Content Based Image Retrieval Based On Color And Texture Features Using KNN Classifier

A DISSERTATION

Submitted in partial fulfillment of the requirements for the award of the degree

of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

Sanjeev Suman

(14535042)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE -247 667 (INDIA)

MAY, 2016

CANDIDATE'S DECLARATION

I hereby declare that the work, which is being presented in the dissertation entitled **‘‘Content Based Image Retrieval based on Color and Texture features using KNN Classifier’’** towards the partial fulfillment of the requirement for the award of the degree of **Master of Technology** in **Computer Science and Engineering** submitted in the Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, Roorkee, Uttarakhand (India) is an authentic record of my own work carried out during the period from July 2015 to May 2016, under the guidance of **Dr. R. Balasubramanian, Associate Professor,** Department of Computer Science and Engineering, IIT Roorkee.

The matter presented in this dissertation has not been submitted by me for the award of any other degree of this or any other institute.

Date:

Place: Roorkee **(Sanjeev Suman)**

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Date:

Place: Roorkee **(Dr. R. Balasubramanian)**

Associate Professor

Department of Computer Science and Engineering

IIT Roorkee

ACKNOWLEDGEMENT

I feel honored in presenting this dissertation report in such an authenticable form of sheer endurance and continual efforts of inspiring excellence from various coordinating factor of cooperation and sincere efforts drawn from all sources of knowledge.

I express deep gratitude for enthusiastic and valuable suggestions that I got from my guide Dr. R. Balasubramanian (Associate Professor), Department of Computer Science & Engineering, Indian Institute of Technology Roorkee, for successful completion of this project. His integrity and commitment has inspire me.

I heartily thank professors and technical staff of Computer Science & Engineering, IIT Roorkee, for helping me in exploring the possibility of carrying out the experiments in their laboratory.

Finally, I would like to acknowledge with gratitude, the support and love of my family and friends. They all kept me going.

ABSTRACT

Content based image retrieval (CBIR) is a technique to retrieve desired images from large databases against a query image on the basis of some similarity between features extracted from images itself. Here content based defines that search is on the basis of contents of the image rather than text based search using keywords, or some tags or metadata linked with the image. CBIR uses the information present in an image for query formulation and to give result. As the internet grows exponentially, it leads to huge number of digital images embedded in databases and traditional text based searching becomes tedious. A fast and efficient search technique required to deal with images in large volume which leads to this new field. In this thesis a new method is proposed which use color and texture features of image for construction of feature vector. K-Means clustering is used to cluster similar type of images and finally KNN is used to retrieve the desired results.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

In the recent years size of multimedia files increased exponentially due to expansion of World Wide Web and increased memory storage because of reduction of cost in memory. With this good part, it has also created a new problem to find the desired image in large databases. Users from different part of world are accessing remote database in exciting ways. So the requirement of a fast and efficient mechanism to retrieve images from massive databases comes in view.

From historical perspective, it is noticed that the premature image retrieval systems were text-based search. The images were indexed and annotated manually based on requirement. But with the increase of the multimedia files this manual task becomes very tedious, and generally it fails to describe the rich information present in the images. Existing software work on the basis of string match and many of the displayed result are not of the user's interest. People started to think to develop software which can use the content present in an image. This leads to the new research area which is called as CBIR.

CBIR is based on the retrieval of color, shape, texture and other informational features present in the image. Different features are chosen according to requirement and a feature vector is formed from them. Then this feature vector is matched with the images in database. Same features are retrieved from the image of database and result is displayed based on some image similarity. Search can be done by association, aim or category. Probably the first use of CBIR was done by T. Kato in 1992 to fetch similar images present in the dataset automatically on the basis of color similarity and shape similarity of an image. After that different type of CBIR system was developed but due to absence of current technology they did not make a big success. First commercial system was made by IBM, named as QBIC (query by image content) which used color, shape and texture of image. Another system Virage of Virage Inc used color, color layout, shape and texture for image retrieval. By looking the high range of uses further research were done to use the rich features present in image for fast and expert image retrieval system. The problem can be divided into broad domain and narrow domain. In narrow domain user is more precise about information and search is limited but in case of broad domain there is no precise knowledge, so there is large possibility of result which may be cumbersome. Therefore, we need a system to reduce the broad domain into narrow domain to retrieve efficiently.

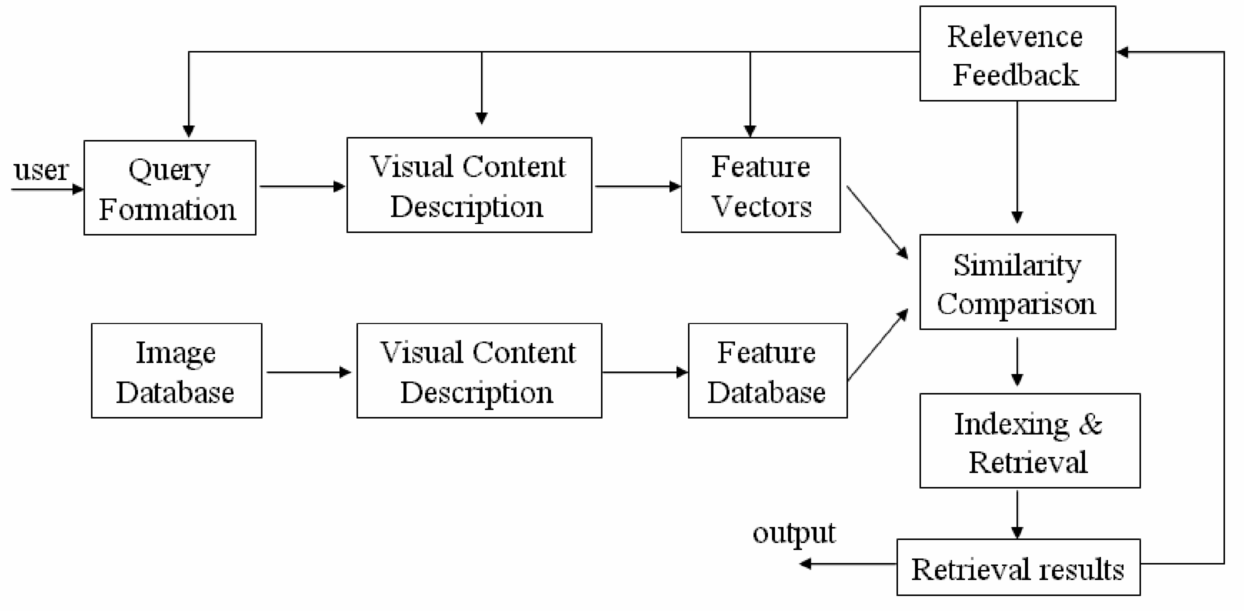

Figure 1.1 Basic diagram of Content-based image retrieval [Long 2003]

Query starts with an image. First query is formed by extracting chosen underlying features from the image provided. After some enhancement a feature vector is developed for image. Then according to the feature vector search is done in image database. On the basis of some image-image similarity comparison, the query image is matched against all the image of database. Top matching results against query is displayed. The process is made more efficient and automated by the use of user feedback.

