International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:10, 2008

A Universal Model for Content-Based Image Retrieval

S. Nandagopalan, Dr. B. S. Adiga, and N. Deepak

Abstract**—**In this paper a novel approach for generalized image retrieval based on semantic contents is presented. A combination of three feature extraction methods namely color, texture, and edge histogram descriptor. There is a provision to add new features in future for better retrieval efficiency. Any combination of these methods, which is more appropriate for the application, can be used for retrieval. This is provided through User Interface (UI) in the form of relevance feedback. The image properties analyzed in this work are by using computer vision and image processing algorithms. For color the histogram of images are computed, for texture co-occurrence matrix based entropy, energy, etc, are calculated and for edge density it is Edge Histogram Descriptor (EHD) that is found. For retrieval of images, a novel idea is developed based on greedy strategy to reduce the computational complexity. The entire system was developed using AForge.Imaging (an open source product), MATLAB .NET Builder, C#, and Oracle 10g. The system was tested with Coral Image database containing 1000 natural images and achieved better results.

Keywords**—**Content Based Image Retrieval (CBIR), Co-occurrence matrix, Feature vector, Edge Histogram Descriptor (EHD), Greedy strategy.

I. Introduction

THE increasing amount of digitally produced images requires new methods to archive and access this data. Conventional databases allow for textual searches on meta data only. Content Based Image Retrieval (CBIR) is a technique which uses visual contents, normally called as features, to search images from large scale image databases according to users' requests in the form of a query image [1], [8], [9].

The commercial image search engines available as on date are: QBIC, VisualSeek, Virage, Netra, PicSOM, FIRE, AltaVista, etc. Region-Based Image Retrieval (RBIR) is a promising extension of CBIR [12]. Almost all the CBIR systems designed so far widely use features like color, shape, textures, and spatial all together or few of these. For example, [2] describes a method for image retrieval purely based on color and texture.

In this paper apart from the usual features like color and texture, a new feature extraction algorithm called edge histogram is introduced. Edges convey essential information to a picture and therefore can be applied to image retrieval. The edge histogram descriptor captures the spatial distribution of edges [2], [16]. Our model expects the input as Query By Example (QBE) and any combination of features can be selected for retrieval.

The focus of this paper is to build a universal CBIR system using low level features. These are mean, median, and standard deviation of Red, Green, and Blue channels of color histograms. Then the texture features such as contrast, energy, correlation, and homogeneity are retrieved. Finally the edge features that include five categories vertical, horizontal, 45 degree diagonal, 135 degree diagonal, and isotropic are added [2].

The rest of the paper is organized as follows. Since there was lot of work done in this area, a comprehensive survey of CBIR is dealt in Section 2. Section 3 describes the overview of the proposed CBIR framework. Section 4, 5, and 6 focuses on extraction of color, texture and edge density features respectively. In Section 7, database tables and types for the feature vector is shown. Section 8 is to provide particulars of experiments conducted followed by results and conclusion.

II. Related Work

Early work on image retrieval can be traced back to the late 1970s. In 1979, a conference on Database Techniques for Pictorial Applications was held in Florence. Since then, the application potential of image database management techniques has attracted the attention of researchers. In the early 1990s, as a result of advances in the Internet and new digital image sensor technologies, the volume of digital images produced by scientific, educational, medical, industrial, and other applications available to users increased dramatically. The difficulties faced by text-based retrieval became more and more severe. The efficient management of the rapidly expanding visual information became an urgent problem.

Local features based methods proved good results [11]. For a successful CBIR, note that the indexing scheme to be efficient for searching in the image database. Recent retrieval

S. Nandagopalan is with the Bangalore Institute of Technology, Bangalore affiliated to Visvesvaraiah Technological University, Karnataka, India (e-mail: snandagopalan@gmail.com).

B. S. Adiga was with National Aeronautical Laboratory, Bangalore. He is now with the Tata Consultancy Services, Bangalore (e-mail: b.s.adiga@tcs.com).

N. Deepak is working as a Research Assistant for this work (e-mail: deepakgi2005@gmail.com)

systems have incorporated users' relevance feedback to modify the retrieval process in order to generate perceptually and semantically more meaningful retrieval results. The works shown in [2] was mixture of color, texture, and edge density for MPEG-7 standards and where as in [4] the edge histogram was used. A similar kind of approach was done in [5], [13] based on edge density for detecting people in images. In [7], [9], color and texture features were used for image retrieval.

Considerable amount of work had already been done for medical images. For these types of images, texture is highly preferred [10], [17]. To make image retrieval faster, several indexing structures were designed. The most popular ones are 2D-S Tree, Graph-based, containment tree, fuzzy-based, relationship tree, etc.

III. Proposed CBIR Model

The proposed CBIR framework is shown in Figure 1. The images are kept in a database called Image Database. After preprocessing, images are segmented by using the method described in [9].

