IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 1, September 2012
ISSN (Online): 1694-0814
www.IJCSI.org

Content-Based Image Retrieval using Color Moment and Gabor Texture Feature

S. Mangijao Singh , K. Hemachandran

**Department of Computer Science, Assam University,
Silchar, Assam, India**

Abstract

Content based image retrieval (CBIR) has become one of the most active research areas in the past few years. Many indexing techniques are based on global feature distributions. However, these global distributions have limited discriminating power because they are unable to capture local image information. In this paper, we propose a content-based image retrieval method which combines color and texture features. To improve the discriminating power of color indexing techniques, we encode a minimal amount of spatial information in the color index. As its color features, an image is divided horizontally into three equal non-overlapping regions. From each region in the image, we extract the first three moments of the color distribution, from each color channel and store them in the index i.e., for a HSV color space, we store 27 floating point numbers per image. As its texture feature, Gabor texture descriptors are adopted. We assign weights to each feature respectively and calculate the similarity with combined features of color and texture using Canberra distance as similarity measure. Experimental results show that the proposed method has higher retrieval accuracy than other conventional methods combining color moments and texture features based on global features approach.

Keywords: *CBIR, color moment, Canberra distance, Gabor wavelet.*

1. Introduction

Content-based image retrieval (CBIR) [1] has become a prominent research topic because of the proliferation of video and image data in digital form. The increased bandwidth availability to access the internet in the near future will allow the users to search for and browse through video and image databases located at remote sites. Therefore, fast retrieval of images from large databases is an important problem that needs to be addressed. High retrieval efficiency and less computational complexity are the desired characteristics of CBIR systems. In conventional image databases, images are text-annotated and image retrieval is based on keyword searching. Some of the disadvantages of this approach are: 1. keyword-based image retrieval is not appropriate because there is no fixed set of words that describes the image content; 2. keyword annotation is very subjective. To avoid manual annotation, an alternative approach is content-based image retrieval (CBIR), by which images would be indexed by their visual content such as color, texture, shape etc. and the desired images are retrieved from a large collection, on the basis of features that can be automatically extracted from the images themselves [2]. Considerable research work has been done to extract these low level image features [3 and 4], evaluate distance metrics, and look for efficient searching schemes [5 and 6]. Basically, most CBIR systems work in the same way: A feature vector is extracted from each image in the database and the set of all feature vectors is organized as a database index. At query time, a feature vector is extracted from the query image and it is matched against the feature vectors in the index. The crucial difference between the various systems lies in the features that they extract and in the algorithms that are used to compare feature vectors. The block diagram of basic CBIR system is shown in Fig. 1.

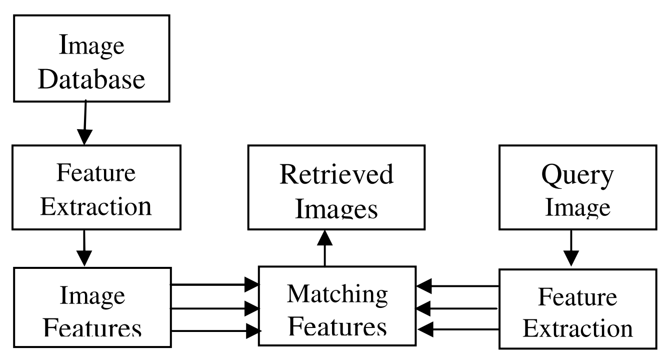


Fig. 1 Block diagram of CBIR

Color, texture, local shape and spatial information in a variety of forms are the most widely used features in such systems [7]–[10]. Because of high demand for searching image databases of ever-growing size, CBIR is becoming very popular. Since speed and precision are important, we need to develop a system for retrieving images that is efficient.

The color features are the most widely used visual features in image retrieval because they are easier to extract compared with texture and shape information. Color feature is relatively robust to background complication and independent of image size and orientation. Statistically, it denotes the joint probability of the intensities of the three color channels. The color indexing work of Swain and Ballard [11], which is based on color histograms, has demonstrated the potential of using color for indexing. Stricker and Orengo [12] have shown that moment based color distribution features can be matched more robustly than color histograms as histograms do not capture spatial relationship of color regions and thus, they have limited discriminating power. The system presented in [13] integrates mostly the algorithms introduced in [11] into a database environment. Reasonable results can be achieved using the above mentioned algorithms, but it is clear that the false positives which are retrieved result from the lack of spatial information in the index. The simplest way to store spatial information in the index is to divide the image into sub-images and then extract the color features for each sub-image. Color correlogram and color coherence vector can combine the spatial correlation of color regions as well as the global distribution of local spatial correlation of colors. These techniques perform better than traditional color histograms when used for content-based image retrieval. However, they require very expensive computation. Color moments have been successfully used in content based image retrieval systems. It has been shown [12] that characterizing one dimensional color distributions with the first three moments is more robust and runs faster than the histogram based methods. Hence, in our proposed method, Color moments are used for extraction of color features.

Texture is an important feature of natural images. A variety of techniques have been developed for measuring texture similarity. Most techniques rely on comparing values of what are known as second-order statistics calculated from query and stored images [14]. These methods calculate measures of image texture such as the degree of contrast, coarseness, directionality and regularity [15 and 16]; or periodicity, directionality and randomness [17]. Alternative methods of texture analysis for image retrieval include the use of Gabor filters [3] and fractals [18].