2. PRIMARY FEATURES

The primary features used in image retrieval field are: color, shape and texture.

2.1 *Color*

Color is the most important visual features used in CBIR systems. Color shows light variation because of change in view angle or orientation of surface and this should be handled very carefully. RGB (red, green, blue) model is most widely used color space model where various colors are represented by combining the different intensity of red, green, and blue color. A new model which uses the opponent color axes has the advantage of putting the two chromatic axes and separate the information related to brightness on the third axis. This approach is useful because our eye is more sensitive to brightness [1].

A histogram represents the intensity of a particular color present in an image. In case of digital images it talks in terms of pixels.

A color histogram H is given by a vector as

$$H = \{H[0], H[1], \ldots, H[n]\} \tag{2.1}$$

Where Ii represents a particular color and

H[i] is number of pixel of color i in image and n is the number of bins.

The HSV representation is mainly used for object retrieval because of its invariant properties. In this model hue is invariant even in change of orientation of surface and camera direction. The invariant color space is for matte patches is given by

$$[(R-G) / (R+G), \ -(B-R) / (B+R), (G-B) / (G+B)] \tag{2.2}$$

The invariant for a shiny surface and white illumination is given by

$$|R-G| / (|R-G| + |B-R| + |G-B|) \tag{2.3}$$

One more approach to limit the inequality in observation due to surface reflection is to find clusters in a color histogram. By looking at elongated streaks we can find the clusters of pixel reflected from surface. Hence it is helpful to identify the pixels originated from a uniformly colored object.

2.2 *Shape*

Shape is another important primitive feature used in image content retrieval. Shape information can be easily obtained by use of contours. Various features related to shape is calculated for every object present in the image stored in database. Skeleton or contour based algorithm is used to extract the skeleton or contour of an image. When a query is done then the same features is extracted from the query image and display images having much similarity. Shape has mainly two types of feature, global features and local features. Moment invariants, aspect ratio comes under global part and the features like boundary segment comes under local part. Shape detection plays an important role in 3-D image.

2.3 *Texture*

Texture is another primitive essential attribute of image which has made a considerable contribution in field of content based image retrieval. Normally texture refers to randomness, coarseness of an image. But in case of CBIR texture is more than this. Various wavelet based transform are used to capture the frequency in an image which contains the texture information. In the field of cbir it is the most widely used features after color. Tamura selected the six important texture features based on the human visual perception. Out of these six, three (Coarseness, Contrast and Directionality) are very effective and widely used for texture retrieval in cbir engine. Remaining three (line-likeness, Regularity, Roughness) are correlated to the above three features and does not add much benefit in texture retrieval. Coarseness tells the information about size of texture elements. Roughness is directly proportional to coarseness. The size of the chosen operator depends on the size of texture present in image. Contrast tells about the quality of picture that is change of grey levels in an image and their biasness towards black and white. Directionality is computed by the use of frequency dissemination of aligned regional edges against their directional angles. Sobel edge detector is used to calculate the edge strength and angle of direction. Regularity is used to measure how regular image is. The line-likeness is given by the midpoint coincidence of the edge orientations that co-occurred in the form of pairs of pixels which are located at a certain distance along the edge orientation in every pixel. The roughness feature is defined as the sum of coarseness measures and contrast measures. Various approaches come over period to extract these features from images. The use of a particular feature is highly depends on the fact that which type of image is going to be indexed and matched for retrieval. Based on the requirement one or more of these features can be used for image retrieval in cbir.

3. Literature Review

Huang[2] presents color correlogram as the new feature for indexing and retrieval of images in cbir. Histogram represents only the count of pixels in an image, not ant spatial information. To use the rich spatial information present in an image, correlogram is used. Correlation of the image color is described as the function of spatial distance. Color correlogram of an image is represented as a table in which k-th entry corresponding to i-th row and j-th column tells the probability of finding a pixel of color j at a distance of k from a pixel of color i. Color correlogram is has all the features which is required for a good feature vector. It shows the global spatial relation between colors in an image, easy to compute and size of feature vector is also low which leads to fast calculation. Color autocorrelogram is a small set that is subset of color correlogram which represent the probability of finding a pixel of same color at specific distance. This feature is both effective and inexpensive in the field of cbir.

Ojala[3] proposed a new approach for texture retrieval of an image. Here, rotation invariance and gray-scale invariance is used to identify the regular pattern present in the local texture of image. The term regular pattern refers to local binary pattern present in a spatial resolution which includes edges, corner and spots. There are two main advantage of this approach, first one that it doesn't change due to monotonic transformation and the other one is its computational simplicity as it needs very few operations to check circular neighborhood using lookup table. This method uses two main fundamental local properties, spatial structure and contrast. Spatial pattern is changed because of change in rotation and remains constant when gray-scale is changed. Opposite to this, contrast varies according to gray-scale but remains unchanged in case of rotation. The propose approach starts with a texture operator which allows to identify LBP at circular neighborhoods of any quantization of angular space at any spatial resolution. To achieve gray-scale invariance, the gray value of center pixel is subtracted from the gray value of all P points in neighborhood. The rotation invariance is increased by finer quantization of angular space.

Jing, Gui-li and seok[4] proposed a new approach of retrieving texture information which gives higher precision compare to existing system. In this approach first edge information is obtained using edge detector and then matched with corresponding grey level co-occurrence matrix. Edges of an image tell about the shape, size and texture. For edge detection, there was a requirement of edge detector. Here, Prewitt operator is selected because it is not susceptible to noise. Four direction edge images (East, southeast, south, southwest) are formed to combine with the four properties of co-occurrence matrix. Usually a threshold value is set to obtain clear edge but in this case no any threshold is set so that they don’t lose any texture information. Edge images formed are divided into 4, 9 or 16 parts as required. Then for each part co-occurrence matrix is constructed and four properties: contrast, entropy, energy and inverse difference moment is calculated. Co-occurrence matrix tells the spatial relationship between two different grey level pixels separated by a certain distance under specific angle. Finally feature vector is formed on the basis of obtained statistical feature.

Murala, Gonde and Maheshwari[5] suggest the use of color histogram combined with Gabor wavelet transform. From the starting days of CBIR, color is the primly used attribute. Color histogram shows the composition of colors (in pixels for digital image) in an image over a given color space (RGB OR HSV). It tells that how many times a particular color appears in the given image? Rotation or translation has no effect on histogram. It shows slight change only because of modification of angle of view. Gabor wavelet is a set of wavelets, where each wavelet captures energy in a specific direction at a specific frequency. Wavelet transform based upon Gabor wavelet contains redundancy, which is reduced by projecting it to filters. After applying filters we get array of magnitudes. Then mean and standard deviation of the magnitude obtained is calculated to represent similar texture feature. Using mean and standard deviation, a feature vector is formed which is matched to retrieve desired result.

