International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 07 | July 2018 www.irjet.net p-ISSN: 2395-0072

Content Based Image Retrieval (CBIR) Using Binary Clustering Approach

Neha Shrivastav[1], Asst. prof. vimal shukla[2]

[1]M.tech Scholar knp collage of science & Technology, Bhopal under RGPV Technical University, Bhopal
[2]vimal shukla knp collage of science & technology, Bhopal
Dept. of computer Science Engineering, Bhopal .Madhya Pradesh, India

---

Abstract - *After a decade of continuous development, CBIR technology has become more and more mature and starts playing a key role in human life. The ability to retrieve appropriate visual information without human assistance is still a complex, and interesting problem. To retrieve appropriate information from large image database, Content Based image retrieval (CBIR) is a popular approach. Content-based image retrieval involves a direct matching operation between a query image and a database of stored images. The three most common image matching features are colour, shape and texture. The fundamental challenge in image retrieval is to determine how low-level, pixel representation contained in a image can be efficiently and effectively processed to identify spatial relationships of colors and objects. In the proposed method binary clustering are used simultaneously on target and query images to retrieve color difference. In this work Geometric spreadness of each color also calculate using coordinate information of clusters and used it with color difference with some weighted. This thesis presents color feature extraction and similarity measure approaches CBIRC for content-based image retrieval.*

Key Words: CBIR, Image processing, RGB Color Model, Feature extraction, Intensity,HSV

1. INTRODUCTION

In the world of technology, information management is become very crucial. As the people know about information technology the use of the computer and digital devices has been increases. People download picture or capture image from digital camera and then upload on the internet. That means we can say that image data are generated very rapidly that create very large image databases. With the massive growth in the amount of visual information available, there exists a real need for systems to catalog and provide retrieval from digital image libraries. Image is described by visual features such as color, texture, shape, space and other features. The features can be classified as low-level feature and high-level features. Users can query example images based on these features. By similarity comparison the target image from the image repository is retrieved. Meanwhile, the next important phase today is focused on clustering techniques. Clustering algorithms can offer superior organization of multidimensional data for effective retrieval. It also allows nearest-neighbor search to be performed efficiently. An image is a group of pixel that represents the object or region in the image. Many color models are exist to represent the image like RGB, HSV, and HSL etc. RGB color model are simple model that uses the primary color (red, green, blue) to represent the color intensity value of the pixel in an image. A color image may be described as three layered image of Red, Green and Blue plane (Figure 1.1).

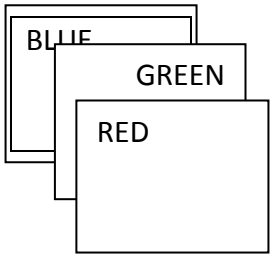

Figure 1.1
Three layer of RGB Color Model

2. PRELIMINARY OF CONTENT BASED IMAGE RETRIEVAL SYSTEM

Image retrieval is the fast growing and challenging research area with regard to both still and moving images. CBIR aims at searching image databases for specific images that are similar to a given query image. For this purpose various image processing technique have been used to improve the performance and result. There are many methods are proposed for image retrieval that are based on visual features such as colour, texture and shape [1].CBIR system are in its under development even though various image retrieval system has been developed like IBM`s QBIC [2], Webseek [3], etc. One reason is that the currently available methods are not completely fulfils the requirement .These requirement include functional requirement as well non functional requirement. Another reason is the semantic gap between low level and high level feature [4]. As the system is growing and to meet the requirements, these gaps have to be minimized so that it can provide more feature and functionality.

3.1 Image

An image may be defined as a two-dimensional function *f(x, y),* where *x* and *y* are spatial coordinates and the amplitude of *f* at any pair of coordinates *(x, y)* is called the intensity of the image at that point. Image may be continuous with

respect to the *x* and *y* coordinates and also in amplitude. Converting such an image to digital form requires that the coordinates and amplitude, be digitized through sampling and quantization respectively.

When *x, y* and the amplitude values are finite and discrete quantities the image is a digital image. A digital image can be defined as a two-dimensional digital space that contains intensity or color information arranged along an *x* and *y* spatial axis. As shown in the equation below the right side of equation represent a digital image and each element of this array is called an image element, picture element or pixel.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

3. APPROACHES FOR CBIR

The approaches considered so far for content-based image retrieval can be classified as one of these three types [5].

