ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
National Conference on Emerging Trends in Computer Science & Information Technology (NCETCSIT-2011)

Content Based Image Retrieval Using Texture and Color Extraction and Binary Tree Structure

Mrs. Saroj Shambharkar, Ms. Shubhangi C. Tirpude

***Abstract*— Content Based Image Retrieval is important research field in many applications. In this paper the CBIR system is implementing a *new binary tree feature* is used .Color and texture are commonly used in most of the CBIR system for finding similar images from the database to a given query image.**

In the implemented system color and texture are used as *basic features* to describe all the images. In addition, a binary tree structure is used to describe higher level features of an image. To extract color information, two histograms i.e. hue and saturation of the image are used. And to extract texture information image quantization and wavelet decomposition is applied to each image blocks. The Hue is quantized into 360 levels and the saturation into 100 levels.

The binary tree structure is implemented based on some steps provided in [1].In this system, the feature extraction and wavelet decomposition for texture extraction is used to compute the feature vectors of image which helps in retrieval process. This approach combines the color and texture features and binary partitioning tree method in order to find the images similar to a specific query image. The Minkowski difference equation is used to measure the distance. The proposed system is implemented using the Matlab software.

***Index Terms*— Binary partitioning tree, Content based image retrieval, Feature extraction, Minkowski difference, Quantization, Wavelet decomposition**

I. Introduction

There are different methods of image retrieval where the meta-data is associated with the image, commonly called as keywords. Content Based Image Retrieval system is the system of retrieving images from the database based on the similarity measured between images in the database and query image. The features can be in the form of keywords to describe the image, or the visual features such as color, texture, shape etc.

Today people are using internet that consists of millions of website and it deals with huge database, where most of the sites contains images. And to organize and classify such huge amount of images is time consuming. Therefore there is a need of system which should able to organize and classify the images in a database system, and this is achieved by using CBIR system.

A. CBIR Overall Structure

The overall structure of a typical image retrieval system is as shown in fig 1. is divided into seven parts:

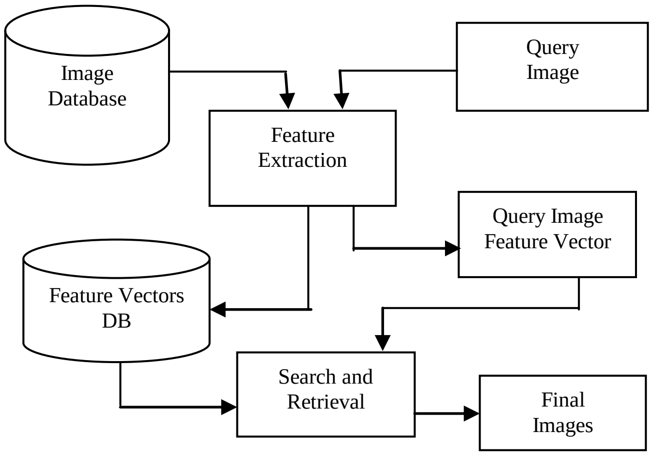

Fig 1. General Structure of CBIR System

- Image Database - The Image database consisting of hundreds, thousands or millions of images among which one of the image can be used as a query image to be search or any image can also be considered.
- Feature Extraction - It retrieves features of every image considered by the user and sends them further for appropriate process.
- Feature Vectors Database - This Database stores the extracted features received from feature extraction process.
- Query Image - The image given by the user which is compared with other images of the database.
- Query Image FV - Feature vector of the query image computed by feature extraction.
- Search and Retrieval - This part is used for searching the query image and finding the images similar to the query image, helps in retrieval.
- Final Images - The images are retrieved and arranged according to their matching score.

II. Application of CBIR System

A. Uses of CBIR

- *Crime Prevention*

For example Law enforcement agencies typically maintain large archives of visual evidence, including past suspects facial photographs, fingerprints, shoeprints.

Whenever a serious crime is committed, they can compare evidence from the scene of the crime for its similarity to records in their archives. This is an example of identify rather than similarity matching though since all such images vary over time. The CBIR is capable of searching an entire database to find the closest matching records.