Murala, Maheshwari and Balasubramanian[6] suggest the use of Local Tetra Patterns (LTrPs) for texture retrieval which is based on four direction encoding. In this proposed method Gabor transformation with three scales and two directions (0 and 90 degree) are applied to compute first order derivatives to (n-1)th-order derivatives in two directions i.e. vertical and horizontal directions. Initially the query image is loaded and translated into gray-scale. Then first order derivatives along 0 degree (horizontal) and 90 degree (vertical) are applied. Direction is calculated for each pixel. The pattern obtained is divided into four pieces on the basis of orientation of center pixel. Tetra pattern is computed and put into 3 different binary patterns. After this, histogram of each binary pattern is calculated. Magnitude of center pixel is calculated. Again binary patterns are constructed and histogram is calculated. Then the previously calculated histogram and new histogram is combined to construct a feature vector. Now, database is searched for the required feature vector and satisfying result is displayed.

Cardoso and Muller[7] suggest the use of multiple SVM ensembles for cbir. This paper is based on the iterative use of SVM. The given dataset which is already divided in N class are trained by the use of N SVM ensemble. Now when a test image is provided to the search engine then this SVM ensemble is use to find the appropriate class for the test image. In this paper one-against-all approach is used to enhance the target search. In the first step feature is extracted from image by using DCT (discrete cosine transform). When a query image is received then a new ensemble is constructed based on the one-against-all approach. In first iteration multiple classes can be returned as result, but iteration stops only one class is returned as result. Now images of the single returned class is used to compute similarity with test image and producing result.

Another approach was proposed by Dr. Anna and S. Vinod[8] in which they suggest the use of Discrete Wavelet Transform (DCT) and Histogram of Oriented Gradients (HOG). They used the RGB color space for image retrieval. An image is transformed using discrete wavelet transform. In this process an image is degraded into four parts: approximated image, vertical (measure variation along vertical edges), diagonal (measure variation along diagonal edges) and horizontal (measure variation along horizontal edges). One or more than one decomposition can be used to achieve more accuracy. For face detection two or more decomposition are useful. Final decomposed image is used to construct the feature vector. After this, histogram of oriented gradients is calculated from image. For HOG calculation first different interest point is detected by using Harris detector (it defines the local structure of an image based on autocorrelation matrix). Now HOG is calculated around the 16 * 16 pixel region of each interest point. Finally size of 128 dimension feature vector is obtained which is used to calculate the distance between test image and the images of database.

Kumar and Sarvanan[9] suggest a new approach of using grid code of image for indexing and retrieval of images which is completely based on the color histogram. Color histogram of an image is calculated by counting the number of pixels of different color present in an image over a chosen color space. The feature vector using histogram can be retrieved by quantizing them. In this approach formulation of grid code is done by quantizing the feature vector that is obtained from histogram of an image. All the images which have similar grid code are kept in a single grid. When an image is queried then corresponding grid code is matched against the grid code of images present in the image of dataset and then according to match between grid codes top results are fetched.

Kimaya and Ajay Agarkar[10] focused on the use of SIFT (Scale Invariant Feature Transform) in cbir. SIFT feature includes the process of detection of scale-space extrema (by the use of difference of Gaussian function), key point localization (based on the stability), orientation assignment (direction of each image gradient), and key point descriptor. SIFT feature with 128 dimension is used for construction of feature vector. Different kind of approach like KDtree with BBF (best bin first) or ANN (approximate nearest neighbor) can be used for indexing or matching purpose. NNDRS is a modified voting method which can be used to display the result based on the greater similarity between images.

Richa, Sitesh and Mukesh[11] suggest another approach for cbir using CM (Columnar mean), DM (diagonal mean), RGBC, HA (histogram analysis) and ED (Euclidean distance). In this method, columnar mean is defined as empirical (average) mean for each column of image. These are matched against columnar mean of query image. Diagonal mean of an image is defined as the average mean of those pixels which are on the standard diagonal of that image. It is beneficial than obtaining mean for each rows as only one value is enough to match from query image. Histogram just tells the number of pixels of different color present in a color space. RGB component is used which represent a color as the combination of red, green and blue color. Finally Euclidean distance is used to calculate the difference between two pixels.

In [12] HSV color space is used for histogram retrieval and three level decomposition of Haar wavelet transform is used for texture information. Result is obtained by calculating Euclidean difference between query image and database image.

In [13] Color histogram, Gray level co-occurrence matrix, Color dominant region is used to construct the feature vector. Similarity matching is done by using K-Means clustering algorithm and Euclidean distance.

In [14] HSV color space is used for histogram extraction, Gray level co-occurrence matrix is used for texture retrieval and gradient method is used for shape extraction.

In [15] Co-occurrence matrix for color feature and DBPSP (difference between pixels of scan pattern) is used for texture retrieval. Artificial neural network is used as an classifier and result is given by finding minimum area between two vectors.

4. Research Gap

Selection of Features: Color, shape and texture are the features which are generally used for image search. There are various methods to extract these three features. Among these different methods, the selection of suitable methods to construct the feature vector of an image is a big problem. The speed and accuracy of the image retrieval system totally depends on the selected features.

Sensory gap: The informational space present between the real world thing and the recorded scene of that thing is called sensory gap. It becomes more difficult if there is no advance knowledge of recording state. When both 2-D and 3-D images are stored then it create problem in making decision. 2-d image may be similar to many 3-D objects and it leads to ambiguity. Sensory gap includes viewpoint, clutter, occlusion etc.

Semantic gap: The existing gap between the result obtained by visualization of data and user's perspective in a given condition is called semantic gap. Two different images can be similar on the basis of retrieved attribute but user may found it different. Similarly different images can found dissimilar on the basis of retrieved attribute but user may find those similar.

5. Proposed Approach

Like every cbir engine, the proposed method is consist of four steps – defining the descriptor (which feature is going to be used), indexing the dataset (extract the features of images from dataset and stored them), defining the similarity metric (metric used to calculate distance between two images based on their feature vector) and searching (same features are calculated from query image and result is displayed according to measure of similarity). First we will discuss all the terms that is used for proposed method. After that we will discuss about algorithm.

5.1 Histogram

In general, histogram is a graphical representation which tells about the distribution of pixels in an image. Mathematically, it is a two dimensional graph where one dimension represent the tone scale between white and black (black at one end and white at another) another dimension shows the count of pixels corresponding to a certain tonal scale.

