image and n is the size of the reference color table. This feature is computed for all the images and stored in the database, along with the original image.

For a particular query image, the feature vector, $\overline{\mathbf{f}}_q$ described above is first computed. Then, this $\overline{\mathbf{f}}_q$ is matched against all $\overline{\mathbf{f}}_i$'s in the database (where $\overline{\mathbf{f}}_i$ is the feature vector of the i^{th} image) to obtain the similarity measure with each image in the database. For computing the similarity, a weighted distance measure is used:

$$D_{q,i} = \omega \sqrt{(\overline{\mathbf{f}}_q - \overline{\mathbf{f}}_i)^2} \quad (6)$$

which leads to:

$$D_{q,i} = \sum_{i=1}^n \omega_i \sqrt{(\lambda_i^Q - \lambda_i^I)^2} \quad \text{where } \omega_i = \begin{cases} \lambda_i^Q & \text{if } \lambda_i^Q, \lambda_i^I > 0 \\ 1 & \text{if } \lambda_i^Q \text{ or } \lambda_i^I = 0 \end{cases} \quad (7)$$

A comparison of the above three techniques has been given in [8]. The reference color table method was found to be the best in terms of retrieval efficiency. However, the drawback of this method is that it requires a pre-defined set of reference colors which can approximately cover all colors in the application. While this condition may be satisfied in some applications, in situations where there are on-going additions/deletions to the database and where knowledge of colors in the images is not available before-hand, such a technique will not give good results. The reference color table method requires a representative sample of all images stored in the database in order to select the reference color table. For example, such a priori knowledge is impossible to obtain in a trademarks database [9]. We have developed a new unsupervised technique based on clustering which does not need any knowledge about the colors in the images to be handled. We discuss the new method in the next section.

4 The Clustering Technique

4.1 The Cluster Feature

The pixels of a color image can be regarded as points in the 3-D color space. Their coordinates are the values of the pixels. The chroma of synthesized color images (also for natural images with a few dominant colors) show continuity in the color space. Usually, since the number of objects in an image is limited, the reflection characteristics of each object in natural light tends to be consistent and results as clusters in the color space. If we perform clustering in the 3-D color space, we would obtain a few clusters, each cluster corresponding to one of the dominant colors in the image. We can take a representative sample (e.g. the mean color) of all such clusters and define a new color feature representing the color information of the image. We

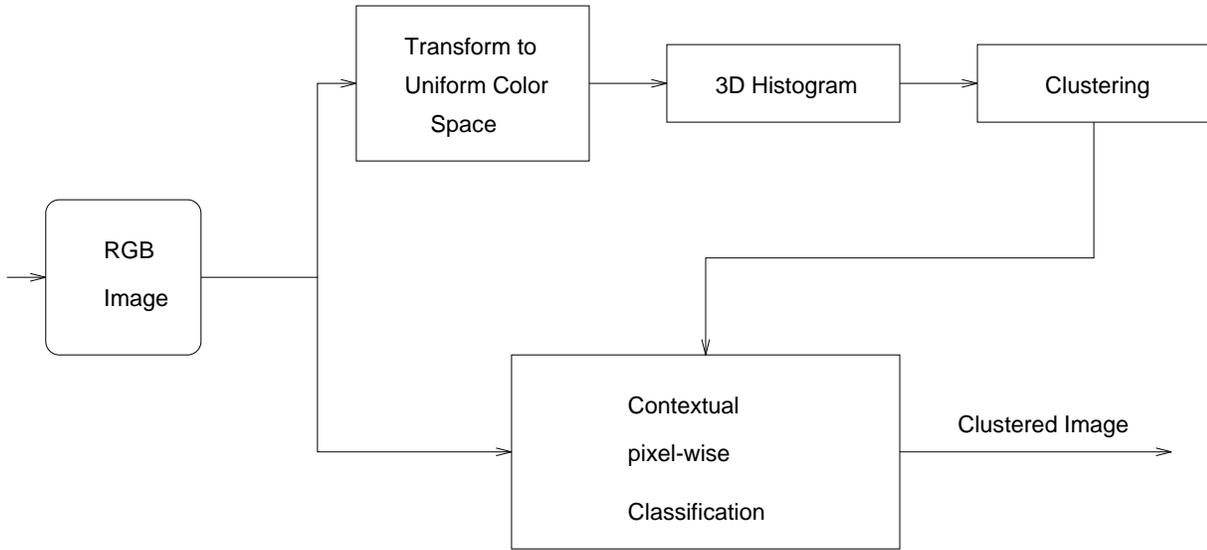


Figure 2: Block Schematic of the Clustering Algorithm

also propose a new distance measure to compute the color similarity between two images based on this feature. But before we give the formal definition of our color feature and similarity measure, we present the clustering algorithm which can be used to perform clustering in color space. This algorithm is based on the method proposed in [10]. An alternative method for clustering is [11].