Gabor filter (or Gabor wavelet) is widely adopted to extract texture features from the images for image retrieval [3, 19, 20, 21, 22, 23], and has been shown to be very efficient. Manjunath and Ma [3] have shown that image retrieval using Gabor features outperforms that using Pyramid-structured wavelet transform (PWT) features, tree-structured wavelet transform(TWT) features and multiresolution simultaneous autoregressive model (MR-SAR) features. Hence, in our proposed method, Gabor filter is used for extraction of texture features.

A color retrieval method based on the primitives of color moments is proposed in [24]. After dividing an image into several blocks, the color moments of all blocks are extracted and clustered into several classes based on a fast non-iterative clustering algorithm. The mean vector of each class is considered as a primitive of the image and all primitives are used as feature vectors. Two test databases from Corel were used and compared the performances of the proposed method with other existing ones. The experimental results showed that the proposed method is usually better than others.

Choras et al. [25] proposed an integrated color, texture and shape feature extraction method in which Gabor filtration is used for determining the number of regions of interest (ROIs). They calculated texture and color features from the ROIs based on threshold Gabor features and histograms, color moments in YUV space, and shape features based on Zernike moments. The features presented proved to be efficient in determining similarity between images.

Xue and Wanjun [26] proposed a method in which the feature extraction methods of color histograms and color moments are integrated to extract the color feature. They reported that the recall and precision had been improved and the index sorting was better.

A method based on color and texture features is proposed in [27]. As color feature, they use color moments of the Hue, Saturation and Value (HSV) of the image and Gabor descriptors are adopted as texture features. They assigned weights to each feature and calculated the similarity with combined features of color and texture using normalized Euclidean distance. They reported that the proposed method has higher retrieval accuracy than the conventional method using color and texture features. The feature dimension is also lower than the conventional methods.

It normally requires complicated segmentation of the object from the background when users specify the query "content" or "objects" of their interest and only wish to retrieve images containing relevant objects, while ignoring irrelevant image areas (such as the background). Kumar et al. [28] proposed a model in which the user can select "object of user's interest" of different shapes, non homogenous texture containing different colors, regardless of many objects present in the same image using varied tools like polygonal, rectangle, circle selector tools. A two state procedure is used to query the image from the image

IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 1, September 2012
ISSN (Online): 1694-0814
www.IJCSI.org

database. First, they integrate global color and texture feature vectors to narrow down the search space and in the next state they process using local features. As color and texture features, they used color moments and sub-band statistics of wavelet decomposition. They reported that objects with non uniform color and non homogenous regions can be found effectively.

A method based on the primitives of color moments is proposed in [29]. In the method, an image is divided into four segments and the color moments extracted from the segments are clustered into four classes. They consider the mean moments of each class as a primitive of the image. All primitives are used as features and each class mean is combined into a single class mean. The distance between query image mean with the corresponding database images are calculated by using Sum-of-Absolute-Differences (SAD). They reported that the proposed method based on color moments shows better performance than the local histogram method.

A multi feature model for the Content Based Image Retrieval System is proposed in [30] by combining the Color Histogram, color Moment, texture, and edge Histogram descriptor features. Users were given options to select the appropriate feature extraction method for best results. They report 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.

Maheshwari et al. [31] have proposed a method in which Color moment and Gabor filter are used to extract features for image dataset. K-means and hierarchical clustering algorithm is applied to group the image dataset into various clusters.

Two methods of content based image retrieval using color and texture features have been implemented in [32]. In both the methods, feature extraction method is done using color moment while feature extraction of texture is done using wavelet texture features and Gabor texture features. Top images are retrieved using Euclidean distance and Chi-square distance and they have made comparative analysis.

We believe that a minimal amount of spatial information encoded in the color index will improve the discriminating power of plain color indexing techniques. In this paper, in order to solve the disadvantages of global color indexing techniques, we encode spatial information in the index by dividing each image in the database into three equal non-overlapping horizontal regions as shown in Fig. 2(e). Experimentally it is found that out of the various image divisions, the retrieval based on color moment by dividing the image as shown in Fig. 2(e) gives the best retrieval performance. Each region is represented by a vector which consists of a total of 9 values i.e., average hue, variance of hue and skewness of hue, average saturation, variance of saturation and skewness of saturation, average of value, variance of value and skewness of value from each of the region. Thus each image in the database is stored as a vector of 27 floating point values (or 3 regions, each region being represented by a vector of 9 floating point values). The experimental results show that the moments extracted from the image divided horizontally into 3 non-overlapping equal regions gives the best performance and hence in our proposed method color moments are extracted from these 3 horizontal regions.

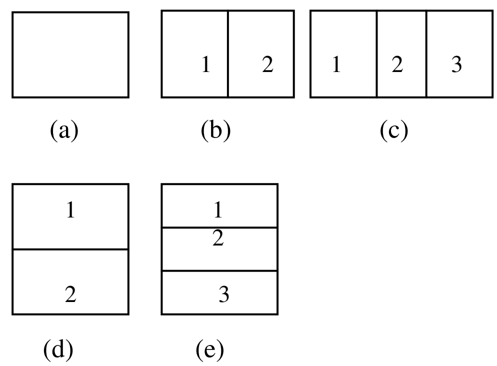


Fig. 2 Different image regions: (a) Whole image (b) Image divided vertically into two equal regions (c) Image divided vertically into three equal regions (d) Image divided horizontally into two equal regions (e) Image divided horizontally into three equal regions

The remainder of the paper is organized as follows: in section 2, color feature extraction and similarity measurements are presented. In Section 3, texture feature extraction and texture similarity measurement are presented. Section 4 outlines the proposed method. Section 5 describes the experiments and results. Finally, conclusions are presented in section 6.

