Proceedings of the 2003 Systems and Information Engineering Design Symposium
Matthew H. Jones, Barbara E. Tawney, and K. Preston White, Jr., eds.

IMAGE RETRIEVAL USING SEGMENTATION

Sridhar R. Avula
Jinshan Tang
Scott T. Acton

Virginia Image and Video Analysis
Department of Electrical and Computer Engineering
University of Virginia,

ABSTRACT

A novel algorithm for image retrieval is presented in this paper. The basic idea of the new algorithm is that the constituent segments of the images are used to retrieve images within a digital library. Image retrieval using segments is distinctive in that the local features of the image are used to retrieve the image instead of the typically utilized global features. In our algorithm, the given image is first segmented into dominant components and then the features of these components are extracted to perform retrieval. The features corresponding to each component are used to calculate the distance between components in the matching process. Each image is ranked based on the component-wise distance measure with respect to the query component. One of the advantages of the algorithm is that, for a given retrieval image, the user can select a query segment with which to perform retrieval, thus it can satisfy different needs from different users.

1 INTRODUCTION

With the exponential growth in image databases available on the internet, we need an efficient storage, cataloging and retrieval system for images. Images in a database are typically indexed using text annotation, which is dependent upon the language and point of view of the operator. Content based image retrieval (CBIR), in contrast, is the method of retrieving images similar to a given query image using only the content of the image. The content of the image can be represented by many features such as color, texture, and shape.

There are many algorithms for image retrieval. However, most of them are based on the global features of the images [1]. One disadvantage of global feature-based image retrieval is that a user is often interested only in a single object or in a few objects in the image, whereas a global mechanism uses all features for retrieval, including the background features. In order to overcome this shortcoming of the global feature-based image retrieval, a new image retrieval algorithm based on local features is proposed.

The new algorithm exploits image segments in the retrieval process. In segment-based image retrieval, a given image is divided into homogeneous regions and retrieval is based on a subset of the objects present in the image. Automated segmentation is a very powerful tool in retrieval because the user can narrow the field of search by selecting the objects in the image. In this paper we attempt to provide a segmentation algorithm and retrieve images based on the component features. Color, texture and shape of the component are extracted and stored in a database for use in CBIR.

Figure 1 provides a flow chart for the algorithm. The images in the library are segmented and features are extracted off-line. When a retrieval is initiated, the query segments are matched with segments from the feature database and the retrieved images are displayed.

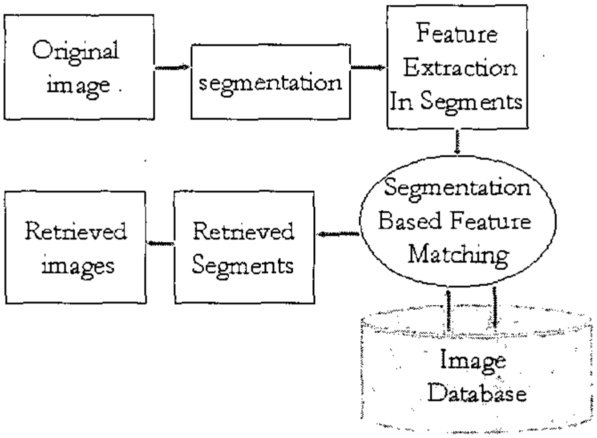

Figure 1. The new image retrieval algorithm.

Section 2 discusses the segmentation algorithm and feature extraction. Section 3 describes the similarity measure and segment based image retrieval. Experimental results are shown in section 4. Conclusions and discussion are made in Section 5.

2 IMAGE SEGMENTATION AND FEATURE EXTRACTION

2.1. Segmentation

Segmentation is the process of dividing an image into homogenous regions. A standard k-means image classification algorithm [2] is used in this paper to segment the image. The k-means is implemented with a different initialization.

In the algorithm, the color image is first converted into a grey scale image and then thresholded into classes to initiate the k-means classification [2]. Here we use the RGB color space but we believe this algorithm can be implemented in other color spaces with the same amount of success.

Let I represent the 2-dimensional color image. We get the grayscale image by converting R,G and B color values to grayscale. The k-means is initiated by thresholding the grayscale image into K classes. Since grayscale image approximately represents the variation in the color intensities in an image, we can increase the efficiency of the k-means classification by this initialization.