4.2 The Clustering Algorithm

A block diagram of the clustering algorithm is shown in Figure 2. The input images are in RGB representation. Color image pixels can be regarded as points in the 3-D color space. Now consider synthetic images such as trademark images (or natural images with a few dominant colors). The number of components in such an image is small and each component is represented by one color, and the optical reflection characteristics of each component tends to be consistent. Thus, the color of each component appears as one cluster in the color space. Color clustering can be used to find out these clusters and to assign a representative color to each of these clusters.

Unfortunately, for a given set of color images, there is not enough information about the number and population of clusters. Moreover, training data is usually not available. Therefore, we make the assumption that each color cluster has normal

distribution with one peak and there is sufficient distance between every two color clusters. In this case, the clustering operation can be simplified to a method for finding peaks and estimating distribution parameters.

The distance measure is very important in clustering. The measure should be consistent with the visual resolution and should also be uniform in all directions in the entire color space. We use the CIE 1976 uniform color system (L^*, u^*, v^*) . The distance between two colors can be calculated by the Euclidean distance:

$$d(L^*, u^*, v^*) = [(\Delta L^*)^2 + (\Delta u^*)^2 + (\Delta v^*)^2]^{\frac{1}{2}} \quad (8)$$

We now describe the initial clustering algorithm which does the peak finding in the 3-D color space:

1. Compute the RGB color histogram.
2. Find all the color peaks (6-neighborhood maxima in the 3-D color space) using the histogram.
3. A peak corresponds to a color cluster. For each cluster, note the RGB values and population from the histogram.
4. Sort the peaks in the descending order, based on the cluster population.
5. Determine the number of peaks which do not have a very small population.
6. If the number of peaks found in step 5 is less than in step 3, merge the very small clusters to their nearest color clusters. The nearest cluster is computed based on the color-distance metric in (L^*, u^*, v^*) space. The representative color for this merged cluster is the weighted mean of the two original clusters.
7. For each pixel, compute the color distance to the different clusters. Assign the pixel to the cluster for which color distance is minimum. Thus every pixel gets assigned to one of the clusters.

We now describe in detail the two major problems in reducing the classification error: the design of the classification method, and the utilization of spatial information.

4.2.1 Design of Classification Method

We have obtained the number of classes (clusters), the coordinates of center of each class, and the corresponding amplitude value through the initial cluster analysis in the color space. For more accurate classification, Bayes classifier is not applicable because the covariance matrices are not available. We can use the minimum distance classifier. Minimum distance classifier makes an implicit assumption that all classes have the same population. It is very often that within an image, the difference in number of pixels for different colors is quite large. In this case, the minimum distance classifier will result in large classification errors. Hence, we use a classifier which takes care of population differences in the color space. Suppose we have N samples to classify into M clusters: $\omega_1, \omega_2, \dots, \omega_M$; each cluster being Gaussian distributed $N(m_i, \sigma_i)$, containing N_i samples. All clusters have the same covariance matrix. With the assumption that $N_i \gg 1$, we have the optimal decision rule:

$$d(\mathbf{x}, \mathbf{m}_j) - 2\log[p(\omega_j)] = \min_i \{d(\mathbf{x}, \mathbf{m}_i) - 2\log[p(\omega_i)]\} \quad (9)$$

where $d()$ is the Euclidean distance from the sample to the center of class ω_i calculated in the uniform color space. The ratio of the population of class ω_i to the total samples, $p(\omega_i)$, can be calculated by the ratio of the corresponding i^{th} amplitude value to the total value, N_i/N , obtained by clustering in the 3-D color space.

4.2.2 Use of Spatial Correlation Information

There has been a lot of research to improve the image classification accuracy through the use of spatial information [12]. These methods fall into three main categories:

1. Relaxation Methods, which represent spatial correlation information by consistency coefficients.
2. Contextual Classifiers, which directly make use of spatial information.
3. Use of image texture measure, together with the spectral data.