Color histogram is the widest used feature of image in the field of content based image retrieval because of its simplicity. It is easy to calculate as it just tells the distribution of different colors present in an image spanned over a color space. Another pros of this feature is that it is extremely flexible means it can be made for any color space, whether it is RGB (a three dimensional color space where x-axis represents number of pixels of red color, y-axis represents number of pixels of green color and z-axis represents the number of pixels of blue color) or it is HSV (a three dimensional color space where x-axis represents the hue, y-axis represents the saturation and z-axis represents the value) also referred as HSB (b stand for brightness) or any other color space.

For the proposed work HSV histogram is taken at the place of RGB histogram. One of the reasons for doing this is the limitation of RGB color space in the field of computer vision. RGB is good for visual perspective (display screens, projectors) but in computer vision it is difficult to predict that what proportion of red, blue, and green color will produce a specific color. To solve this problem, HSV is used which gives more suitable representation. Another reason of using it is easy conversion of RGB to HSV and HSV to RGB. In this color space whiteness refers to value and colorfulness refers to saturation. Hue is shown by putting one color at its full intensity, second as variable and third as zero.

In the proposed work the given input image is quantized using HSV color space into 8 * 2 * 2 bins. First image is divided into HSV plane and then quantized by using quantization value as 8 for hue, 2 for saturation, and 2 for value. Quantization reduces the number of colors required to display an image and is very useful where less number of colors are available to display an image. After that histogram matrix of size 8 * 2 *2 is obtained which finally is flattened to obtain a 1 * 32 feature vector as output.

5.2 Correlogram

Histogram feature of an image is easy to calculate but it only tells the composition of colors in that image, not any spatial information and because of this, two different object may have same histogram. For example, a cup of red color and a plate of red color may have same histogram and the cbir engine which only works on histogram will tell it similar object. Also histogram is sensitive to lighting intensity and other errors like quantization error.

Color correlogram of an image is a feature which tells about the spatial distribution of color in the image. Correlation of the image color is described as the function of spatial distance. Color correlogram of an image is represented as a table in which k-th entry corresponding to i-th row and j-th column tells the probability of finding a pixel of color j at a distance of k from a pixel of color i. Color correlogram is has all the features which is required for a good feature vector. It shows the global spatial relation between colors in an image, easy to compute and size of feature vector is also low which leads to fast calculation. It outperform old histogram based method and histogram refinement method for retrieval of images from a database.

Color autocorrelogram is a small set that is subset of color correlogram which represent the probability of finding a pixel of same color at specific distance. Mathematically, probability of finding a pixel of color i, at a distance of k from a pixel of color i. Also color correlogram is hugely independent of difference caused in look and shape by variation of view angle or zooming.

In the proposed work color autocorrelogram is used because it tells the spatial information of same color and gives relatively smaller feature vector than correlogram. The time required to retrieve images from database is depend upon the size of feature vector. Therefore smaller size of autocorrelogram makes it inexpensive and helps in faster retrieval. It shows how correlation between different pair of color changes with distance.

The given input image is quantized using 64 colors in RGB space. The obtained feature vector from the code is of 1 * 64. For each image in database colorautocorrelogram is calculated and concatenated with the previously obtained feature vector. At the time of query again it is calculated from input image and gives result based on the corresponding match.

5.3 Gabor Wavelet & Filters

After color, texture is an important feature present in an image which extraction is necessary for efficient content based image retrieval. Similarity between textures of two different images is measured by various methods. Most of these methods are based on second-order statistics calculation from both image (query image and images stored in database). Normally texture refers to the property like contrast, coarseness, randomness, periodicity, directionality etc. traditional methods of image retrieval used the different combination of these properties. Another way of texture retrieval is by using Gabor filter. Gabor wavelet is useful in case of edge detection, corner detection and BLOB detection.

Mathematically Gabor function g(x, y) is given by

$$g(x,y) = \left(\frac{1}{2\pi\sigma_x\sigma_y}\right)\exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)+2\pi jWx\right] \qquad (5.1)$$

and its Fourier transform is written as

$$G(u,v) = \exp\left\{-\frac{1}{2}\left[\frac{(u-W)^2}{\sigma_u^2}+\frac{v^2}{\sigma_v^2}\right]\right\} \qquad (5.2)$$

where $\sigma_u = 1/2\pi\sigma_x$ and $\sigma_v = 1/2\pi\sigma_y$.

g(x, y) is treated as mother wavelet and self-similar is obtained by rotation and dilation through chosen generator function. Gabor wavelet is closely related to Gabor filter. By the experiment result it is shown that Gabor filter can easily reflect the response of the natural vision system to a larger extent.

The discrete wavelet transform for a specific image I(x, y) is given by:

$$G_{mn}(x, y) = \sum_{s}\sum_{t} I((\text{x-s, y-t})\,\psi^{*}_{\text{mn}}(s, t) \tag{5.3}$$

Where s and t represents filter mask size variables and function of s and t is the complex conjugate of self-similar function produce by the appropriate dilation and rotation of the above mother wavelet g(x, y).

Gabor filter are defined as set of wavelets, where each wavelet captures energy at a particular frequency in a specific direction. These capture the localized features of an image and by this group of wavelets we can extract the texture features. The frequency and orientation tuning property of Gabor filter makes easier texture retrieval from an image.

As result, Gabor filter results into array of magnitudes which show the energy distribution at different orientation and frequency (scale) of the image. Wavelet transform based upon Gabor wavelet contains redundancy, which is reduced by projecting it to filters. After applying filters we get array of magnitudes. Then mean and standard deviation of the magnitude obtained is calculated to represent similar texture feature.

The mean μ is given by:

$$\mu = \sum_{x}\sum_{y} G(x, y) \tag{5.4}$$

and the standard deviation is given by:

$$\sigma = \sqrt{\sum_{x}\sum_{y}(|G(x, y)| - \mu)}^{2} \tag{5.5}$$

In our project, mean squared energy and mean amplitude at different scale and rotation is retrieved from query image and the images from database. Using this mean squared energy and mean amplitude, a feature vector is formed which is matched to retrieve desired result. The experiment shows that the use of Gabor features based wavelet transform has significant improvement over other wavelet based transform like pyramid-structured or tree-structured wavelet transform. Combining it with histogram and correlogram outperforms the individual use of color histogram or correlogram or

Gabor wavelet transform, standard wavelet transform or combination of other wavelet transform and color histogram.

5.4 K-Means Clustering

Input: k, the number of cluster,

Output: D, a data set consists of n objects clustered into k number of cluster

K-Means and the variation of K-Means are the most well-known and widely used partitioning methods in the field of image processing and data mining. It is cluster based partitioning method which takes an input value k (number of cluster) and a dataset D of size n (number of objects) and divides the set of n objects into k number of cluster. It partition the dataset in such a way that the similarity between object of same set is high and similarity between set of different object is low. Suppose two points i and j are in cluster c1 and other two point x and y are in cluster c2, then feature of i will be more similar to j as compare to i and x or i and y. Similarly j will be more similar to i as compared to j and x or j and y. Same phenomena will exist for the object of other cluster. It is a centroid based clustering algorithm where the term similarity is measured with respect to mean value of the object in a particular cluster.