For the first iteration, K initial 3-D cluster centers $z_1(1), z_2(1), \ldots, z_K(1)$ (1 means the first iteration here) are calculated using the R, G, B values of color image. Each dimension represents the cluster center in each color.

$$z_i(1) = \frac{1}{N_i} \sum_{(x,y)\in A_i(1)} I(x,y) \tag{1}$$

for all $i = 1,2,\ldots,K$. Here $z_i(1)$ is the cluster center and N_i is the number of pixels for the i-th cluster, $A_i(1)$ denotes the set of pixels whose cluster center is $z_i(1)$.

In subsequent iterations, the color image is used to iterate the k-means. The distance between the mean intensity of each class is calculated as the distance between two points in a 3-D (RGB) Euclidian space. For the n-th iterative step the pixels of the image can be distributed among the cluster centers using $(x,y) \in A_j(n)$ if

$$\|I(x,y) - z_j(n)\| < \|I(x,y) - z_i(n)\| \tag{2}$$

for all $i = 1,2 \ldots,K$; $i \neq j$ where $A_j(n)$ denotes the set of pixels whose cluster center is $z_j(n)$ in the RGB domain. After each stage the cluster centers are updated using RGB intensities of all the pixels of the updated classes as follows

$$z_j(n+1) = \frac{1}{N_j} \sum_{(x,y)\in A_j(n)} I(x,y). \tag{3}$$

If the cluster size N_j of each class $j = 1,2,\ldots,K$ remains unchanged, the classification is terminated.

Each class extracted from the k-means algorithm is stored as a binary matrix B_i.

$$B_i(x,y) = \begin{cases} 1 & if \quad (x,y) \in A_i(T) \\ 0 & \quad others \end{cases} \tag{4}$$

for all $i = 1,2,\ldots, K$, where B_i is the binary matrix and T is the number of iterations.

The binary classes are processed by a set of order statistic filters to connect any two unconnected components which actually are a part of the same region in the image but are separated, for example, by a thin line. The morphology is also effective in disconnecting two regions which are connected by a thin "isthmus" region.

$B_i' = \text{median}(B_i, M)$ for all $i = 1,2,\ldots,K$ where M is a window.

Each class i is labeled into p_i regions using connected component labeling (with 4-connectivitity) [3][4]. Let R_{it} represent the t–th region in the i-th class. The region R_{is_i} with maximum area is chosen as the dominant component in each class.

$C_i(x,y) = 1$, if $(x,y) \in R_{is_i}$, for all $i=1,2,\ldots,K$.

where $s_i = \arg(\max_{t=1}^{p_i}(area(R_{it})))$ is the region of maximum area of all the p_i regions labeled in each class i which is stored in C_i.

All the "holes" in the region C_i, with area very small compared to the area of component, are filled. These holes are usually caused by bright spots, noise and speckles in the image that become classified into a different class by the k-means algorithm. This is achieved by labeling C_i as a foreground (R_{fg})region and the rest as a background(R_{bg}) region. All the background regions whose boundaries are included in the component or the "holes" in the component and whose area is significantly small compared to the foreground region are merged into the component. This is done so that the boundary of the component now represents its shape.

$C_i' = C_i - R_{bg}$ if area(R_{bg}) << area(R_{fg}) for all $i = 1,2,\ldots,K$.

After this post-processing, the components C_i' are stored in the database for each image and are accessed when needed for retrieval. Figure 2 shows an example of our image segmentation algorithm.

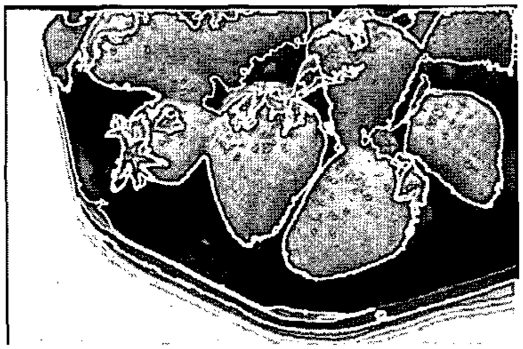

Figure 2. Segmentation of a color image.

2.2 Feature Extraction:

For each segment, we extract three features: color, texture and shape. Texture is an important feature which describes the regularity, smoothness and coarseness of the image. Many techniques have been proposed to describe texture feature. We use the method developed in [5] to extract the texture features. The only difference is that the extraction of texture feature here is not from the whole image but from the segments. To calculate the coefficients a square is fit inside the component such that all parts of the square are inside the component. This part of the component is used to calculate the wavelet coefficients. The similarity in texture D_T between any two components is calculated as the Euclidean distance between the derived wavelet coefficients.

$$D_T(Q_i, D_j) = \sqrt{\sum_k (W_{i,Q}(k) - W_{j,D}(k))^2} \quad (5)$$

where $W_{i,Q}(k)$ and $W_{j,D}(k)$ are the energies in the k-th subband of the i-th component of the query image Q and of the j-th component of the database image D, respectively.