2. Color Representation

Color is one of the most important features that make possible the recognition of images by humans and color feature is one of the most commonly used visual features in image retrieval. Color is a property that depends on the reflection of light to the eye and the processing of that information in the brain. It is an important dimension of human visual perception that allows discrimination and recognition of visual information [33]. Color features are relatively easy to extract and match, and have been found to be effective for indexing and searching of color images in image databases.

One of the main aspects of color feature extraction is the choice of a color space. A color space is a multidimensional space in which the different dimensions represent the different components of color. An example of a color space is RGB, which assigns to each pixel a three element vector giving the color intensities of the three primary colors, red, green and blue. The space spanned by the R, G, and B values completely describes visible colors, which are represented as vectors in the 3D RGB color space. As a result, the RGB color space provides a useful starting point for representing color features of images. However, the RGB color space is not perceptually uniform. More specifically, equal distances in different intensity ranges and along different dimensions of the 3D RGB color space do not correspond to equal perception of color dissimilarity.

The RGB color space can be transformed to generate other color spaces. The idea for color space transformation is to develop a model of color space that is perceptually similar with human color vision. Color spaces such as HSV, CIE 1976 (LAB), and CIE 1976 (LUV) are generated by nonlinear transformation of the RGB space. The CIE color spaces represent the three characteristics that best characterize color perceptually: hue, lightness, and saturation. However, the CIE color spaces are inconvenient because of the calculation complexities of the transformation to and from the RGB color space. HSV color space is also a nonlinear transformation of the RGB, but it is easily invertible [33]. The HSV color space is approximately perceptually uniform. In this paper, we use HSV color space to extract color features.

The HSV color space is widely used in the field of color vision. The chromatic components hue, saturation and value correspond closely with the categories of human color perception. The HSV values of a pixel can be transformed from its RGB representation according to the following formula:

$$H = \cos^{-1} \frac{\frac{1}{2}[(R\text{-}G)+(R\text{-}B)]}{\sqrt{(R\text{-}G)^2+(R\text{-}B)(G\text{-}B)}}$$

$$S=1 - \frac{3[\min(R,G,B)]}{R+G+B}$$

$$V=\left(\frac{R+G+B}{3}\right)$$

2.1 Color Feature Extraction

The goal of color indexing is to retrieve all the images whose color compositions are similar to the color composition of the query image. Histograms are useful because they are relatively insensitive to position and orientation changes and they are sufficiently accurate [11]. However, they do not capture spatial relationship of color regions and thus, they have limited discriminating power. Many publications focus on color indexing techniques based on global color distributions. These global distributions have limited discriminating ability because they are unable to capture local color information. Color correlogram and color coherence vector can combine the spatial correlation of color regions as well as the global distribution of local spatial correlation of colors. These techniques perform better than traditional color histograms when used for content-based image retrieval. But they require very expensive computation. Color moments have been successfully used in content based image retrieval systems. It has been shown [12] that characterizing one dimensional color distributions with the first three moments is more robust and runs faster than the histogram based methods.

In this paper, in order to improve the discriminating power of color indexing techniques, we divide the image horizontally into three equal non-overlapping regions and from each of the three regions, we extract from each color channel the first three moments of the color distribution and store the 27 floating point numbers in the index of the image. If we interpret the color distribution of an image as a probability distribution, then the color distribution can be characterized by its moments [12]. If the value of the ith color channel at the jth image pixel is I_{ij} and the number of pixels is N, then the index entries related to this color channel and the region 'r' are:

$$E_{r,i} = \frac{1}{N}\sum_{j=1}^{N} I_{ij} \quad (1)$$

$$\sigma_{r,i} = \left(\frac{1}{N}\sum_{j=1}^{N}(I_{ij} - E_{r,i})^2 \right)^{\frac{1}{2}} \quad (2)$$

$$s_{r,i} = \left(\frac{1}{N}\sum_{j=1}^{N}(I_{ij} - E_{r,i})^3 \right)^{\frac{1}{3}} \quad (3)$$

The entries $E_{r,i}$ (1 <= i <=3) are the average color of the region r. The entries $\sigma_{r,i}$ and $s_{r,i}$ are the variance and the skewness of each color channel in this region 'r'.

The index entry for one image consists of
Index size = number of regions X number of color Channels X 3 floating point numbers.

In our case, the image is divided horizontally into three equal regions and we have to store 27 floating point numbers per image.

So, the feature vector f_c of length 27 is given by:

$$f_c = \{E_{1,1}, \sigma_{1,1}, s_{1,1}, E_{1,2}, \sigma_{1,2}, s_{1,2}, E_{1,3}, \sigma_{1,3}, s_{1,3} \ldots\ldots\ldots\ldots E_{r,i}, \sigma_{r,i}, s_{r,i}\}$$

(1 <= r, i <=3), r represents the region and i represents the color channel.