The first two methods are very complicated and computation-intensive. The third method does not seem to be applicable in our case. We therefore use a predictive classifier based on the Markov random field model of image data. In the first order Markov random field, a pixel is closely related to its two neighbors (top and left). It means the probability that a pixel and its neighbors belong to the same color class is relatively high. The following decision procedure is then applicable:

Perform classification using equation 9 in a raster scan order. For the current pixel,

- Record the class label of top and left neighbor pixels ω_t and ω_l and compute corresponding distance measures $d_t()$ and $d_l()$ (i.e. the distance of the pixel from the class mean).
- Calculate the decision function $d_c()$ which measures the distance between the present pixel and its current class ω_c . This distance $d_c()$ is used for deciding the new label for the current pixel. Then, if $|d_t - d_c| < \theta$ then the pixel is assigned class ω_t , otherwise class ω_l is given the same consideration. If the pixel can not be classified to either class ω_t or class ω_l , it is classified using equation 9.

This method makes use of spatial correlation property of image data. It also reduces the effect of noise in the clustering and saves considerable computation time.

4.3 Cluster Matching

We present the new color feature and the similarity measure in this section. Suppose we have an image I of size N pixels and assume that a clustering operation on the image obtains m color clusters. Each cluster C_i is defined as follows:

$$C_i = \{R_i, G_i, B_i, \lambda_i\} \text{ where } i = 1, 2, \dots, m \quad (10)$$

where (R_i, G_i, B_i) is the representative color of the cluster (we use the mean) and λ_i is the fraction of the pixels in that cluster as compared to the total number of

pixels, i.e.

$$\lambda_i = \frac{\text{Number of pixels } \in C_i}{N}. \quad (11)$$

Now, we can define the color feature of image I as:

$$\bar{\mathbf{f}}_I = \{C_i \mid i = 1, 2, 3 \dots m\} \quad (12)$$

C_i refers to a cluster in the color space computed by some clustering algorithm. Now this feature vector can be computed for every image and stored in the database.

The similarity between a pair of images Q and I is given by a distance function, $d(Q, I)$. It is not necessarily true that the two images to be compared have similar or same number of colors. Assume that image Q has m clusters and image I has n clusters and that $m \leq n$ (the reverse case can be similarly handled). We first compute the permutation function P for Q which maps every cluster i of Q to the closest cluster (in color space) $P(i)$ of image I . This permutation can be very easily computed, as follows:

1. Form the distance matrix $G = [g]_{pq}, 1 \leq p \leq m, 1 \leq q \leq n$ where $g_{pq} = \text{coldis}(C_p^Q, C_q^I)$.
2. Find the minimum entry g_{xy} in G .
3. $P(x) = y$.
4. Delete row x and column y (but do not change the index numbers of the rows & columns of matrix G).
5. If G is empty (all rows and columns have been deleted) then stop else go to step 2.

The color distance in the above algorithm can be the simple euclidean distance:

$$\text{coldis}(C_p^Q, C_q^I) = [(R_p^Q - R_q^I)^2 + (G_p^Q - G_q^I)^2 + (B_p^Q - B_q^I)^2]^{\frac{1}{2}} \quad (13)$$

If desired, the RGB representation could be converted to the 1976 CIE color representation and the distance can be computed in the CIE space using equation 8. The CIE space distance is perceptually a better distance measure than the RGB space.

For computing the similarity, we use a weighted distance measure:

$$D(Q, I) = \omega \sqrt{(\mathbf{f}_q - \mathbf{f}_i)^2} \quad (14)$$

which leads to:

$$D(Q, I) = \sum_{i=1}^{\min(m,n)} \omega_i \sqrt{(\lambda_i^Q - \lambda_{P(i)}^I)^2} + \sum_{i=1}^{\min(m,n)} \text{cdist}(C_i^Q, C_{P(i)}^I) \quad (15)$$

where:

$$\omega_i = \begin{cases} \lambda_i^Q & \text{if } \lambda_i^Q, \lambda_{P(i)}^I > 0 \\ 1 & \text{otherwise} \end{cases} \quad (16)$$

and

$$\text{cdist}(C_i^Q, C_{P(i)}^I) = \begin{cases} \text{coldis}(C_i^Q, C_{P(i)}^I) & \text{if } \lambda_i^Q, \lambda_{P(i)}^I > 0 \\ 1 & \text{otherwise} \end{cases} \quad (17)$$

where \mathbf{f}_q and \mathbf{f}_i are the color features of the query image Q and database image I respectively. Note that λ_i^Q and $\lambda_{P(i)}^I$ are the i^{th} cluster's relative pixel frequency of the query and database images respectively. If the number of clusters is not the same, the value of $\lambda = 0$ is assumed for the corresponding cluster of any unmatched cluster in the image with the larger number of clusters.