- *Fashion and Interior Design*

The designer has to work within externally imposed constraints, such as choice of materials. The ability to search a collection of fabrics to find a particular combination of color and texture is increasingly being recognized as a useful aid in the design process. There are general purpose CBIR packages for specific tasks such as color matching of items from electronic versions of mail order catalogues, and identifying textile samples bearing a desired pattern.

- *Journalism and Advertising*

Both newspapers and stock agencies maintain archives of still photographs to illustrate articles or advertising copy. These archives can often be extremely large, and very expensive to maintain if detailed keyword indexing is provided. Broadcasting corporations faced with even bigger problem that they have to deal with millions of hours of archive videos footage. The CBIR technology provides efficient and effective retrieval of still images from photo libraries, eliminating or at least substantially reducing the need for manual keyword indexing.

- *Medical Diagnosis*

The increasing reliance of modern medicine on diagnostic techniques such as radiology, histopathology etc. has resulted in an explosion in the number and importance of medical imaging. The prime requirement for medical imaging systems are to be able to display images relating to a named patient. Therefore there is increasing interest in the use of CBIR technique to aid diagnosis by identifying similar past cases.

- *Home Entertainment*

Much home entertainment is image or video-based, including holiday snapshots, home videos and scenes from favorite TV programs or films. The CBIR technology could include management of family photo albums or clips from commercial films.

- *Education and Training*

It is often difficult to identify good teaching material to illustrate key points in a lecture or self-study module. The availability of searchable collections of video clips providing examples of avalanches for a lecture on mountain safety, or traffic congestion for a course on urban planning, could reduce preparation time and lead to improved teaching quality. In some cases such videos might even replace a human tutor.

III. Features Used By Existing CBIR System

Most of the existing CBIR systems are based on visual features of images such as color, texture and shape to describe the image [1].

A. Color

Several methods for retrieving images on the basis of Color similarities are available, but most are variations on the same basic idea. Each image added to the collection is analyzed to compute a color histogram which shows the proportion of pixels of each color within the image. The color histogram for each image is computed and then stored in the database. While searching any image, the user can either specify the desired proportion of each color, or submit an example or query image from which a color histogram is calculated. The matching process retrieves those images whose color histograms match with those of the query image.

B. Texture

The ability to retrieve images on the basis of texture similarity may not seem very useful. But the ability to match on texture similarity can often be useful in differentiating the regions of various images with similar color. Sometimes the color may seems to be same like color of leaves and grass as green, but the computed values makes them different from each other. The variety of techniques are used, the best is based on comparing values of what are known as feature vectors calculated for query and stored images. This helps in calculating the relative brightness of selected pairs of pixels from each image of say leaves and grass. From these it is possible to calculate measures of image texture such as the degree of contrast, coarseness, directionality and regularity. An alternative method of texture analysis for retrieval includes the use of Gabor filters and fractals. By selecting examples of desired textures from a palette, or by supplying an example query image, the system retrieves images with texture measures most similar in value to the query. In the proposed approach the wavelet decomposition is used for texture extraction.

C. Shape

The considerable evidence for the natural objects is primarily recognized by their shape. A number of features characteristic of object shape (but independent of size or orientation) are computed for every object within each stored image. Queries are then answered by computing the same set of features for the query image, and retrieving those stored images whose features are most closely match those of the query. Two main types of shape features commonly used are global features and local features. The global features are aspect ratio, circularity and moment invariants and local features are sets of consecutive boundary segments.

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
National Conference on Emerging Trends in Computer Science & Information Technology (NCETCSIT-2011)

IV. Proposed System

In this section the design and implementation of CBIR system using feature extraction and search retrieval process for image database is discussed. The features such as texture, color of the image are used and the partitioning binary tree approach for feature extraction is used.