K-Means algorithm works in the following manner. Initially when there is no cluster it selects k points randomly and these k points become the cluster mean or the center of cluster. Now when k cluster is known to it, the remaining objects (n-k) are assigned to some cluster. The remaining objects are assigned to a cluster on the basis of similarity, means to the cluster which is most similar to this object among all clusters. This similarity depends upon the distance between the object and the mean value of the cluster. The object is assigned to a cluster having minimum distance between mean value of that cluster and object. After this it calculate the new mean of the cluster. Now again each object are assigned to the cluster based on similarity which depends upon distance metric according to new mean. Again new mean is calculated and updated for each cluster and again the same process repeats. This loop continues until change occurs in two continuous iteration. When there is no change means criterion function coincides then the process stops.

The algorithm is based on the aim to partition the dataset D into k (number of cluster) partitions having minimal square-error. The calculation of means, calculation of similarity and the selection of starting k means may change according to variation of K-Means. A variation of K-Means algorithm is K-Modes where instead of cluster mean, cluster mode is calculated. Modes of clusters are updated by using frequency based methods. Also these two variations can be combined to cluster that type of dataset which have numerical as well as categorical value.

K-Means algorithm:

Step 1: choose k random points form dataset D as the starting centers;

Step 2: repeat

Step 3: assign each object to the cluster which is most similar to this object, based on mean value;

Step 4: calculate the new mean for each cluster and update the mean value of each cluster;

Step 5: until no change;

This algorithm clusters D into k number of cluster such that square error function remains minimal. The square-error function is given by:

$$E = \sum_{i=1}^{K} \sum_{p \in Ci} |p - mi|^2 \tag{5.6}$$

Where E is the sum of squared error,

P is the point in space and m represents the mean

The minimal value of this squared error criterion makes the cluster compact and separate that means intra cluster object have greater similarity and inter cluster objects have less similarity.

The time complexity of this algorithm is O(nkt), where n represents the number of objects, k represents the number of cluster and t represents the number of iteration. This is the reason which makes K-Means relatively scalable and works fine for the large size of dataset. This algorithm often terminates at local optimum. One thing in this algorithm is necessity of k in advance that means user have to specify the number of cluster in advance. Also it can be applied only when mean of a cluster is defined that means work well with numerical data but not with categorical data.

5.5 K-Nearest-Neighbor (KNN) classifier

There are two types of learners, one is eager learners and second is lazy learners. When a set of training data is provided to early learners then at first they prepare a classification model on training dataset before receiving any test data. Once the classification model is organized it is eager to classify the new data. Lazy learners are just opposite to them. In case of lazy learners, they do not construct a classification model when training dataset is provided. Instead of this they wait until a test data arrives. When training dataset is provided to lazy learners, they store data without any processing or with some processing and wait for test data. When test data is arrived then to classify it they measure the similarity between test data and training data. Lazy learners are also called instance based learners.

Simply it means that eager learners do more amount of work during training and less amount of work during testing. In contrast to eager learners, lazy learners do a little work at training time and more amount of work at testing time. In compare to eager learners lazy learners are computationally costly but they support incremental learning. Support vector machines, decision tree based classification, backpropagation based classification, rule based classification comes under eager learners and k-nearest-neighbor is the example of lazy learners.

K-nearest neighbor (KNN) algorithm is widely used lazy learners. It classified the test data based on the similarity with the training data. Suppose the data are n-dimensional. All the training tuples have n-dimensional attribute space. When a new test tuples comes having n-dimension, then the distance between all the n-dimension of test tuple and all stored training tuple is calculated. Based on the high closeness means better similarity top k training tuples are selected which are known as k nearest neighbor for the given test tuple.

Here closeness or similarity is calculated by the use of distance metric, like Euclidean distance or city block (Manhattan) distance or any other distance. In case of Euclidean distance the component-wise difference is calculated and then squared and accumulated. After this, square root of integrated distance is calculated. Each component or attribute are normalized by using any normalization technique (like min-max normalization) to intercept outweighing. K nearest neighbor is obtained for a given test tuple and this is assigned to the top matching class means having highest similarity among k nearest neighbors. Nearest neighbor classifier is also useful for prediction in which it predict for the given test tuple. In that case it provides the mean value of the valued labels annotated with obtained k nearest neighbor of the test tuple.

Now a problem arise that the above described method is fine for those attribute which are numeric, but what to do when attributes are not numerical but categorical, like color? A simple method to compare categorical attribute is to match all the corresponding value of a test tuple to the stored tuple. Let say there are two tuples x and y. Now for the categorical data like color, if both x and y have the same color then the difference between them is considered as zero. If both x and y have different color then the difference between them can be taken as one.

If the value of a particular attribute is not present in one of the tuples or in both tuples then difference is taken as maximum possible difference. In case of categorical attribute, if any one of the attribute is not present in any tuple or that attribute is not present in both tuple then difference is considered as one. Similarly for numeric data if attribute is missing from both tuple then the difference is considered as one.

One more problem is to find the appropriate value for k. The value of k is obtained experimentally. Starting with the initial value of k as one, a test set is used to estimate the fault rate of the classifier. In each repetition the value of k is incremented by one that is finding one more neighbor. At each value of k the error rate is calculated. By comparing all error rate obtained for different value of k, the k having minimum error rate can be selected. Generally the value of k is directly proportional to the number of training tuples.

K-nearest neighbor classifier use distance metric to calculate the closeness between the new tuple and the existing tuple which are stored. These distance based metric generally give equal amount of weight to each attribute. Therefore they are sensitive to noisy or outlier attribute and this may lead to poor accuracy. However, this classifier can be modified to support weighted attribute and elimination of noise.

This classifier is very slow during testing. If dataset is D and $k = 1$ then this method requires $O(|D|)$ comparisons to classify the given test tuple. By some kind of sorting and storing them into balanced search tree can decrease the number of comparisons to $O(\log|D|)$. Parallel implementation of this classifier can reduce the number of comparison to constant value. A technique called partial distance calculation can used to improve the speed of classification. In this method, instead of calculating distance over n components, distance is calculated only over a subset of n attribute. An appropriate threshold value is chosen. During distance calculation, if a stored tuple exceeds this threshold value then further calculation of similarity with this tuple is stopped and the method moves on to the next training tuple. Useless tuples can be removed by editing or pruning method.