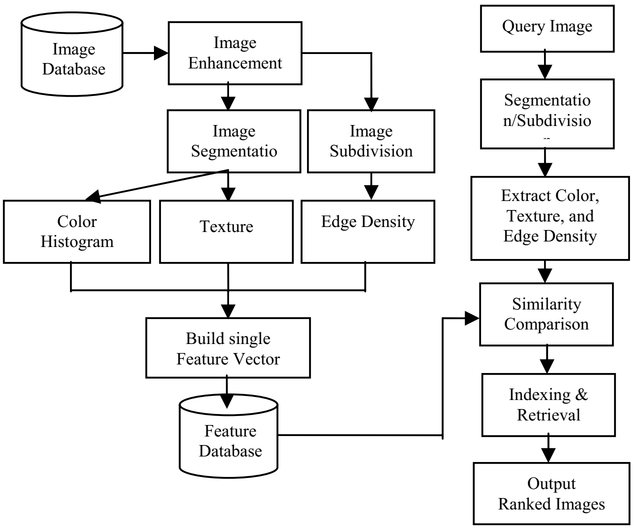


Fig. 1 Proposed CBIR Framework

Only the dominant segments are considered for feature extraction namely color histogram features, texture features, and image density features (explained in the subsequent sections). Then a single feature vector is constructed and stored in the feature database. When a query image is submitted by the user, the same work is done as explained above to get its feature vector. For similarity comparison between the query image and the database image, the Euclidean distance method is used. Using an appropriate threshold, images that are semantically closer are retrieved from the database and displayed as a thumbnail.

IV. Color

In image retrieval systems color histogram is the most commonly used feature. The main reason is that it is independent of image size and orientation. Also it is one of the most straight-forward features utilized by humans for visual recognition and discrimination. Statistically, it denotes the joint probability of the intensities of the three color channels. Once the image is segmented, from each region the color histogram is extracted. The major statistical data that are extracted are histogram mean, standard deviation, and median for each color channel i.e. Red, Green, and Blue. So totally 3 × 3 = 9 features per segment are obtained. All the segments need not be considered, but only segments that are dominant may be considered, because this would speed up the calculation and may not significantly affect the end result.

V. Texture

There is no precise definition for texture. However, one can define texture as the visual patterns that have properties of homogeneity that do not result from the presence of only a single color or intensity. Texture determination is ideally suited for medical image retrievals [17]. In this work, computation of gray level cooccurrence matrix is done and from which a number of statistical measures are derived.

The autocorrelation function of an image is used to quantify the regularity and the coarseness of a texture. This function is defined for an image *I* as:

$$\rho(x,y) = \frac{\sum_{u=0}^{N}\sum_{v=0}^{N} I(u,v)\,I(u+x),(v+y)}{\sum_{u=0}^{N}\sum_{v=0}^{N} I^2(u,v)} \tag{1}$$

A texture is characterized by a set of values called energy, entropy, contrast, and homogeneity. The following formulas are used to calculate the features and are shown in equations 2 to 5 [18].

$$\text{Energy} = \sum_i \sum_j P_d^2(i,j) \tag{2}$$

$$\text{Entropy} = - \sum_i \sum_j P_d(i,j) \log P_d(i,j) \tag{3}$$

$$\text{Contrast} = \sum_i \sum_j (i-j)^2 P_d(i,j) \tag{4}$$

$$\text{Homogeneity} = \sum_i \sum_j \frac{P_d(i,j)}{1+|i-j|} \tag{5}$$

The performance of the texture features are tested using test images from Corel image database just like color.

VI. Edge Density

A novel approach in the field of image retrieval is use of edge information. The edge histogram is normally used in the area of computer vision primarily in tracking of moving objects [14]. Edges convey essential information to a picture, and their accurate detection is of primary importance. The identification of edges inside one image is the first step to recognize geometric shapes within one image [16].

A. Edge Histogram Descriptor (EDH)

The Edge Histogram Descriptor represents the local edge distribution in the image which is obtained by subdividing the whole image into 4 × 4 sub images. For each of these sub images we compute the histogram. This means a total of 16 × 5 = 80 bins are required. The histograms are categorized into four directional edges called vertical, horizontal, 45 degree, 135 degree, and one non-directional edge. To detect the edge strength, filter coefficients shown in Figure 2 were applied. Edge blocks that are greater than a given threshold is selected [4].

1	-1
1	-1

1	1
-1	-1

$\sqrt{2}$	0
0	$-\sqrt{2}$

2	-2
-2	2

Fig. 2 Filter Coefficients

For each sub image the edge density can be calculated using equation (6). Let (x1, y1) and (x2, y2) are the top left corner and the bottom right corner of the sub image. Then the edge density f is given by,

$$f = \frac{1}{a_r} \sum_{x=x1}^{x2} \sum_{y=y1}^{y2} e(x,y) \tag{6}$$

where a_r is the region area. All these features are put in the feature vector table.