A. Feature Extraction

To compare similarity between images the numerical value is needed. To perform this comparison the feature vectors available in the form of numbers are used. In this regard feature extraction is a way to show visual information of an image in scale of numbers so that they can be analogous.

a. Color Extraction

Various color spaces such as RGB, HSV, HIS, NTSC, YCbCr etc. can be used by CBIR System ,depending on the type of application and user's choice. However no color space is dominant in all applications. In the proposed system, the HSV color space is used shown in Fig 2. The approach used is to extract two histograms, one for Hue and one for Saturation. Since V is directly related to brightness level, therefore it is not considered in color measurement approach.

b. Texture Extraction

For texture extraction, Wavelet decomposition is used is shown in Fig 3. The JPEG format images are used, and convert the RGB image into gray scale image which is divided into blocks. By applying Wavelet decomposition on the gray-level image, four sub images were produced.

- A low resolution copy of original image.
- Horizontal Component.
- Vertical Component.
- Diagonal Component.

The first component is called as the approximation of original image and last three components are said to be the detailed component of the original image.

The following steps were used in the texture extraction process of the image:

- The image is divided into 8 x 8 equivalents to 64 regions.
- After obtaining the regions of an image the Wavelet decomposition transformation on each region is applied. In the proposed system the Haar wavelet decomposition is used.
- The mean and variance of four sub images corresponds to each region is calculated and concatenating them.
- Then from the obtained mean and variance of each region, all mean and variance feature vectors of every region of the image is concatenated.

B. Binary Tree Construction

Binary partition tree is a structure used to represent the regions of an image. A binary tree is used as a base for each region of each image in the database. In the binary partitioning tree given in Fig.4 nodes 1, 2, 3, 4, 5 are leaves which represent regions belonging to the initial partition.

The leaf nodes 1 and 2 are first merged and create a new node 6, then this node is merged with leaf node 3, which creates node 7, this process is repeated until the binary partition tree is obtained. The root node corresponds to the entire image.

To construct a binary tree, the algorithm starts from an arbitrary region which considered as the first node and then selecting a neighbor region as its sibling, these nodes are added as children of their parent. We start with node 1, and then choose a neighbor i.e. 2. These two nodes are children of parent node 6.This process are repeated until all regions have been added to the binary partitioning tree shown in fig.5.

To have more precise measure, each image in the database is divided into equal fixed-sized squared blocks. A distinct tree should be created for each block. For each node of the constructed binary partitioning tree, calculate the mean color and area of its corresponding region using the available functions in Mat lab. These values i.e. mean color and mean area for all nodes are concatenated to construct a feature vector representing one block. The process of concatenating feature vectors of a block is repeated for all blocks to construct a feature vector for entire image.

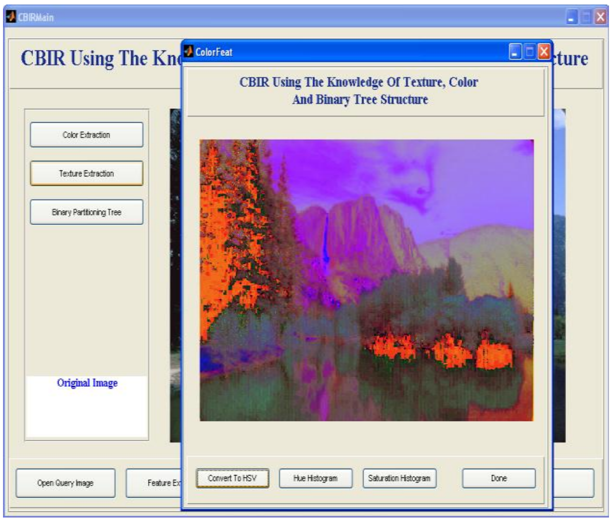

Fig 2. Conversion of RGB image into HSV

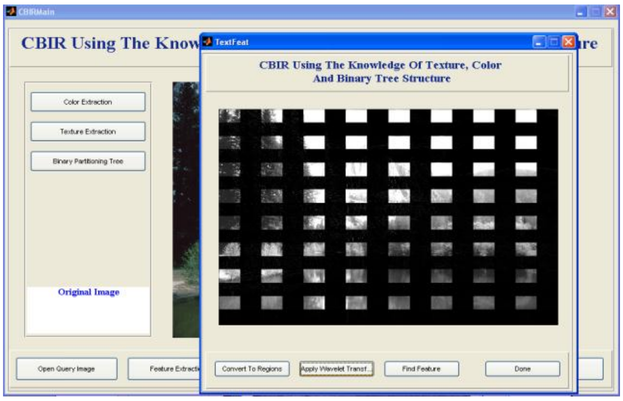

Fig 3. Wavelet Decomposition

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
National Conference on Emerging Trends in Computer Science & Information Technology (NCETCSIT-2011)