2.2 Color Similarity Measure

To determine the similarity of two images at query time, we measure the similarity between their indices. Let H and I be two color images with c color channels. If the index entries of these images for their regions are $E_{r1,i}$ resp. $F_{r2,i}$, $\sigma_{r1,i}$ resp. $\rho_{r2,i}$ and $s_{r1,i}$ resp. $t_{r2,i}$, then the similarity of the region r1 of H and the region r2 of I is defined as

$$d_{r1,r2}(H,I) = \sum_{i=1}^{c}\left(\frac{|E_{r1,i} - F_{r2,i}|}{|E_{r1,i}| + |F_{r2,i}|} + \frac{|\sigma_{r1,i} - \rho_{r2,i}|}{|\sigma_{r1,i}| + |\rho_{r2,i}|} + \frac{|s_{r1,i} - t_{r2,i}|}{|s_{r1,i}| + |t_{r2,i}|} \right) \quad (4)$$

Where $(1 \le i \le 3)$ and c is the number of channels.

To compute the total similarity of the two images H and I, we define the total similarity as

$$d(H,I) = \sum_{r1=r2=1}^{r=3} d_{r1,r2}(H,I) \quad (5)$$

where r is the total number of regions.
If

$$f_c^Q = \{E_{1,1}, \sigma_{1,1}, s_{1,1}, E_{1,2}, \sigma_{1,2}, s_{1,2}, E_{1,3}, \sigma_{1,3}, s_{1,3} \ldots\ldots\ldots\ldots E_{r,i}, \sigma_{r,i}, s_{r,i}\}$$

denote color feature vector of query image and

$$f_c^T = \{E_{1,1}, \sigma_{1,1}, s_{1,1}, E_{1,2}, \sigma_{1,2}, s_{1,2}, E_{1,3}, \sigma_{1,3}, s_{1,3} \ldots\ldots\ldots\ldots E_{r,i}, \sigma_{r,i}, s_{r,i}\}$$

denote color feature vector of database image, then distance between them is given by:

$$d_1 = \sum_{i=1}^{d} \frac{|f_c^Q - f_c^T|}{|f_c^Q| + |f_c^T|} \quad (6)$$

where d is the dimension of the vector.

The Canberra distance measure is used for similarity comparison. It allows us the feature set to be in un-normalized form. The Canberra distance measure is given by:

$$CanbDist(x, y) = \sum_{i=1}^{d} \frac{|x_i - y_i|}{|x_i| + |y_i|} \quad (7)$$

Where x and y are the feature vectors of query and database image respectively, of dimension d.

The following algorithm is proposed to determine the similarity between query image and an image in the image database:

Step 1. Input query image I
Step 2. Convert RGB color space image into HSV color space.
Step 3.Partition the image into three equal non-overlapping horizontal regions
Step 4. Calculate the moments $F_{r2,i}$, $\rho_{r2,i}$, $t_{r2,i}$ (section 2.1) for each color channel of each region to get 27 numbers from three regions of the query image I.
Step 5. Apply Step 2 to Step 4 to the image H_j in the database, to calculate the moments $E_{r1,i}$, $\sigma_{r1,i}$, and $s_{r1,i}$ from the regions of H_j to get 27 numbers.

Step 6. Calculate the Distance $d_j(H_j,I)$ between the two images using Eq.(5) and store in an array d.

$$d_j(H_j, I) = \sum_{r1=r2=1}^{r} d_{r1,r2}(H_j, I)$$

where r is the number of regions.
Step 7. Increment j, repeat step 5 and 6 for all the images in the database.
Step 8. The array d is sorted in ascending order. The image corresponding to the first element of d is the most similar image compared with the query image I. The first 10 top most similar images are then displayed.

3. Texture Representation

Texture [34] is defined as structure of surfaces formed by repeating a particular element or several elements in different relative spatial positions. Generally, the repetition involves local variations of scale, orientation, or other geometric and optical features of the elements. Image textures are defined as images of natural textured surfaces and artificially created visual patterns. It contains important information about the structural arrangement of the surface i.e., clouds, leaves, bricks, fabric, etc. It also describes the relationship of the surface to the surrounding environment. It is a feature that describes the distinctive physical composition of a surface.

Gabor wavelet is widely adopted to extract texture from the images for retrieval and has been shown to be very efficient. Basically Gabor filters are a group of wavelets, with each wavelet capturing energy at a specific frequency and specific orientation. The scale and orientation tunable property of Gabor filter makes it especially useful for texture analysis. The design of Gabor filter is done as follows: [35 and 36]

For a given image I(x,y) with size PXQ, its discrete Gabor wavelet transform is given by a convolution:

$$G_{mn}(x,y)=\sum_s\sum_t I(x-s,y-t)\,\psi^*_{mn}(s,t) \tag{8}$$

Where, s and t are the filter mask size variables, and ψ^*_{mn} is a complex conjugate of ψ_{mn} which is a class of self-similar functions generated from dilation and rotation of the following mother wavelet:

$$\psi_{m}(x,y)=\frac{1}{2\pi\sigma_x\sigma_y}\exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right].\exp(j2\pi Wx) \tag{9}$$

where W is called the modulation frequency. The self-similar Gabor wavelets are obtained through the generating function:

$$\psi_{mn}(x,y)=a^{-m}\psi(\bar{x},\bar{y}) \tag{10}$$

Where m and n specify the scale and orientation of the wavelet respectively, with m=0,1,……….M-1,n= 0,1,……….N-1, and

$$\bar{x}=a^{-m}(x\cos\theta+y\sin\theta) \tag{11}$$

$$\bar{y}=a^{-m}(-x\sin\theta+y\cos\theta)$$

Where a>1 and $\theta=n\pi/N$.