B. Similarity Comparison and Greedy Method

For similarity comparison, we have used Euclidean distance, d using equation 7.

$$d = \sqrt{\sum_{i=1}^{N} (F_Q[i] - F_{DB}[i])^2} \tag{7}$$

where $F_Q[i]$ is the i^{th} query image feature, and $F_{DB}[i]$ is the corresponding feature in the feature vector database. Here, N refers to the number of images in the database.

The main issue in image retrieval systems is the number of dimensions of the feature vector which is normally huge. For example QBIC system reduces the 20-dimension feature vector to two or three using Principle Component Analysis (PCA) [19]. It explores exponentially with the increasing of the dimensionality and eventually reduces to sequential searching. To overcome these problems a simple method called greedy strategy is used.

Consider three database images and their corresponding segments as $I_1(S_1, S_2, S_4)$, $I_2(S_2, S_5, S_8, S_7)$, and $I_3(S_1)$. The sequence of the segments shown in I_1, I_2, and I_3 are based on descending order of the size/area of each segment. Similarly, let $Q_I(S_7, S_2)$ denotes the segments of the query image. The algorithm shown in Figure 3 uses the greedy strategy to compare the similarity between the query image and the database images.

```
Algorithm ImageSimilarity
  // $I[N]$ – Image DB with $N$ images
  // $Q_I$ – Query Image
  foreach (Image $I$ in $I[N]$)
    foreach (Segment $s$ in $SegmentSet$)
      if (Euclidean($Q_I$[s], $I[s]$) < $threshold$)
        // continue to check other segments
      else
        // no need to check other segments
end.
```

Fig. 3 Algorithm for Similarity comparison based on greedy strategy

Suppose if we fix 20 features for each segment and if there are five segments on an average per image, then we must repeat the comparison for each segment. With this proposed method we obtain a reasonable increase in performance when the number of segments is more.

VII. EXPERIMENTAL SETUP AND RESULTS

A Dell Precision Pentium Core2 Duo Workstation with 2GB RAM computer is used for conducting the experiments. The main software tools used were Visual Studio 2005, C# .NET Framework for developing UI components, building the logic, etc. For the image processing work, the open source products like AForge.Imaging and AForge.Math from Google (http://code.google.com/p/aforge/) were used.

To store the images and the feature vector, Oracle 10g database was selected for various reasons. Oracle Multimedia (formerly Oracle *inter*Media) is a feature that enables Oracle Database to store, manage, and retrieve images, audio, video, or other heterogeneous media data in an integrated fashion with other enterprise information. Corel Image database with 1000 natural images were used for testing the proposed CBIR system.

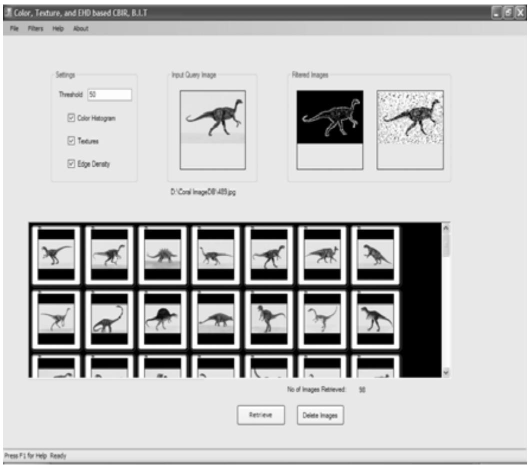
A. Retrieval Efficiency

The retrieval efficiency, namely recall and precision were calculated using 1000 natural color images (100 in each category) from Corel image database. Figure 4 shows the screenshot of the framework.

Standard formulas have been used to compute the precision and recall for four query images (See Figure 5).

$$precision = \frac{No.of\ relevant\ images\ retrieved}{Total\ No.of\ images\ retrieved}$$

$$recall = \frac{No.of\ relevant\ images\ retrieved}{Total\ No.of\ relevant\ images\ in\ the\ Database}$$



Threshold 50

☑ Color Histogram

☑ Textures

☑ Edge Density

D:\Coral ImageDB\489.jpg

No of Images Retrieved: 50

Press F1 for Help Ready

Fig. 4 Screenshot of the CBIR Model

By randomly selecting four query images from the CorelImage Database, the system was tested and the results are shown in Table I.