1. Manual annotation,
2. Automatic feature extraction and retrieval, and
3. Combinations of both.

In traditional retrieval systems features are added manually, e.g. adding text strings describing the content of an image. These systems require too much manpower taking into account the amount of image data available nowadays. Additionally the growth of available image data is faster than annotations can be added.

3.2 Color

Color is one of the most prominent perceptual features, because it more understandable and recognizable for the human eye in the image. Most commercial CBIR systems include color as one of the features such as QBIC, WebSeek etc. Generally CBIR systems first extracts color feature from the image than calculate histogram and then finding the distance between the images. Color is also used with other feature for the information retrieval from the image. Images characterized by color features have many advantages [6].

• **Robustness:** The color histogram is invariant to rotation of the image on the view axis, and changes in small steps when rotated otherwise or scaled [7]. It is also insensitive to changes in image and histogram resolution and occlusion.

• **Effectiveness:** There is high percentage of relevance between the query image and the extracted matching images.

• **Implementation simplicity:** The construction of the color histogram is a straightforward process, including scanning the image, assigning color values to the resolution of the histogram, and building the histogram using color components as indices.

• **Computational simplicity:** The histogram computation has *O*(X, Y) complexity for images of size X × Y. The complexity for a single image match is *linear*(*n*), where *n* represents the number of different colors, or resolution of the histogram.

• **Low storage requirements:** The color histogram size is significantly smaller than the image itself, assuming color quantization.

3.3 Texture

Texture is another important property of images. Texture refers to the visual patterns that have properties of homogeneity that do not result from the presence of only a single color or intensity. It is an innate property of virtually all surfaces, including cloud, trees, bricks, hair, fabric, etc. It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment.

3.4 Shape

Shape is an important visual feature and it is one of the primitive features for image content description. Shape based image retrieval is the measuring of similarity between shapes represented by their features. Shape content description is difficult to define because measuring the similarity between shapes is difficult. Shape descriptors can be divided into two main categories: region based and contour-based methods. Region-based methods use the whole area of an object for shape description, while contour-based methods use only the information present in the contour of an object [6].

3.5 Image Similarity Measure

There are many techniques that have been proposed by many researchers to measure the distance between two images e.g. Euclidian distance, Root mean distance etc. Each metric has some important characteristics related to an application.

3.6 Euclidean distance: The Euclidean distance between two n-dimensional (row or column) vectors *x* and *y* is defined as the scalar.

$$D_{Euclidian}(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

3.7 Manhattan distance: The Manhattan distance between two items is the sum of the differences of their corresponding components.

$$D_{Manhattan}(x,y) = \sum_{i=1}^{n} |x_i - y_i|$$

Where n is the number of variables, and x_i and y_i are the values of the i^{th} variable, at points x and y respectively.

3.8 Earth Movers Distance: Distance between two distributions is measured by Earth Mover's Distance [8]. Consider a metric space X endowed with distance function d_X. Then, for two multi-sets $A, B \subseteq X$, of size $s = |A| = |B|$, the earthmover distance (EMD) between A and B is defined as

$$EMD_X(A,B) = \frac{1}{s} \min_{\emptyset: A \to B} \sum_{x \in A} d_X(x, \emptyset(x))$$

Where the minimum is taken over all bisections $\emptyset: A \to B$.

EMD is based on the minimal cost that must be paid to transform one distribution into another.EMD matches perceptual similarity well and can operate on variable-length representations of the distributions; it is suitable for region-based image similarity measure [9], [10] but EMD is based on simplex method, and suffer from high complexity.

CHIC Method: In [9] author use CHIC method to compare the histogram by clustering, to measure between query image histogram and target image histogram. Here both histograms are based on color distribution of images, so they are different in size and also have different intervals between histogram bins. Due to heterogeneity of histogram Euclidian distance or manhattans distance are unable to perform similarity measurement. CHIC method in first step performs clustering of color feature. In second step it finds discrepancies between all clusters, and then adds all discrepancies to find distance between images. Compare to EMD, CHIC method is fast.