The variables in the above equation are defined as follows:

$$a=(U_h/U_l)^{\frac{1}{M-1}},\quad W_{m,n}=a^mU_l$$

$$\sigma_{x,m,n}=\frac{(a+1)\sqrt{2In2}}{2\pi a^m(a-1)U_l}, \tag{12}$$

$$\sigma_{y,m,n}=\frac{1}{2\pi\tan(\frac{\pi}{2N})\sqrt{\frac{U_h^2}{2In2}-(\frac{1}{2\pi\sigma_{x,m,n}})^2}}, \tag{13}$$

In our implementation, we used the following constants as commonly used in the literature:

U_l =0.05, U_h =0.4, s and t range from 0 to 60, i.e., filter mask size is 60x60.

3.1 Texture Feature Extraction

After applying Gabor filters on the image with different orientation at different scale, we obtain an array of magnitudes:

$$E(m,n)=\sum_x\sum_y\left|G_{mn}(x,y)\right|, \tag{14}$$

m=0,1,……, M-1; n=0,1,………N-1.

These magnitudes represent the energy content at different scale and orientation of the image. The main purpose of texture-based retrieval is to find images or regions with similar texture. It is assumed that we are interested in images or regions that have homogenous texture, therefore the following mean μ_{mn} and standard deviation σ_{mn} of the magnitude of the transformed coefficients are used to represent the homogenous texture feature of the region:

$$\mu_{mn}=\frac{E(m,n)}{PXQ} \tag{15}$$

$$\sigma_{mn}=\frac{\sqrt{\sum_x\sum_y(\left|G_{mn}(x,y)\right|-\mu_{mn})^2}}{PXQ} \tag{16}$$

A feature vector f_g (texture representation) is created using μ_{mn} and σ_{mn} as the feature components [3]. Four scales and 6 orientations are used in common implementation and the feature vector of length 48 is given by:

$$f_g=\{\mu_{00},\sigma_{00},\mu_{01},\sigma_{01},..............,\mu_{35},\sigma_{35}\}$$

3.2 Texture Similarity Measure

The texture similarity measurement of a query image Q and a target image T in the database is defined by:

$$d(Q,T)=\sum_{m}\sum_{n} d_{mn}(Q,T) \qquad (17)$$

where $d_{mn}=\frac{\left|\mu_{mn}^{Q}-\mu_{mn}^{T}\right|}{\left|\mu_{mn}^{Q}\right|+\left|\mu_{mn}^{T}\right|}+\frac{\left|\sigma_{mn}^{Q}-\sigma_{mn}^{T}\right|}{\left|\sigma_{mn}^{Q}\right|+\left|\sigma_{mn}^{T}\right|}$

If $f_g^Q=\{\mu_{00},\sigma_{00},\mu_{01},\sigma_{01},..............,\mu_{35},\sigma_{35}\}$ denote texture feature vector of query image and

$f_g^T=\{\mu_{00},\sigma_{00},\mu_{01},\sigma_{01},..............,\mu_{35},\sigma_{35}\}$ denote texture feature vector of database image, then distance between them is given by:

$$d_2=\sum_{i=1}^{d}\frac{\left|f_g^Q-f_g^T\right|}{\left|f_g^Q\right|+\left|f_g^T\right|} \qquad (18)$$

The Canberra distance measure is used for similarity expression.

4. Proposed Method

The novel method of color moment (based on division of the image into 3 equal non overlapping horizontal regions) + Gabor texture features gives better results compared to the color moment (based on whole image) + Gabor texture features and others using only single feature.

4.1 Color Feature

In the proposed method, we are retrieving the images using low level features, color and texture. While retrieving the images using color feature, the RGB image is converted into HSV space. The global distributions have limited discriminating ability because they are unable to capture local color information. To improve the discriminating power of color indexing, we encode a minimal amount of spatial information in the index by dividing the image horizontally into three equal non-overlapping regions and extracting moments from these regions. In our experiment, we calculate the color feature vector f_c of length 27 for query image and images in the image database and distance between them is computed using Canberra distance measure. The results of a query are displayed in decreasing similarity order.

4.2 Texture Feature

In the case of low level texture feature, we apply Gabor filters on the image with 4 scales and 6 orientations and we obtain an array of magnitudes. The mean μ_{mn} and standard deviation σ_{mn} of the magnitudes are used to create a texture feature vector f_g of length 48. Canberra distance measure is used for computing the distance and the results of a query are displayed in decreasing similarity order.

4.3 Image Database

For evaluation of the proposed method, it has been implemented using Matlab 6.5 and tested on a general-purpose WANG database [37] containing 1000 Corel images in JPEG format of size 384 x 256 or 256 x 384. The image set comprises 100 images in each of 10 categories. In our experiment, we have selected 100 images randomly, containing 10 images in each category and the images are resized to 256 x 384. Within this database, it is known whether any two images are of the same category. In particular, a retrieved image is considered a match if and only if it is in the same category as the query.