Color is another important feature used in image retrieval. In the recognition of objects, color has enormous importance because it is largely view invariant and resolution independent [6]. The color histogram is obtained by measuring the frequency of a particular color in the image array. A benefit of this basic representation is that histograms are invariant to translation and rotation [6]. For a given component of an image its color content is extracted in the form of the histogram of each R, G and B value. Here again the distance D_C between components is calculated similarly taking the histogram values of each component.

$$D_C(Q_i, D_j) = \sqrt{\sum_k (H_{i,Q}(k) - H_{j,D}(k))^2} \quad (6)$$

where $H_{i,Q}, H_{j,D}$ are the histogram values of the i-th component of the query image Q and of the j-th component of the database image D respectively.

Besides color and texture, we also use shape to perform retrieval. Shape of an object can be used to distinguish the object from other objects with the same color and texture.

The component C_i' of the i-th class is dilated [7] in order to get a smooth shape.

$C_i'' = C_i' \oplus G'$ for all i= 1,2,...,k. where G_i' is a structuring element.

A shape description that is robust to translation, rotation and scaling is the Fourier shape descriptors (FSD). The standard FSD approach first samples the contour at equal angles and then measures the radial distances of the samples and the angles subtended at the center of mass of the contour. The main drawback of this contour sampling is the non-uniform sampling of the contour – the portions of the contour that are closer to the center of mass are sampled more densely than the parts that are more distant.

In our algorithm, we use a contour tracing algorithm to trace the contours. The given contours are assumed to be closed, continuous and 8-connected. The radial sweep algorithm [8] is employed to trace the contour, which is then down-sampled at regular intervals to obtain the point coordinates. The center of mass of the contour is calculated by taking the average of the coordinates in the x and y directions. For each of the sampled points on the contour, the radial distance from the center of mass and the angle made with some starting point is calculated and stored. The radial distances and the angles of all the sampled points are derived and stored in a complex number form as shown in equation (7). We then normalize the radial distances for all the coordinates.

The radial distances and the angles of all the sampled points are stored in the form of a complex-range vector. The real part of each complex number represents the distance, and the imaginary part represents the angle subtended in radians as:

$$z(i) = r(i) + j\theta(i) \quad (7)$$

where $r(i)$ is the radial distance and $\theta(i)$ is the angle subtended at the center of mass by the sample.

The Fourier shape descriptors of the coordinates are calculated using the following equation [9]

$$Z(k) = \left(\frac{1}{N}\right)\sum_{i=1}^{N} z(i)\exp(-j2\pi ik/N) \quad (8)$$

where N is the number of sampled points. Here $N = 64$ and $Z(k)$ is the k^{th} Fourier coefficient.

The magnitude of the Fourier coefficients for each image are calculated and stored for use in the retrieval of similar shapes. Let $Z_Q(k)$ and $Z_D(k)$ represent the coefficients of the query component and an image component in the database. The similarity between shapes D_S of these two components is measured by

$$D_S(Q_i, D_j) = \sqrt{\sum_k (Z_{i,Q}(k) - Z_{j,D}(k))^2} \qquad (9)$$

3 SEGMENTS BASED IMAGE RETRIEVAL

In the retrieval process, a query image is first segmented and the dominant components found are highlighted. Once a dominant component is chosen, the corresponding features are compared with the features of the other image components in the database, and the *M* closest matches are retrieved.

In this paradigm, we use a similarity measure combing the three features. Let the normalized color similarity and the normalized texture similarity between two segments be D_C and D_T which can be obtained by the method in Eq. (5) and (6). D_S is the normalized similarity which can be obtained by (9). Then the similarity between two segments d is computed as

$$d(Q,D) = w_c D_C + w_t D_T + w_s D_S \qquad (10)$$

where Q is the query segment and the D is a segment from an image in the database. w_c, w_t and w_s are weights for color, texture and shape features respectively.