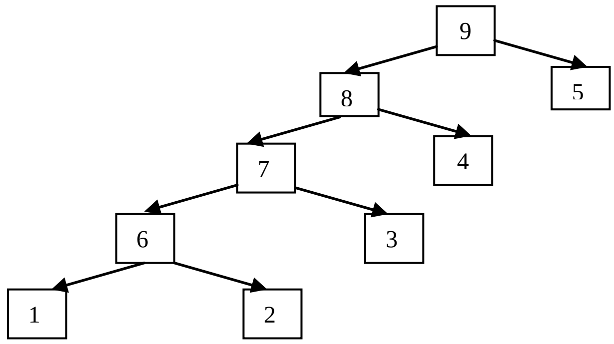

Fig 4. Complete Binary Partitioning Tree

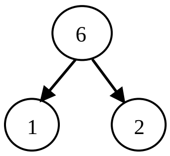

Fig 5. Part of binary tree

C. Image Partioning

There are various ways to partition an image into separate regions. In this we have used color homogeneity, which is based on the assumption that an object is composed of a group of similar colors. To do this, a color palette including less number of colors than the available in real color space is required to quantize an image.

D. Quantization

To quantize the image using fewer colors than actually present safe color cube is used. The safe color cube covers the entire RGB color space. Safe color cube consist of 216 colors in RGB mode. Each R, G and B can only be 0, 51, 102, 153, 204 or 255.Thus RGB triples of these six values can give the number of possible colors as $(6)^3$ which is equals to 216 colors.

To quantize an image, mean color is quantized into safe color palette. Therefore, by having such images each with 216 specific colors, we are expecting to have distinctive regions with homogeneous color.

Quantization in terms of color histograms refers to the process of reducing the number of bins by taking colors that are very similar to each other and putting them in the same bin. By default the maximum number of bins one can obtain using histogram function in Matlab is 256. For the purpose of saving time, we can quantize the number of bins. Obviously quantization reduces the information regarding the content of images but as mentioned this is the tradeoff when we want to reduce the processing time.

E. Design of Implemented System

The image is read from any available source and forms an image database. Now the feature extraction process is applied to each image of the database,. This process consists of color, texture extraction and binary tree structure construction. As a result of it feature vectors of each image is created.

Next the query image is taken and same feature extraction process is applied and feature vector is obtained. The distance between feature vectors of images and feature vector of query image is calculated using Minkowski difference equation is. Finally the images are ranked and retrieved according to their rank value. The System flow diagram is shown in Fig. 6.

F. Search and Retrieval

The search and retrieval process is demonstrated in Fig.7. The distance for each image with query image is calculated an arranged either in ascending or descending order of their matching score value. The search process is to get feature vectors of an input image called and retrieve the images with similar feature vector. This search strategy is called nearest-neighborhood.

In image retrieval systems color histogram is the most commonly used feature. The hue and saturation histogram are computed for the color histograms .The main reason is that it is independent of image size and orientation. Also it is one of the most straightforward features utilized for visual recognition and discrimination.