4.4 Combining the Feature

The retrieval result using only single feature may be inefficient. It may either retrieve images not similar to query image or may fail to retrieve images similar to query image. Hence, to produce efficient results, we use combination of color and texture features.

The similarity between query and target image is measured from two types of characteristic features which includes color and texture features. Two types of characteristics of images represent different aspects of property. So, during similarity measure, appropriate weights are considered to combine the features. The distance between the query image and the image in the database is calculated as follows:

d =w1*d1+w2*d2 (19)

Here, w1 is the weight of the color features, w2 is the weight of the texture features and d1 and d2 are the distances calculated using color moment using Eq.(6) and texture features using Eq.(18). Experiments show that better retrieval performances are achieved when we set w1=0.80 and w2=0.20. The weight factor of color feature distance is higher than the weight factor of texture feature distance because our database consists of mostly natural images.

The above distance'd' is calculated between the query image and all the images in the database and it is sorted in ascending order. The image corresponding to the first element of d is the most similar image compared with the query image. The first 10 top most similar images are then displayed.

5. Experiments and Results

The performance of a retrieval system can be measured in terms of its precision and recall. Precision measures the ability of the system to retrieve only models that are relevant, while Recall measures the ability of the system to retrieve all models that are relevant. They are defined as

$$\text{Precision} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}} = \frac{A}{A+B} \tag{20}$$

$$\text{Recall} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of relevant images}} = \frac{A}{A+C} \tag{21}$$

Where A represents the number relevant images that are retrieved, B represents the number of irrelevant items and C the number of relevant items those were not retrieved.

If Precision (P), and recall (R) for query image I_k (k=1,…..1000) are defined as:

$$P(I_k, N) = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved (N)}} \tag{22}$$

$$R(I_k) = P(I_k, |A(I_k)|) \tag{23}$$

Where, $|A(I_k)|$ represents the numbers of relevant images in the respective category, then the average precision for images belonging to the qth category (A_q) has been computed by [38]:

$$\bar{P}_q = \sum_{k \in A_k} P(I_k) / |A_q|, \ q=1,2,...10 \tag{24}$$

Finally, the average precision is given by:

$$\bar{P} = \sum_{q=1}^{10} \bar{P}_q / 10 \tag{25}$$

The average recall is also computed in the same manner. Some of the images of the database used for retrieval experiment are shown in Fig. 3.

The experiment was carried out with the number of retrieved images set as 10 to compute the average precision P of each query image. In order to assess the discriminating power of the techniques proposed in this paper, we carried out the experiments based on Gabor texture feature (GTF), color moment - based on whole image (CMW), color moment - image divided into 3 equal non overlapping horizontal regions (CMR), color moment - based on whole image + Gabor texture feature (CMW + GTF), and color moment - image divided into 3 equal non overlapping horizontal regions + Gabor texture features (CMR + GTF).

Some of the results using the same query image of Fig. 4 (a) are shown in Fig. 4. From Table 1. , it is seen that the average precision (%) based on (CMR+GTF) is 61.0 and the average precision (%) based on (CMW+GTF) is 58.2. Thus the proposed method demonstrates clearly that our encoding of spatial information in the color index from different regions of the image significantly increases the discriminating power compared to the color moment (based on whole image) + Gabor texture features indexing techniques in which color moments are extracted from the entire image.

It is also seen that the value of the average precisions (%) based on single features i.e. only Gabor texture features or only Color moments are less than the average precisions (%) of combined features of color moments and Gabor texture features as shown in Table 2. and Table 3. This also shows that there is considerable increase in retrieval efficiency when both color and texture features are combined for CBIR.

Table 1: Precision of retrieval for top 10 images. Gabor texture feature (GTF); Color moment- whole image (CMW); Color moment – image divided into 3 equal non overlapping horizontal regions(CMR); Color moment- whole image + Gabor texture feature (CMW + GTF); Color moment- image divided into 3 equal non overlapping horizontal regions + Gabor texture feature (CMR + GTF)

Class	Average Precision using				
	GTF	*CMW*	*CMR*	*CMW + GTF*	*CMR + GTF*
African	0.37	0.75	0.75	0.74	0.74
Beach	0.27	0.46	0.38	0.38	0.38
Building	0.33	0.25	0.35	0.30	0.36
Buses	0.35	0.67	0.78	0.60	0.77
Dinosaurs	0.99	0.74	0.83	0.96	0.95
Elephants	0.39	0.60	0.45	0.58	0.44
Flowers	0.75	0.42	0.61	0.71	0.69
Horses	0.27	0.55	0.70	0.47	0.67
Food	0.44	0.67	0.62	0.72	0.69
Mountain	0.20	0.43	0.43	0.36	0.41
Average Precision (%)	43.6	55.4	59.0	58.2	61.0

Table 2: Average retrieval of: Gabor texture features (GTF); Color moment - whole image (CMW); Color moment- whole image + Gabor texture feature (CMW+GTF)

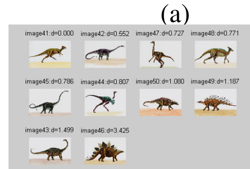
	GTF	*CMW*	*GTF+CMW*
Average Precision (%)	43.6	55.4	58.2