5.6 Distance Metric

In the field of image processing, the similarity between images is measured by some kind of distance metric. Images having smaller difference of distance are considered as more similar as compare to images having more difference. There are different methods to measure the distance between images. Some of them are following:

Euclidean distance: It measures the shortest distance between two points means displacement. It is also known as crow distance because crow flies like this. It is calculated by finding the square of component wise difference between points, add them and finally square root of accumulated result is obtained. Euclidean distance d(p, q) between two points, p = [p_1, p_2, …..p_n] and q = [q_1, q_2, …..q_n] is given by:

$$d(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \tag{5.7}$$

Manhattan (City block) distance: In this metric moving diagonally is not permitted. This is defined as sum of horizontal and vertical difference between two points. Manhattan distance d(p, q) between two points, p = [p_1, p_2, …..p_n] and q = [q_1, q_2, …..q_n] is given by:

$$d(p, q) = \sum_{i=1}^{n} |p_i - q_i| \tag{5.8}$$

Chessboard distance: This metric is used when speed is more important than accuracy. It is defined as the maximum of horizontal difference and vertical difference between two points. Chessboard distance d(p, q) between two points, p = [p_1, p_2, …..p_n] and q = [q_1, q_2, …..q_n] is given by:

$$d(p, q) = \max(|pi - qi|) \tag{5.9}$$

Minkowski difference: This difference metric is the generalization of Manhattan and Euclidean distance. Minkowski distance d(p, q) between two points, p = [$p_1, p_2, \ldots p_n$] and q = [$q_1, q_2, \ldots q_n$] is given by:

$$d(p,q) = (\sum_{i=1}^{n} |p_i - q_i|^c)^{1/c} \tag{5.10}$$

Generally the value of c is taken as 1 or 2. For c = 1 it becomes Manhattan distance and for c = 2 the above equation becomes Euclidean distance.

Cosine similarity: This metric is used to find the orientation difference between two vectors. Generally it is used in positive space where values are in the range of [0, 1]. The value of cosine similarity is given by 1 for same orientation, 0 for vectors which are perpendicular and -1 for vectors which are opposite to each other. The cosine of two vectors A and B is given by:

$$A.B = |A|\,|B| \cos\theta \tag{5.11}$$

Therefore cosine similarity can be given by:

$$\cos\theta = \frac{A.B}{|A||B|} \tag{5.12}$$

Some other kinds of distance metric are correlation distance, spearman distance, standard Euclidean distance, Jaccard distance, Mahalanobis distance etc. which can be used to measure the similarity between two images.

5.7 Algorithm of proposed method

1. For 1:size(dataset)
2. Compute HSV histogram
3. Compute color auto correlogram
4. Compute color moment
5. Obtain Gabor features
6. Compute wavelet moment
7. Make a feature vector using result obtained from step 2 to 6
8. Apply K-means to make cluster
9. A representative feature vector is assigned to each cluster.
10. Store them into database.
11. End of for
12. Take query image as input
13. Step 2 to 7 is repeated for query image
14. Feature vector of query image is matched against feature vector of clusters
15. Distance with all the cluster is calculated
16. Query image is assigned to the cluster having minimum difference
17. KNN is applied on the resultant cluster
18. Top K (number of returned image) images are displayed as output
19. Go to step 12

In the first step that is defining the descriptor, HSV histogram (represent the number of pixels present in the image over HSV color space), color autocorrelogram (to obtain the spatial information between two pixels of same color present in the image), Gabor wavelet (for calculation of mean squared energy and mean amplitude), color moment and wavelet moment is chosen.

In the second step that is indexing the dataset, all the above mention features are extracted from each image present in the dataset. In HSV histogram calculation first the input image in RGB is converted to HSV and splitted into corresponding planes. These values are quantized using quantization level as 8, 2 and 2 respectively. Finally a vector of 1 * 32 is obtained from this part. After this color autocorrelogram is calculated by quantizing the image using 64 colors. A vector if 1* 64 is obtained which tells the spatial information between pixels. Then color moment is computed which gives 1 * 6 vector which gives first and second order color moments (mean and standard deviation). Now image is converted into gray scale and Gabor features are calculated. Fast Fourier Transform is applied to image for convolution. Two vectors, one stores mean squared energy and another stores mean amplitude at different scale and orientation. Wavelet moment is calculated which gives 1 * 20 vector containing the mean and standard deviation of wavelet. After getting all these features, all of them are concatenated together to form a feature vector. Now K-Means is applied on dataset to make cluster. Suitable number of cluster is square root of size of dataset. Now each cluster has their representative vector and these are stored.

In the third step that is defining similarity metric, Euclidean distance is chosen to compute the similarity between query image and stored image. It is calculated by taking the square root of sum of square of component wise difference between two feature vectors. Another distance metric like Manhattan, Chessboard distance can also be used.

In the fourth and final step that is searching, a query image is required which will be searched again the stored image. When a query image is provided, then all the chosen features which were extracted from the image of dataset, are extracted from the query image. All features are concatenated together and feature vector is constructed. Now this feature vector is matched against the feature vector of all clusters that are stored in the database. The distance between feature vector of query image and the representative feature vector of cluster are calculated by the computation of Euclidean distance. When distance is calculated for all clusters then the query image is assigned to the cluster having minimum difference. It means that the query image belongs to the cluster having minimum difference. After this K-nearest-neighbor algorithm is applied to that cluster to find out the K most similar images against query image, where k is the number of image returned as the result of query. Based on the result of KNN, K obtained images are selected and sorted and finally displayed to the user as output.

6. Experimental Results

6.1 Parameter Setting

We used MATLAB 2015(b) to implement all the experiments in the window environment on a 64 bit 1.70 GHz Intel(R) Core(TM) i3-4005U PC with 16 GB RAM.

6.2 Dataset

For the retrieval of results WANG database is used which is a subset of Corel Database. It consists of 1000 images which have been manually selected to be a database of 10 classes of 100 images each. The images are subdivided into 10 classes, such that it is almost sure that a user wants to find the other images from a class if the query is from one of these 10 classes. This is the major advantage of this database because due to the given classification it is possible to evaluate retrieval results.

Semantic name	A sample of the images in each cluster
Africa	
Beach	
Building	
Buses	
Dinosaurs	
Elephants	
Flowers	
Horses	
Mountains	
Food	

Figure 6.1 Example images from each of the 10 classes of WANG database

http://wang.ist.psu.edu/docs/related

6.3 Discussion on the result

Table 6.1 shows the average precision for each class for number of images retrieved K chosen as 5, 10, 20, 50, and 100. From each class 10 random images are selected as query image and k numbers of images are retrieved against that query image. Average precision is calculated for 10 image of each class and displayed.