4. PROPOSED METHOD

For effective image retrieval clustering are used in extensively manner by many researcher. In the proposed method called "Content Based Image Retrieval using Clustering" (CBIRC) we use binary clustering to cluster the color data. First we apply preprocessing steps that involve resize the image in equal size and some suitable format so that color processing would be easy and effectively work on the all images. Let I_q is the quarry image i.e. image to be search and I_t is the image from the image database i.e. target image. After preprocessing all three matrix plane of I_q and I_t are combined row by row and considered as a single image matrix I as shown in Figure 4.1 and 4.2. Then arrange these matrix planes in one column index form using (1) such that entries of each column sequentially fill up by rows i.e. first quarry image are placed than after target image are place that we call image I. On this image matrix I apply clustering.

To identify any pixel belongs to I_q or I_t a variable d_p are used that store total number of pixel in I_q or I_t. In matrix plane I if the index value of any pixel is greater than d_p then it belong to I_t otherwise it belong to I_q. Then apply binary clustering on the data set S so that S_i number of clusters are generated and each S_i contain sub cluster S_{i_q} and S_{i_t}.

The sub cluster S_{i_q} and S_{i_t} contains quarry image and target image pixels respectively. After the clustering process we find the difference between the images by absolutely adding the discrepancy of each cluster.

This method also save the index value in the index cluster G_i. G_i is generated corresponding to each and every color value of the cluster S_i called index cluster and used for retrieving the geometrical feature of the image. On these cluster we calculate geometric mean and geometric standard deviation and then find the geometric difference between the images.

The index clusters are stored corresponding to color cluster of image I. We classify the color data in two class say x_0 and x_1. Actually by doing this we are finding out the hyper-plane that will divide the color data set S in x_0 and x_1.

As shown in Figure 4.2 quarry image pixel $R_{(x_i y_i)_q}$, $G_{(x_i y_i)_q}$ and $B_{(x_i y_i)_q}$ and target image $R_{(x_i y_i)_t}$, $G_{(x_i y_i)_t}$ and $B_{(x_i y_i)_t}$ respectively are combined and treated as a single image I.

Coordinate Index Conversion: Any pixel value (x_i, y_i) are converted in index form. If M is the matrix with n rows and m columns than index formulas are

$$I = m * (x - 1) + y \quad (1)$$

$$x = (I - y)/(m + 1) \quad (2)$$

$$y = m \bmod I \quad when\ m \bmod I > 0 \quad (3)$$

$$m \quad otherwise$$

Where I is the index value, x is the row of the element i.e. x coordinate and y is the column of element i.e. y coordinate.

R_{11}	R_{12}	R_{12}
R_{21}	R_{22}	R_{23}
R_{31}	R_{32}	R_{33}

Red

G_{11}	G_{12}	G_{13}
G_{21}	G_{22}	G_{23}
G_{31}	G_{32}	G_{33}

Green

B_{11}	B_{12}	B_{13}
B_{21}	B_{22}	B_{21}
B_{31}	B_{32}	B_{33}

Blue

Quarry image

R'_{11}	R'_{11}	R'_{11}
R'_{21}	R'_{21}	R'_{21}
R'_{31}	R'_{31}	R'_{31}

Red

G'_{11}	G'_{11}	G'_{11}
G'_{11}	G'_{11}	G'_{11}
G'_{11}	G'_{11}	G'_{11}

Green

B'_{11}	B'_{11}	B'_{11}
B'_{11}	B'_{11}	B'_{11}
B'_{11}	B'_{11}	B'_{11}

Blue

Target Image

Figure 4.1 RGB planes of Quarry Image and Target Image

For the input color data set *S* we use variance σ_t as a threshold value. One can decide value of σ_t empirically. For splitting any cluster S_i, the variance σ_i of that cluster must be greater than σ_t. To divide the data set *S* of image in two subsets S_k and S_{k+1} assign some index value k_i for each and every pixel p_i of image *I*. These index values are also used to calculate geometric difference.