Table 3: Average retrieval of: Gabor texture feature (GTF); Color moment- image divided into 3 equal non overlapping horizontal regions (CMR); Color moment- image divided into 3 equal non overlapping horizontal regions + Gabor texture feature (CMR+GTF)

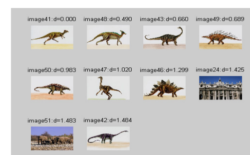
	GTF	*CMR*	*CMR+GTF*
Average Precision (%)	43.6	59.0	61.0

Fig. 3 Sample of WANG Image Database

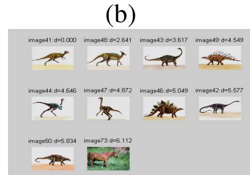
(a)



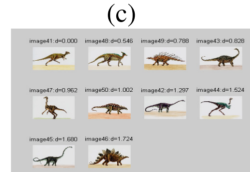
(b)



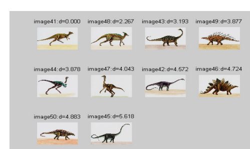
(c)



(d)



(e)



(f)

Fig. 4 Different image retrieval results for the same query image: (a) query image (b) result based on Gabor texture feature (c) result based on Color moment (whole image) (d) result based on Color moment (image divided into 3 equal non overlapping horizontal regions) (e) result based on Color moment (whole image) + Gabor texture feature (f) result based on Color moment (image divided into 3 equal non overlapping horizontal regions) + Gabor texture feature

6. Conclusion

In this paper, we have proposed an efficient image retrieval method based on color moments and Gabor texture features. To improve the discriminating power of color indexing techniques, we encode a minimal amount of spatial information in the index by extracting features from the regions of the image divided horizontally into three equal non overlapping regions. In this approach, from each region in the image, we extract from each color channel the first three moments of the color distribution and store the 27 floating point numbers (or 3 regions, each region being represented by a vector of 9 floating point numbers) of the image in the index. As its texture feature, Gabor texture descriptors are adopted. We calculate the similarity with combined features of color and texture using Canberra distance as similarity measure. Our Experimental results demonstrate that the proposed method has higher retrieval accuracy than other conventional methods combining color moments and texture features based on global features approach. The experiment also shows that only color features or only texture features are not sufficient to describe an image. There is considerable increase in retrieval efficiency when both color and texture features are combined. Thus it is rightly said that only color or only texture cannot differentiate a cheetah and a tiger.

References

[1] R.Datta, D. Joshi, J.Li, J.Z. Wang, " Image retrieval: ideas, influences, and trends of the new age ", ACM Computing Surveys 40(2), 2008, pp. 1-60.

[2] V.N. Gudivada, and V.V. Raghavan, "Content based image retrieval systems", IEEE Computer, Vol. 28, No. 9, 1995, pp. 18-22.

[3] B.S. Manjunath, and W.Y. Ma, " Texture Features for browsing and retrieval of image data", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 8, 1996, pp. 837-842.

[4] Y. Rui, T.S. Huang, M. Ortega, S. Mehrotra, " Relevance feedback : a power tool for interactive content based image retrieval ",IEEE Circuits and Systems for Video Technology , Vol. 8, No. 5, 1998, pp. 644-655.

[5] D. Swets, and J. Weng, "Hierarchical discriminant analysis for image retrieval", IEEE "PAMI, Vol. 21, No. 5, 1999, pp. 386-400.

[6] H. Zhang and D. Zhong, "A scheme for visual feature based image retrieval", Proc. SPIE storage and retrieval for image and video databases, 1995.

[7] A.M. Smeulders, M. Worring and S. Santini, A. Gupta and R. Jain, " Content-based image retrieval at the end of the early years", IEEE Trans Pattern Anal Machine Intell 22:, 2000, pp. 1349-1380.

[8] R. Choras, "Content-based image retrieval using color, texture, and shape information", In. Sanfeliu, Riuz-Shulcloper J. (eds) Progress in pattern recognition, speech and image analysis. Springer, Heidelberg. 2003.

[9] R. Corners, and C. Harlow, "A theoretical comparison of texture algorithms", IEEE Trans Pattern Anal Machine Intell 2: 1980, pp. 204-222.

[10] P. Howarth, and S. Ruger, "Evaluation of texture features for content based image retrieval", In: Enser P. et. al. (eds) Image and video retrieval. Springer LNCS 3115: pp. 326-334.

[11] M.Z. Swain, and D.H. Ballard, "Color Indexing", Intl. J. of Computer Vision 7(1): 1991, pp. 11-32.

[12] M. Stricker, and M. Orengo, "Similarity of color images", In SPIE Conference on Storage and Retrieval for Image and Video Databases , volume 2420, 1995, pp. 381-392, San Jose, USA.

[13] V.E. Ogle, and M. Stonebraker, "Chabot: Retrieval from a relational database of images", Computer, 1995, pp. 40-48.

[14]J.P. Eakins, and M.E. Graham, "Content-based Image Retrieval: A report to the JISC Technology Applications Program" http://www.unn.ac.uk/iidr/research/cbir/report.html.

[15] H. Tamura, S. Mori, T. Yamawaki, "Texture features corresponding to visual perception", IEEE Trans. On Systems, Man and Cybernetics. 6(4): 1976, pp. 460-473.