TABLE I
PRECISION AND RECALL VALUES IN %

Query Image	Color	Texture	EHD	All
1	21.8	50.0	23.6	35.2
	28.0	15.0	34.1	60.0
2	100.0	75.0	87.0	100.0
	98.0	62.0	68.0	78.0
3	74.6	20.0	65.0	42.8
	59.0	10.0	37.0	90.0
4	91.7	75.0	85.6	92.0
	24.0	33.0	34.9	28.0

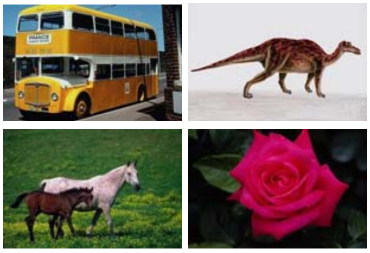


Fig. 5 Query Images 1 to 4 (top left to bottom right)

In Table I, the first line in each query image indicates precision and the second line indicates recall. Figure 5 shows the query images used in conducting the experiment.

VIII. CONCLUSION AND FUTURE WORK

This paper proposed a universal model for the Content Based Image Retrieval System by combining the color, texture, and edge density features or individually. Users were given options to select the appropriate feature extraction method for best results. The advantages of global and local features together have been utilized for better retrieval efficiency. The results are quite good for most of the query images and it is possible to further improve by fine tuning the threshold and adding relevance feedback.

REFERENCES

[1] Tristan Glatard, John Montagnat, “Texture based Medical image indexing and retrieval: application to cardiac images".

[2] B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, and Akio Yamada, "Color and Texture Descriptors". In: IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 6, June 2001, pp. 70-715.

[3] Zhe-Ming Lu1, Su-Zhi Li, and Hans Burkhardt, "A Content-Based Image Retrieval Scheme in JPEG Compressed Domain", International Journal of Innovative Computing, Information and Control ICIC International 2006 ISSN 1349-4198, Volume 2, Number 4, August 2006, pp. 831-839.

[4] Minyoung Eom, and Yoonsik Choe, "Fast Extraction of Edge Histogram in DCT Domain based on MPEG7", Proceedings of World Academy of Science, Engineering and Technology Volume 9 November 2005 ISSN 1307-6884, pp. 209-212.

[5] Son Lam Phung and Abdesselam Bouzerdoum, "A New Image Feature for Fast Detection of People in Images", International Journal of 2007 Institute for Scientific Information and Systems Sciences Computing and Information Volume 3, Number 3, pp. 383-391.

[6] Paul Stefan, et al.: Segmentation of Natural Images for CBIR.C. J. Kaufman, Rocky Mountain Research Lab., Boulder, CO, private communication, May 1995.

[7] P. S. Hiremath , Jagadeesh Pujari, "Content Based Image Retrieval using Color, Texture and Shape features", 15th International Conference on Advanced Computing and Communications, IEEE Computer Society 2007, pp. 780-784.

[8] Remco C. Veltcamp, Mirela Tanse, "Content Based Image Retrieval Systems". A Survey, Technical Report UU-CS-2000-34, October 2000, pp. 1-62.

[9] Mustafa Ozden and Ediz Polat, "Image Segmentation using Color and Texture features".

[10] John Montagnat, et al, "Texture-based Medical Image Indexing and Retrieval on Grids", Medical Imaging technology, vol 25 No. 5 Nov 2007. pp. 333-338.

[11] C. R. Shyu, et. al, "Local versus Global Features for Content-Based Image Retrieval", IEEE Workshop on Content-Based Access of Image and Video Libraries, 1998.

[12] Roger Weber and Michael Mlivoncic, "Efficient Region-Based Image Retrieval", ACM CIKM '03 November 3-8, 2003, USA.

[13] S. L. Phung and A. Bouzerdoum, "Detecting People in Images: An Edge Density Approach", IEEE, ICASSP 2007. pp. 1229-1232.

[14] Bohyung Han, Changjiang Yang, et al, "Bayesian Filtering and Integral Image for Visual Tracking".

[15] Vincent Arvis, Christophe Debain, et. al, "Generalization of the Cooccurrence Matrix for Color Images Application to Color Texture Classification", Image AnalStereol 2004, pp. 63-72.

[16] Alberto Amato, Vincenzo Di Lecce, "Edge Detection Techniques in Image Retrieval: The Semantic Meaning of Edge", 4th EURASIP Conference on Video/Image Processing and Multimedia Communications, Zagreb, Croatia. pp. 143-148.

[17] Thomas M. Lehmann, et al, "Automatic categorization of medical images for content-based retrieval and data mining", Computerized Medical Imaging and Graphics. Elsevier 2004. pp. 143-155.

[18] Dong Yin, Jia Pan, et al, "Medical Image Categorization based on Gaussian Mixture Model", IEEE 2008 International Conference on BioMedical Engineering and Informatics, pp. 128-131.

[19] Dr. Fuhui Long, Dr. Hongjiang Zhang and Prof. David Dagan Feng, "Fundamentals of Content-Based Image Retrieval" - http://research.microsoft.com/asia/dload_files/group/mcomputing/2003P/ch01_Long_v40-proof.pdf