4 EXPERIMENTS

Our first experiment used color and texture to perform retrieval, where color and texture have been given equal weight. In the above experiments the number of classes was chosen as $K = 5$. The images are typically of the size 500 X 700 pixels. Shape has been ignored in this example. Fig 3. (b)-(f) show the retrieved images when using only the water segment as a query segment and (g) - (k) show retrieved images when using the global color and texture features.

In the second example, we use color and shape to perform retrieval. Figure 4 shows the experimental results. For quantitative analysis, we plot the number of images successfully retrieved against the total number of images retrieved using some or all features in measuring the similarity of images in Figure 5. An image is deemed "successfully retrieved" if it belongs to be the desired class of images in the query, e.g., images containing a body of water. From the graph we observe that percentage of successful images retrieved has improved when all the three features were used for retrieval.

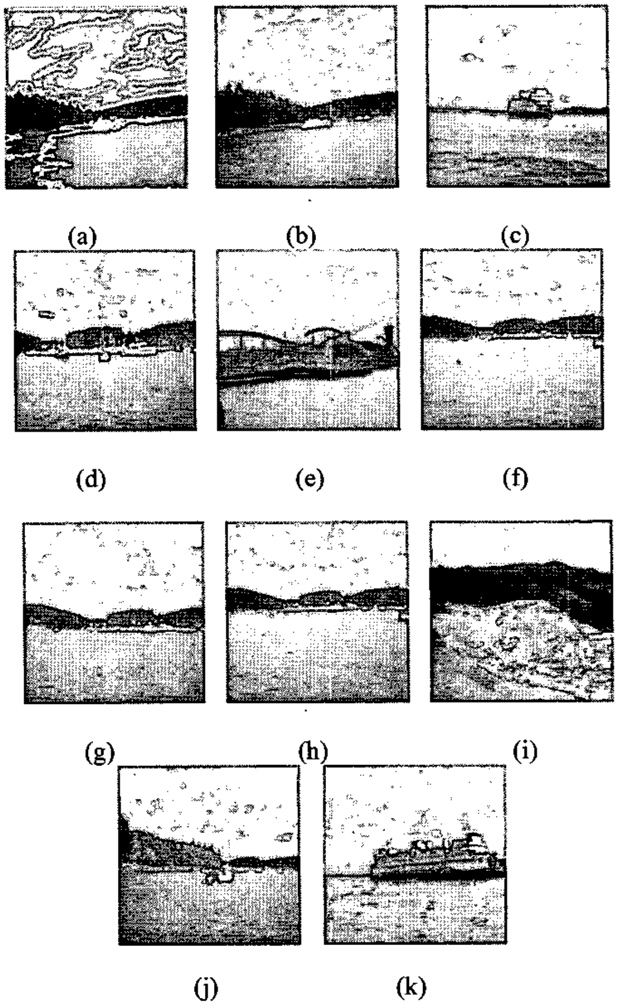

Figure 3. (a) is the query image . (b)-(f) are retrieved images using only the water segment as a query segment. (g)-(k) are retrieved images using the global color and texture features.

5 CONLCUSION AND DISCUSSION

A new retrieval method using segmentation is proposed for a general image database. The segmentation step can improve the retrieval algorithm by decreasing the scope of search. Experimental results are encouraging and indicate the potential effectiveness of the algorithm. Future work includes improving the k-means step to automatically choose the number of classes before clustering. Furthermore, the next deployment of the classifier will include texture in the classification step. Finally, extensive testing over a vast, diverse database containing thousands of images will provide quantitative evaluation of the retrieval efficacy.

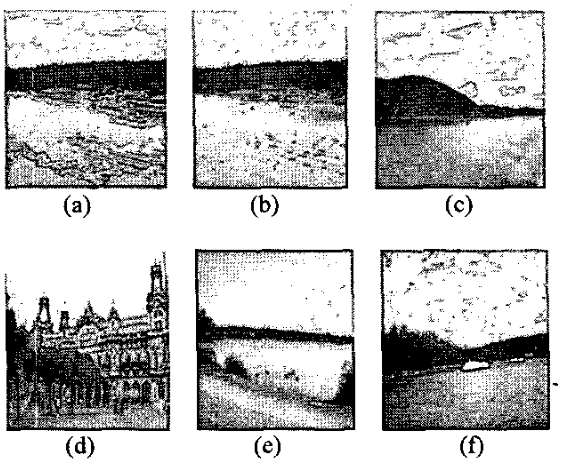

Figure 4. (a) is the query image . (b)-(f) are retrieved images using only the green segment as a query segment.