[16]W. Niblack et. al., "The QBIC Project: Querying Images by Content Using Color, Texture and Shape". Proc. Of the Conference Storage and Retrieval for Image and Video Databases, SPIE vol. 1908, 1993, pp. 173-187.

[17] F. Liu, and R.W. Picard, "Periodicity, directionality and randomness: Wold features for image modelling and retrieval", IEEE Transactions on Pattern Analysis and Machine Intelligence 18(7): 1996, pp.722-733.

[18] L.M. Kaplan et al., "Fast texture database retrieval using extended fractal features" in Storage and Retrieval for Image and Video Databases VI(I.K. Sethi and R.C. Jain eds), Proc. SPIE 3312, 1998, pp.162-173.

[19] J.R. Smith, "Integrated Spatial and Feature Image System: Retrieval, Analysis and Compression", Ph.D. thesis, Columbia University, 1997.

[20]Y. Deng, "A Region Representation for Image and Video Retrieval", Ph.D. thesis, University of California, Santa Barbara, 1999.

[21] W.Y. Ma, "Netra: A Toolbox for Navigation Large Image Databases", Ph.D. thesis, University of California, Santa Barbara, 1997.

[22]S. Jeanin (ed.), "ISO/IEC JTCI/SC29/WG11/N3321: MPEG-7 Visual Part eXperimentation Model Version 5.0", Nordwijkerhout, 2000.

[23]A. Dimai, "Rotation Invariant Texture Description using General Moment Invariants and Gabor Filters", In Proc. of the 11th Scandinavian Conf. on Image Analysis. Vol I, 1999, pp. 391-398.

[24] J.L. Shih, and L.H. Chen, "Color image retrieval based on primitives of color moments", IEEE Proceedings online no. 20020614., 2002.

[25]R.S. Choras, T. Andrysiak, M. Choras, "Integrated color, texture and shape information for content-based image retrieval", Pattern Anal Applic. 10: 333-343, 2007.

[26]B. Xue and L. Wanjun, "Research of Image Retrieval Based on Color", IEEE International Forum on Computer Science-Technology and Applications, 2009.

[27]Z.C. Huang, P.P.K. Chan, W.W.Y. Ng, D.S. Yeung, " Content-based image retrieval using color moment and Gabor texture feature", in Poceedings of the IEEE Ninth International Conference on Machine Learning and Cybernetics, Qingdao, 2010, pp. 719-724.

[28]D.K. Kumar, E.V. Sree, K. Suneera, P.V.Ch. Kumar, "Content Based Image Retrieval – Extraction by objects of user interest", International Journal of Computer Science and Engineering (IJCSE), Vol.3,No.3., 2011, pp. 1068-1074

[29]T.V. Saikrishna, A. Yesubabu, A. Anandrao, T.S. Rani, "A Novel Image Retrieval Method using Segmentation and Color Moments", ACIJ, Advanced Computing : An International Journal, Vol. 3, No.1, 2012, pp.75-80.
[30]R.S.Dubey, R. Choubey, J. Bhattacharga, " Multi Feature Content Based Image Retrieval", (IJCSE) International Journal on Computer Science and Engineering, vol. 02, No. 06, 2010 , pp. 2145-2149.
[31]M. Maheshwari, S. Silakari, M. Motwani, "Image Clustering using Color and Texture", Computational Intelligence, Communication Systems and Networks, 2009, pp. 403-408.
[32]P.P. Buch, M.V. Vaghasia, S.M. Machchchar, "Comparative analysis of content based image retrieval using both color and texture", Engineering(NUiCONE), Nirma University Internatonal Conference, 2011, pp. 1-4.
[33] J. Smith,"Color for Image Retrieval", Image Databases: Search and Retrieval of Digital Imagery, John Wiley & Sons, New York, 2002, pp. 285-311.
[34] J. Zhang, G. Li, S. He, "Texture-Based Image Retrieval by Edge Detection Matching GLCM", The 10th IEEE International Conference on High Performance Computing and Communications.
[35] D.A. Clause, M. Ed. Jerni, Gan, "Designing Gabor filters for optional texture separability", Pattern Recognition, 33, 2000, pp. 1835-1849.
[36] D. Zhang, A. Wong, M. Indrawan, G. Lu, " Content- based Image Retrieval Using Gabor Texture Features", available online, Australia, 2003.
[37 http://wang.ist.psu.edu/
[38]H.A. Moghadam, T. Taghizadeh, A.H. Rouhi, M.T. Saadatmand, "Wavelet correlogram: a new approach for image indexing and retrieval", J. Elsevier Pattern Rec. 38(2005), pp. 2506 – 2518.

S. Mangijao Singh received the M.Sc. degree in Physics from Manipur University, Manipur in 1991 and the MCA degree from Punjab Technical University, Punjab in 2011. He is currently working as Deputy Secretary in the Board of Secondary Education, Manipur, India. He is pursuing Ph.D. in the Department of Computer Science, Assam University, Silchar. His research area is Image Processing.

Prof. K. Hemachandran obtained his M.Sc. degree from Sri Venkateswara University, Tirupati and M.Tech. and Ph.D. degrees from Indian School of Mines, Dhanbad. He is associated with the Department of Computer Science, Assam University, Silchar, since 1998. His areas of research interest are Image Processing, Software Engineering and Distributed Computing.