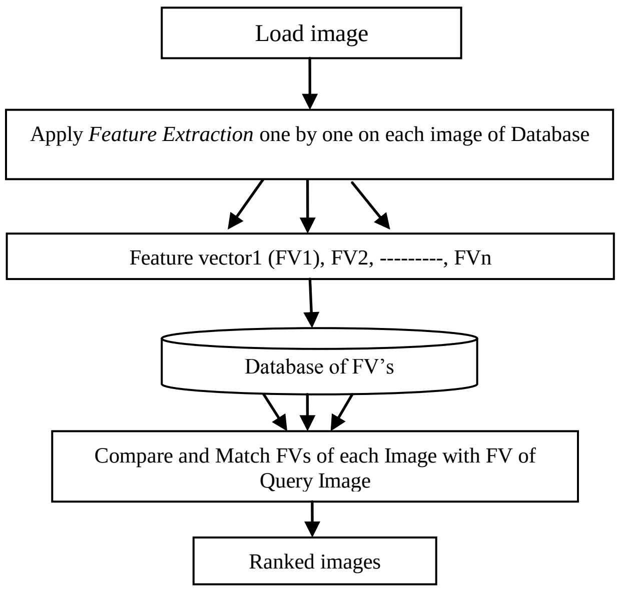

Fig 6. Block diagram of implemented CBIR System.

a. Distance Measure

The Minkowski difference equation is used to calculate feature vector distance, which is given in equation (1):

$$DM^{rho}(k,l) = (\sum_{i=1}^{n}(|k_i - l_i)^{1/rho})) \qquad (1)$$

Where,
DM denotes Minkowski Difference
i is used to denote the image number
n for total number of images
k and l are two n dimensional feature vectors related to query and images from the database.

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
National Conference on Emerging Trends in Computer Science & Information Technology (NCETCSIT-2011)

The value of rho can be 1 or 2, rho=1 is used for color histogram and area of binary tree, rho=2 is used for texture.

The table I shows the feature properties and their coefficients which are considered in the implementation of CBIR system.

b. Procedure of Ranking the images

The feature distances are summed in order to obtain the final distance. The prerequisite for this operation is normalization in each feature level. That is done by dividing all distance values of specific feature to maximum gained distance. Add all difference values which give a measure of how an image is different from the query image. The most k high-rank images are selected for display to the user.

To obtain the final rank of the image following equation used:

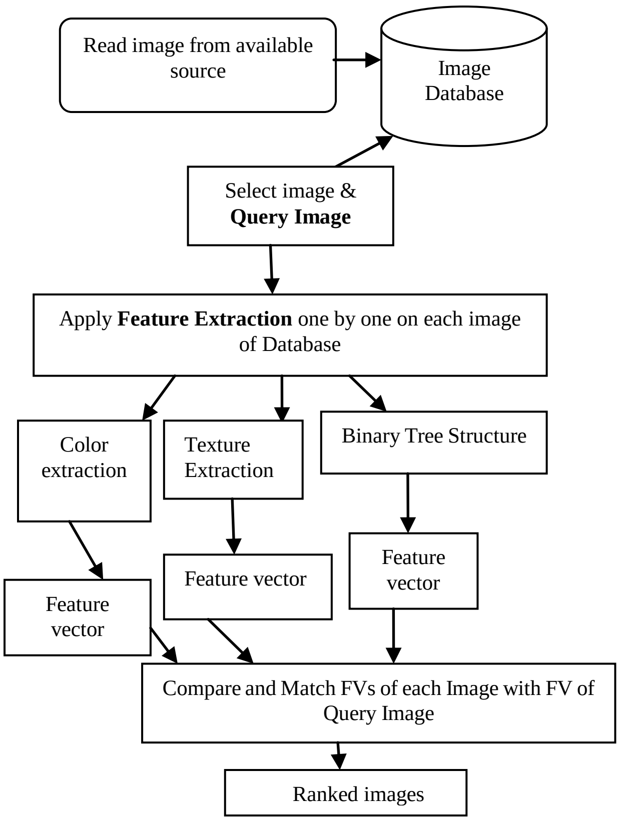

Fig. 7. Search and retrieval process

For color extraction

$$D_{S_{i,color}} = \frac{D_{i,color}}{Max(D_{i,col}} \quad (2)$$

$$S_{i,color} = \frac{1}{D_{S_{i,color}}} \quad (3)$$

$$T_{i,color} = 68 * S_{i,color} \quad (4)$$

For binary tree construction

$$D_{S_{i,bin}} = \frac{D_{i,bin}}{Max(D_{i,bin})} \quad (5)$$

$$S_{i,bin} = \frac{1}{D_{S_{i,bin}}} \quad (6)$$

$$T_{i,bin} = 3 * S_{i,bin} \quad (7)$$

For texture extraction

$$D_{S_{i,text}} = \frac{D_{i,text}}{Max(D_{i,text})} \quad (8)$$

$$S_{i,text} = \frac{1}{D_{S_{i,text}}} \quad (9)$$

$$T_{i,text} = 6 * S_{i,text} \quad (10)$$

Therefore, based on the values obtained from the above equation, the rank of each image is calculated as per equation given below:-