Class\No of Images	**5**	**10**	**20**	**50**	**100**
Africa	0.88	0.81	0.66	0.632	0.544
Beach	1	0.92	0.855	0.824	0.715
Monuments	0.92	0.73	0.62	0.645	0.582
Buses	1	0.89	0.825	0.778	0.7
Food	1	0.84	0.715	0.712	0.657
Dinosaurs	1	1	1	0.964	0.912
Elephants	0.94	0.86	0.735	0.684	0.604
Flowers	1	0.97	0.925	0.836	0.767
Horses	0.96	0.81	0.84	0.668	0.634
Mountains	0.92	0.78	0.615	0.622	0.563
Average	0.962	0.861	0.779	0.737	0.668

Table 6.1: Average precision value for each class at different value of K

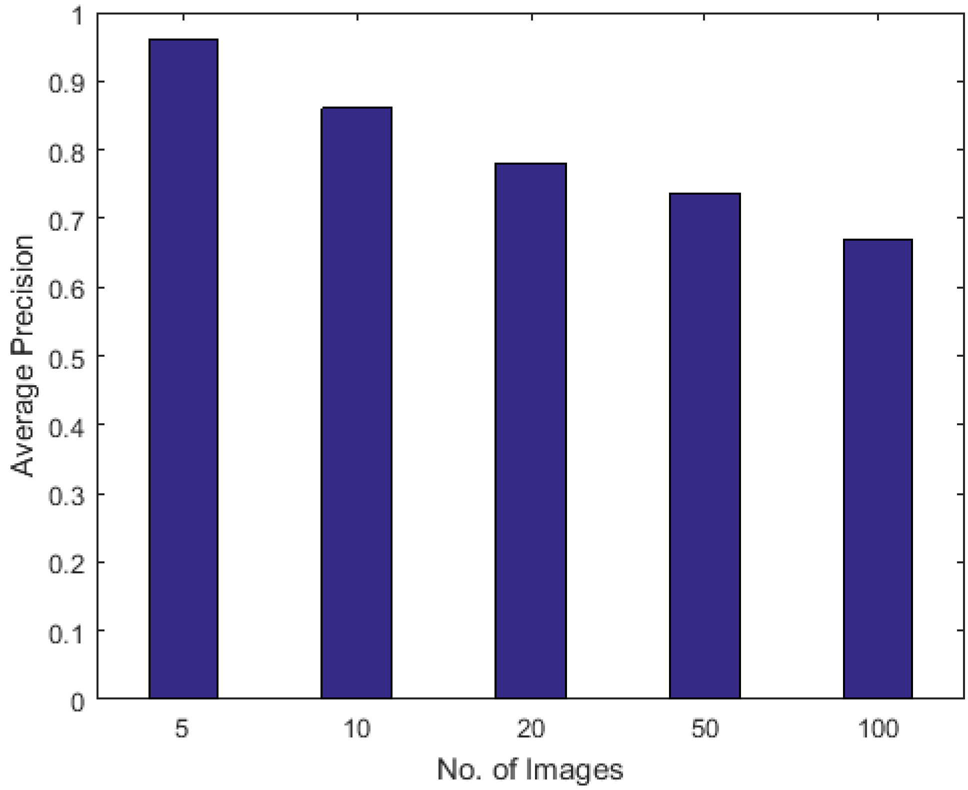

Figure 6.2 Graphical representation of average precision of proposed method

In the graph we can see that precision of proposed method is high for small number of retrievals and gradually decreases for more number of retrievals. For initial values precision decrease by more amount but after then it decreases by small amount. This method is able to give good results for large number of retrieval also.

Table 6.2 shows the average precision for each class obtained by different model. From each class 10 random images are selected as query image and K numbers of images are retrieved against that query image. Here value of K is chosen as 20. 10 random images are selected from each class and average precision is calculated for 20 as retrieval set.

Class/Models	Elalami [15]	Chuen [17]	Jhanwar [18]	Huang [19]	Proposed
Africa	0.72	0.68	0.45	0.42	0.66
Beach	0.59	0.54	0.4	0.44	0.86
Monuments	0.58	0.56	0.37	0.41	0.62
Buses	0.89	0.89	0.74	0.85	0.83
Food	0.77	0.73	0.37	0.42	0.72
Dinosaurs	0.99	0.99	0.91	0.58	1
Elephants	0.7	0.66	0.3	0.42	0.74
Flowers	0.93	0.89	0.85	0.89	0.93
Horses	0.85	0.8	0.56	0.58	0.84
Mountains	0.56	0.52	0.29	0.26	0.62
Average	0.76	0.72	0.52	0.53	0.78

Table 6.2: Average precision value for different models for K = 20

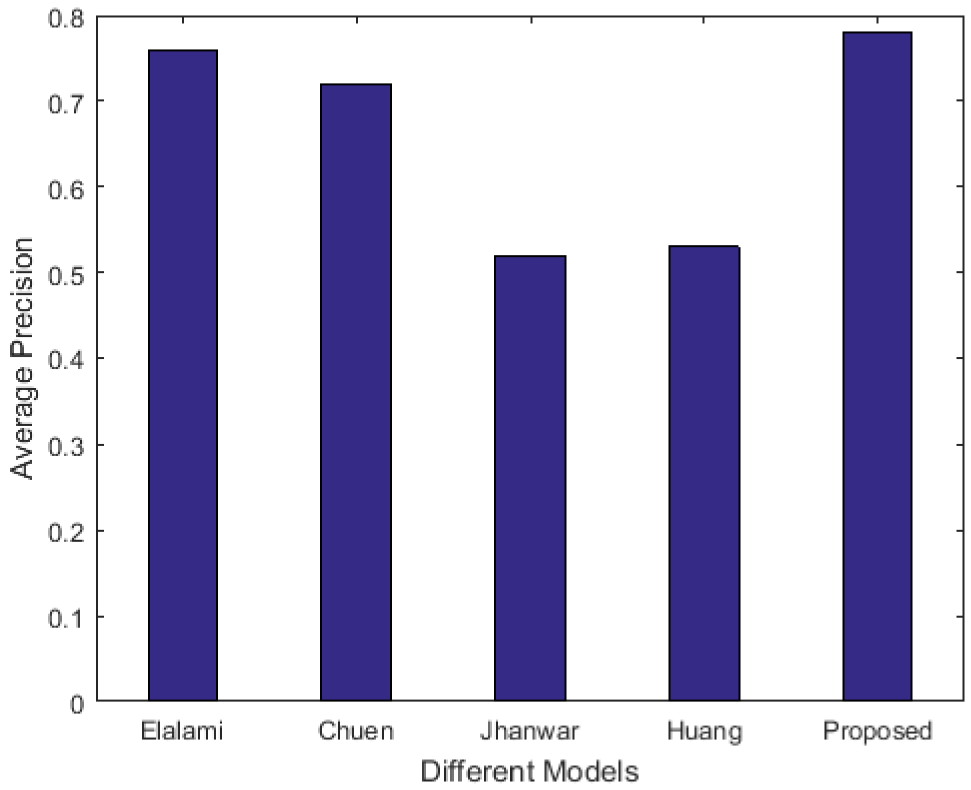

Figure 6.3 Average precision of different models

By the above results we can see that the proposed method has better precision than others which makes it efficient. The small size of feature vector makes it fast for image retrieval.

7. APPLICATIONS

CBIR has a large range of applications in various fields. Some of them are mentioned below.