The distance formula has two factors – the relative frequency of the pixels of the corresponding clusters as well as the color distance between the two clusters. Note that ω is the weight factor used. For a pair of corresponding clusters, if both the histogram bins are non-zero, then the weight ω_i used is λ_i^Q since we want to take the relative proportion of color i in that image. If either of the corresponding histogram bins have a value of zero (which means that it is an unmatched cluster), then ω_i is taken to be unity. In this case the relative difference of the two bins is used as a *push factor* to separate the two images in the similarity measure. Therefore the first term (relative frequency term) represents the proportion of each individual color in the image. This factor in the distance expression helps to rank the images based on color similarity when the retrieved images have either an unequal number of colors

(more or less number of colors) or a different mix of proportions of the individual colors than the query image. If the two corresponding clusters have exactly the same relative frequency, then the color distance becomes the primary factor in deciding the distance. It is obvious here that the distance of an image from itself is zero.

As far as the computational complexity is concerned, both the cluster method and the reference color table method require $O(m^2)$ time for a $m \times m$ image for the feature computation (which is done only once). For the matching part, the reference color table requires $O(L)$ time where L is the size of the reference color table. The cluster method requires $O(pq)$ time for the matching, where p and q are the number of clusters respectively in the two matched images. The actual time taken is not very high since the number of dominant colors in an image is usually very small.

5 Test Results and Discussion

We have implemented the cluster-based color matching algorithm in an image information system and tested it on two databases. We first used one database of 100 airline trademark images and then tested on another database of 70 flag images of different countries.

We designed the tests in this manner: we picked 10 query images each for both the databases which represented the population well. For each of these query images, we manually listed the similar images found in the database. Then we applied the techniques to each query image against all the images in the database to obtain *short lists* of similar images. The summary of results are presented in table 2. For any query, we define the efficiency of retrieval, η_T , for a given short list of size T as follows:

$$\eta_T = \begin{cases} \frac{n}{N} & \text{if } N \leq T \\ \frac{n}{T} & \text{if } N > T \end{cases} \quad (18)$$

where n is the number of similar images retrieved in the short list and N is the total number of similar images in the database. Table 1 represents the retrieval efficiency, η_T , for both the methods, averaged over 10 queries. We have presented the results

Matching Method	Airlines DB				Flags DB			
	T=5	T=10	T=15	T=20	T=5	T=10	T=15	T=20
Hist. Intersection	0.547	0.645	0.713	0.771	0.475	0.625	0.778	0.868
Distance Method	0.880	0.925	0.986	0.993	0.612	0.837	0.958	0.983
Ref. Color Table	0.930	0.987	0.987	1.000	0.943	0.986	0.988	1.000
Cluster Matching	0.912	0.968	0.986	0.997	0.919	0.967	0.979	1.000

Table 1: Average Efficiency of different methods over 10 queries.

for short list sizes of $T = 5, 10, 15$ and 20 . It can be seen from table 1 that the efficiency of the clustering method is superior to that of the histogram intersection and the distance method. While the efficiency of the reference color method is better than the clustering method, both are almost of the same order. This shows that if we have a priori knowledge of the colors in the database, then one can use a suitable reference color table to obtain good retrieval efficiency. However, in the cases, where such information is not available, the clustering method is very good since it can give almost a similar efficiency. Figure 3 shows a sample query image with the retrieved images for the airline database with a short list of size $T = 9$ using the clustering method. The image on the lower-right is the query image and the results of the query are presented as a 3×3 grid of image icons. The images are ranked from most similar to least similar in a left-to-right and top-to-bottom scan order. Any of these icons can be clicked in the system and the corresponding image is displayed on the lower-left area.

6 Conclusion

We have presented a new method of color matching for the purpose of image retrieval. We have defined a new color feature based on color clustering and have defined a new distance measure suitable for matching this feature. The advantage of our new technique is that it gives very high retrieval efficiency without requir-



Figure 3: Retrieval Results for a Sample Query Image

ing prior knowledge of the colors involved in an application. We are planning to incorporate the color feature in our Trademarks image database which uses shape, text and phonetics as other features [9]. We believe that the cluster feature can be useful for indexing purposes since clustering can be viewed as a generalized form of indexing [13].

7 Acknowledgement

We thank G.C. Man of Nanyang Technological University for his help in the implementation. We are also grateful to our colleagues: C.P. Lam for implementing the clustering algorithm and A. Desai Narasimhalu for giving many insightful comments.