$$Rank = D_{S_{i,color}} + D_{S_{i,bin}} + D_{S_{i,text}} \quad (11)$$

Fig 8.Retrieved Images and their respective matching score

TABLE
FEATURES CONSIDERED IN THE SYSTEM AND ITS IMPORTANCE

	Features considered	Dimension considered for Feature Vector corresponding to each feature	Weightage given to each feature(R_i)
1	Color	360-bin Hue Hist	0.68
2		100-bin Sat Hist	0.14
3	Texture	Wavelet Mean	0.06
4		Wavelet Var	0.06
5	Binary Partitioning Tree	Color Mean	0.03

ISSN 2249-6343
International Journal of Computer Technology and Electronics Engineering (IJCTEE)
National Conference on Emerging Trends in Computer Science & Information Technology (NCETCSIT-2011)

V. Result and Conclusion

The proposed CBIR System, where the images gone through various transformations such as the input RGB image converted to gray scale image, wavelet transformation, and processes such as color extraction, Texture extraction and binary tree construction to obtain feature vectors.

The image matching process is performed on the basis of these feature vectors and the relevant images are retrieved from the database according to highest rank or score. It is clear in Fig. 8 that the image with matching score 0 is exactly matched with the query image. So we can conclude that the CBIR system retrieves the images and if its score is 0 then it is totally similar with the query image. The last image shown in result having the maximum distance as compared to other of the images is not considered to be closer to query image. This shows that the retrieved image having larger value is less similar and image having small value is more similar and image having 0 matching score is exactly same as query image.

VI. Future Work

The success of the proposed CBIR system is that it improves the performance of the system. In this paper the color, texture and binary tree structure is used as features to achieve the performance of the system. By using this proposed system the retrieved images are more relevant against the query image.

This work can be further extended by including some other feature along with the features used in the proposed system, to describe the image.

By adding more features, the performance of the system will be improved. Also the accuracy of obtaining the retrieved images from large database or web will result into increase in the retrieval efficiency of the system.

References

[1] Zahra Mansoori, Mansour Jamzad, (2009), "Content based image retrieval using the knowledge of texture, color and binary tree structure", IEEE.

[2] S.Nandagopalan, Dr.B.S.Adiga, N.Deepak, (2009), "A Universal Model for Content-Based Image Retrieval", International Journal of Computer Science.

[3] P.S.Suhasini, Dr.K.Sri Rama Krishna, Dr.I.V.Murali Krishna "CBIR Using Color Histogram Processing", Journal of Theoretical and Applied Information Technology, 2005-2009.

[4] Johannes Imo, Sebastian Klenk and Gunther Heidemann, (2008), "Interactive Feature Visualization for Image Retrieval", IEEE.

[5] I.Stojanavic, S.Bogdanova and M. Bogdanav, "Content-Based Image Retrieving Improved by Pixel-Based Search", IEEE.

[6] Content-based image retrieval systems, IEEE Computer, 28(9), 1995.

[7] Andre Folkers and Hanan Samet, "Content-Based Image Retrieval Using Fourier Descriptors on Logo database", 2002.

[8] P.S. Hiremath and Jagadesh Pujari, "Content-Based Image Retrieval based on color, texture and shape features using image and its complement", 2007.

[9] Kein-Ping Chung, Jia Bin Li, Chun Che Fung and Kok Wai Wong, "A Parallel Architecture for feature Extraction in Content-Based Image Retrieval System", 2004.

[10] D.N.F.Awang Iskandar, James A.Thom, S.M.M.Tahaghoghi, (2008), "Content-based Image Retrieval Using Image Regions as Query Examples", IEEE.