The military

CBIR is used in many military applications like spotting the destination using satellite images, specification of instructions system for using missiles and identification of rival aircraft from radar screens.

Crime prevention

To minimize the crime and enforce law, Government agencies maintain huge collection of visual evidence which include facial photographs, fingerprints, shoeprints. If a crime happens, then evidence can be matched to the existing records.

Intellectual property

It is used for Copyright protection, Trademark registration etc. where new one is matched against the records that already exist so that two cannot be same.

Architectural and engineering design

Most of the features are common in this area. Having similar records is valuable as it will help to understand problem better and also reduce the time span as well as cost.

Fashion and interior design

In every design field there are many similarities. By use of CBIR we can use similarity as a power tool. Previously stored images helps to design a new fabric in less time which is similar to old one. The pictures of interior design can give a prior look to decorate properly. CBIR is used by many professionals in this field. It reduces the workload in terms of time and cost and give better result. Possible failure is known in advance.

Journalism and advertising

It is another important area of CBIR application. In this field a large archives of photographs is maintained to understand articles or publicize copy. Broadcasting corporations use this to deal with large archive footage.

Cultural heritage

The ability to identify visually similar objects can be useful for art lovers who can easily get painting or sculptures of their interest. It also helps to researchers to know the historical influence.

Education and training

The availability of searchable video clips of good teaching material could reduce preparation time and improve teaching quality. It also help as a human tutor.

Geographical information and remote sensing systems

It is used to obtain information about mineral present underground. It is used by marketing and planning managers to search by spatial attribute. Physical geographers and agriculturalists use images for research and practical purposes like, study of ground for appropriate crops.

Medical diagnosis

It is a prime area of CBIR. It is helpful to identify the internal problem which is not visible. Radiology, histopathology, computerized tomography, various scanning are used in medical for diagnosis and their results are stored to get important information.

Entertainment

Entertainment is mostly photo or video based like TV programs and films. Channels stored images and videos for future use. Normal user can also want to store pictures of his interest. In the recent years CBIR developed in this field for mass user.

Web searching

Difficulty in locating images on the web leads to need for efficient image search tools. In this field there is a need to handle large collection of still images. In these days we can find image based searching which is rather image based but still improvement is required.

8. Conclusion

A new method using color and textural feature of image has been proposed for image searching. With small size of features, the size of feature vector of an image is kept small to reduce the retrieval time. Clustering with KNN technique performs well and gives overwhelming results. Relevance feedback can be used to improve its accuracy. Along with this proposed method this report reviewed content-based image retrieval in its early days, need of CBIR, primitive features used for information retrieval along with some noted approach to make it more efficient. This field has large potential of development and various research studies are still in progress to make it more fast and efficient.

References

[1] Arnold W.M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain, “Content-based image retrieval at the end of the early years,” *IEEE Transaction Pattern Analysis and Machine Intelligence*, Vol. 22, no. 12, pp. 1349-1380, Dec. 2000.

[2] Jing Huang, S. Ravi Kumar, Mandar Mitra, Wei-Jing Zhu, and Ramin Zabih, “Image indexing using color correlograms” *IEEE Computer Vision and Pattern Recognition*, pp 762-768. 1997.

[3] Timo Ojala, Matti PietikaÈinen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, no. 7, pp. 971-987, Jul. 2002.

[4] Jing Zhang, Gui-li Li, and Seok, “Texture-based image retrieval by edge detection matching GLCM,” *The 10th IEEE International Conference on High Performance Computing and Communications*, pp. 782-786, Sep 2008.

[5] Subrahmanyam Murala, Anil Balaji Gonde, and R. P. Maheshwari, “Color and texture features for image indexing and retrieval,” *IEEE International Advance Computing Conference*, pp. 1411-1416, Mar. 2009.

[6] Subrahmanyam Murala, R. P. Maheshwari, and R. Balasubramanian, “Local tetra patterns: A new feature descriptor for content-based image retrieval,” *IEEE Transactions on Image Processing*, Vol. 21, no. 5, pp. 2874-2886, May 2012.

[7] Cardoso, Douglas Natan Meireles, “Iterative Technique for Content-Based Image Retrieval using Multiple SVM Ensembles,” *J. Clerk Maxwell, A Treatise on Electricity and Magnetism,* 2: 68-73.

[8] Anna Saro Vijendran, S. Vinod Kumar, “A new content based image retrieval system by HOG of wavelet sub bands,” *International Journal of Image Processing, Signal Processing and Pattern recognition*, Vol. 8, no. 4, pp. 297-306, 2015.

[9] Kumar, A. Ramesh, and D. Saravanan, “Content based image retrieval using color histogram.” *International journal of computer science and information technologies,* 4.2 (2013).

[10] Kimaya S. Meshram, Ajay M. Agarkar, “Content based image retrieval systems using SIFT: A survey,” *SSRG International Journal of Electronics and Communication Engineering*, Vol. 2, Issue. 10, Oct 2012.

[11] Richa Jain, Sitesh Kumar Sinha, Mukesh Kumar, “A new image retrieval systems based on CBIR,” *International Journal of Emerging Technology and Advanced Engineering*, Vol. 5, Issue. 2, Feb 2015.

[12] Niraj S. Fegade, Dinesh D. Patil, “Content based image retrieval using combination of wavelet transform and color histogram,” *International Journal of Computer Science and Mobile Computing*, Vol.5 Issue.1, pp. 161-169, Jan. 2016.

[13] R. Durga Prasad, B.V.K Sai Kumar, K. Sai Ram, B. Veera Manoj, “Content based image retrieval using dominant color and texture features,” *International Journal for Modern Trends in Science and Technology*, Vol.2, Issue.4, ISSN 2455 – 3778, Apr. 2016.

[14] Apurva N. Ganar, C. S. Gode, Sachin M. Jambhulkar, “Enhancement of Color retrieval by using color, texture and shape features. for image indexing and retrieval,” *IEEE International Conference on Electronic Systems, Signal Processing and Computing Technologies*, 2014.

[15] M. E. Elalami, “A new matching strategy for content based image retrieval system,” Applied Soft Computing, pp. 407-418, 2014.

[16] M. E. Elalami, M. Esmel. “A novel image retrieval model based on the most relevant features,” *Knowledge-Based Systems* 24.1 (2011): 23-32.

[17] Lin, Chuen-Horng, Rong-Tai Chen, and Yung-Kuan Chan. “A smart content-based image retrieval system based on color and texture feature,” *Image and Vision Computing* 27.6 (2009): 658-665.

[18] Jhanwar, N., et al. “Content based image retrieval using motif cooccurrence matrix,” *Image and Vision Computing* 22.14 (2004): 1211-1220.

[19] Huang, Po-Whei, and S. K. Dai. “Image retrieval by texture similarity,” *Pattern recognition* 36.3 (2003): 665-679.

[20] http://wang.ist.psu.edu/docs/related