References

- [1] 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 colour, texture and shape. *Proc. of SPIE*, 1908:173–187, 1993.
- [2] I. Andreadis, M.A. Browne, and J. Swift. Image pixel classification by chromaticity analysis. *Pattern Recognition Letters*, 11:51–58, 1990.
- [3] T. Caelli and D. Reye. On the classification of image regions by colour, texture and shape. *Pattern Recognition*, 26(4):461–470, 1993.
- [4] G. Healey. Segmenting images using normalized color. *IEEE Transactions on Systems, Man and Cybernetics*, 22(1):64–73, 1992.
- [5] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.
- [6] Y. Gong, H. Zhang, H.C. Chuan, and M. Sakauchi. An image database system with content capturing and fast image indexing abilities. In *Proc. IEEE In-*

- ternational Conference on Multimedia Computing and Systems*, pages 121–130, 1994.
- [7] H. Lu, B.C. Ooi, and K.L. Tan. Efficient image retrieval by color contents. *Applications of Databases, First International Conference ADB-94, Lecture Notes in Computer Science*, 819:95–108, 1994.
- [8] 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.
- [9] J.K. Wu, B.M. Mehtre, Y.J. Gao, C.P. Lam, and A.D. Narasimhalu. Star - a multimedia database system for trademark registration. *Applications of Databases, First International Conference ADB-94, Lecture Notes in Computer Science*, 819:109–122, 1994.
- [10] J.K. Wu. Color coding of images. *Signal Processing (Official Journal of ASSP, China)*, 3(1):1–7, 1987.
- [11] M. Celenk. A recursive clustering technique for color picture segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 437–444, 1988.
- [12] T.Y. Young and K.S. Fu (editors). Handbook of pattern recognition and image processing. Academic Press, 1986.
- [13] W.I. Grosky and R. Mehrotra. Image database management. *Advances in Computers (M. Yovits, ed., Academic Press)*, 35:237–291, 1992.

Author Biographies

Mohan S. Kankanhalli received the B.Tech. degree in Electrical Engineering from the Indian Institute of Technology, Kharagpur in 1986. He obtained the M.S. and Ph.D. degrees, both in Computer & Systems Engineering, in 1988 & 1990 from the Rensselaer Polytechnic Institute, Troy, New York. His doctoral work involved developing techniques for designing parallel algorithms for geometric problems. These techniques have been applied to computer graphics, CAD and solid modeling. Since then, he has been a researcher with the Institute of Systems Science (ISS) – National University of Singapore, where he initially worked in Medical Visualization & Imaging. He developed several algorithms for medical data manipulation and visualization. He also developed a volume modeling system for orthopedic surgery. He is now with the Multimedia group, working on projects in Content-based image retrieval and Spatial Information Systems. His research interests include computer graphics & imaging, multimedia systems, geometric algorithms, parallel algorithms and medical imaging.

Babu M. Mehtre is a researcher with the archival and retrieval of multimedia information group at ISS. He got his B.E. (Electronics and Communications Engineering) with distinction from Gulbarga University in 1983, M. Tech. (Automation and Control Eng'g), and Ph.D. (Engineering) from Indian Institute of Technology, Kharagpur, in 1984 and 1990 respectively.

Dr Mehtre was with the Image Processing Group at CMC R&D Centre, Hyderabad, India, from 1985 to 1992. During this time, he worked on a fingerprint identification system project and a map understanding system. He has received four national awards from the Government of India for his work in fingerprints. He has

also received a special award for his Master's thesis. He was a visiting scholar at the Pattern Recognition and Image Processing Lab, Michigan State University during the first quarter of 1990.

Dr Mehtre has reviewed papers for various international journals and conferences, and is on the program committee of the International Workshop on Graphics Recognition (GREC'95), Pennsylvania State University, August 1995. He is a member of IEEE, Pattern Recognition Society, and Singapore Computer Society. His areas of interests include graphics, image processing, pattern recognition, multimedia systems, and applications.

Jian Kang Wu received his Bachelors degree from the University of Science & Technology of China (USTC), and PhD from the University of Tokyo. He is now with ISS as a Member, Research Staff. Prior to joining ISS, he worked in USTC as a full professor. He also worked as an Alexander-Humboldt research fellow at University of Erlangen-Nunberg in 1989, visiting associate professor at University of Pittsburgh in 1986, foreign research fellow at the University of Tokyo in 1982 and academic visitor at Imperial College, London in 1980.

He is now heading a group doing multimedia research. His research interests are: multimedia data modeling; flexible indexing, retrieval, and intelligent utilization of multimedia data; composition, recognition, and planning with multimedia database systems; fuzzy-reasoning and neural network self-organization for multimedia data storage, retrieval and utilization; 2-D and 3-D spatial database systems; computer vision and neural networks.

Dr. Wu is an author of 4 books and more than 50 journal papers. He has conducted more than 15 projects. He has been awarded the title of "Research Pioneer" from the National Committee of Science & National Committee of Education in China and several prizes from Chinese Academy of Sciences.