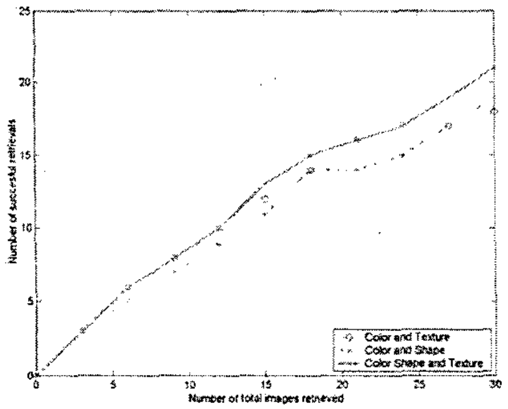

Figure 5. Plot of number of images successfully retrieved against total number of images retrieved using color and texture only (red circle), color and shape only (green dash dot) and using all the three (blue solid).

REFERENCES

[1] A. Pentland, R.W. Picard and S. Sclaroff. . Photobook: Content-based manipulation of image databases. *Int. Journal of Computer Vision*, vol. 18, pp. 233-254, 1996.

[2] S. Ray and R. H. Turi, "Determination of number of Clusters in K-Means Clustering and Application in Colour Image Segmentation," 4th ICAPRDT Proc., pp.137-143, 1999.

[3] S.T. Acton and D. P. Mukherjee, "Scale Space Classification using Area Morphology," *IEEE Trans. IP*, vol. 9, pp. 623-635, 2000.

[4] S. Marchand-Mailet and Y. M. Sharaiha, *Binary Digital Image Processing: A Discrete Approach*, Academic Press, 2000.

[5] Jinshan Tang, Scott Acton, " A Decentralized Image Retrieval System for Education," See the paper in this Proceeding .

[6] M. J. Swain and D.H. Ballard. "Indexing via color histograms", *Computer Vision Proc.*, pp. 390-393, 1990.

[7] A. Bovik, *Handbook of Image and Video Processing*, Academic Press, pp. 43-49, 2000.

[8] A.G. Ghuneim, "Contour tracing algorithms tutorial on implementation" [online]. Available via http://www.cs.mcgill.ca/~aghnei/alg.html [accessed April 1, 2003].

[9] R.C. Gonzales and R.E. Woods, *Digital Image Processing*, Addison-Wesley Pub. Co., pp. 497-500, 1992.

AUTHOR BIOGRAPHIES

SRIDHAR R. AVULA is a research assistant graduate student scheduled to complete the requirements for an M.S. degree in Electrical Engineering in August 2003. Avula received a Bachelor of Technology degree from Indian Institute of Technology Bombay, India in 2001. Avula's areas of research include content based image retrieval. He is presently working on segmentation of images into objects and deriving the Fourier shape descriptors as a part of the "DIRECT" content based image retrieval system.

JINSHAN TANG received his Ph.D. degree from Beijing University of Posts and Telecommunications, P.R. China, in 1998. From September 1998 to June 2000, he worked as an invited researcher in ATR Media Integration and Communications Research Laboratories (MIC), Kyoto, Japan. In June 2000, he went to Harvard to work on image enhancement technology. Since Dec. 2001, Dr. Tang has been working as a research scientist in Department of Electrical and Computer Engineering, University of Virginia. Dr. Tang is a senior member of the IEEE. His research interests are smart computers, content based image retrieval, cell detection and tracking, face detection and recognition, image enhancement for low-vision.

SCOTT T. ACTON received his M.S. and Ph.D. degrees in Electrical and Computer Engineering from the University of Texas at Austin in 1990 and 1993, respectively. He has a B.S. degree in Electrical Engineering from Virginia Tech (1988). He has worked in industry for AT&T, the MITRE Corporation and Motorola, Inc. and in academia for Oklahoma State University. For his research in video tracking, Dr. Acton was given an ARO Young Investigator Award. He received the Halliburton Outstanding Young Faculty Award in 1998. In 1997, he was named the Eta Kappa Nu Outstanding Young Electrical Engineer -- a national award that has been given annually since 1936. Currently, Dr. Acton is the Walter N. Munster Associate Professor and Faculty Fellow in the Department of Electrical and Computer Engineering at the University of Virginia. At U.Va., he was named the Outstanding New Teacher in Electrical and Computer Engineering. Dr. Acton is a senior member of the IEEE and has served as Associate Editor for the *IEEE Transactions on Image Processing* and the *IEEE Signal Processing Letters*.