Red **Green** **Blue**

R_{11}	R_{12}	R_{12}	G_{11}	G_{12}	G_{13}	B_{11}	B_{12}	B_{13}
R_{21}	R_{22}	R_{23}	G_{21}	G_{22}	G_{23}	B_{21}	B_{22}	B_{21}
R_{31}	R_{32}	R_{33}	G_{31}	G_{32}	G_{33}	B_{31}	B_{32}	B_{33}
R'_{11}	R'_{11}	R'_{11}	G'_{11}	G'_{11}	G'_{11}	B'_{11}	B'_{11}	B'_{11}
R'_{21}	R'_{21}	R'_{21}	G'_{11}	G'_{11}	G'_{11}	B'_{11}	B'_{11}	B'_{11}
R'_{31}	R'_{31}	R'_{31}	G'_{11}	G'_{11}	G'_{11}	B'_{11}	B'_{11}	B'_{11}

Figure 4.2 Combined Quarry Image and Target Image

Initially *S* contains all color values of image *I*. To split image cluster *S* in two classes calculate the mean and variance of the Image *S*. If the variance σ_i is greater than some threshold variance σ_t then it is a splitable cluster thus split this cluster in S_k and S_{k+1}. This process is repeats until we meet all the clusters are unsplitable.

Red Column **Green Column** **Blue Column**

R_{11}	G_{11}	B_{11}
R_{12}	G_{12}	B_{12}
R_{13}	G_{13}	B_{12}
R_{21}	G_{21}	B_{21}
R_{22}	G_{22}	B_{22}
R_{23}	G_{23}	B_{23}
R_{31}	G_{31}	B_{31}
R_{32}	G_{32}	B_{32}
R_{33}	G_{33}	B_{33}
R'_{11}	G'_{11}	B'_{11}
R'_{12}	G'_{12}	B'_{12}
R'_{13}	G'_{13}	B'_{12}
R'_{21}	G'_{21}	B'_{21}
R'_{22}	G'_{22}	B'_{22}
R'_{23}	G'_{23}	B'_{23}
R'_{31}	G'_{31}	B'_{31}
R'_{32}	G'_{32}	B'_{32}
R'_{33}	G'_{33}	B'_{33}

Figure 4.3 Converted RGB planes in single column format

Now calculate mean $\bar{x}_m$ for image *I*.

$$\bar{x}_m = \frac{1}{n}\sum_{i=1}^{n}(R_i, G_i, B_i)$$

$$\bar{x}_m = \frac{1}{n}\sum_{i=1}^{n} p_i$$

Than calculate difference D_i between $\bar{x}_m$ and x_i where x_i represents the color value of (R_i, G_i, B_i) for pixel p_i.

$$D_i = (x_i - \bar{x}_m)$$

$$D_i = (R_i - R_m) + (G_i - G_m) + (B_i - B_m)$$

If the D_i is negative than the pixel are placed in sub cluster S_k otherwise it will placed in sub cluster S_{k+1}.Than calculate variance of cluster S_k and S_{k+1} i.e. σ_{s_k} and $\sigma_{s_{k+1}}$ .We first split that cluster whose variance is maximum. Suppose $\sigma_{s_{k+1}}$ is greater than σ_{s_k} and σ_t then split this cluster. Otherwise if $\sigma_{s_{k+1}}$ is not greater than σ_t than S_{k+1} will not split and the splitting process will stop. Similarly if σ_{s_k} is greater than σ_t then split S_k. This procedure repeats for each cluster S_{k+n} until we meet all the unsplitable clusters i.e. the variance of all cluster is less than threshold value σ_t. Each cluster represents the same type colors of the image *I*. Then find the discrepancy of each cluster.

$$Dis_i = \left| car(S_{i_t}) - car(S_{i_q}) \right|$$

$$Color\ Difference = \sum_{i=1}^{n} Dis_i$$

By adding discrepancy of all clusters we get the color difference between the I_q and I_t.

In any cluster S_i some pixel are from quarry image and some pixel are from target image. Corresponding to each cluster S_i, index cluster G_i is generated that contain index value k_i corresponding to each pixel p_i. The index value k_i is converted in coordinate (x_i, y_i) using (2) and (3).

For x and y co-ordinate of image I_t, and are the geometric mean respectively. $\bar{x}_{gm_{t_i}}$ $\bar{y}_{gm_{t_i}}$

$$\bar{x}_{gm_{t_i}} = \sqrt[car(GS_{t_i})]{\prod_{\forall x_{t_l} \in GS_{t_i}} x_{t_l}}$$

$$\bar{y}_{gm_{t_i}} = \sqrt[car(GS_{t_i})]{\prod_{\forall y_{t_i} \in GS_{t_i}} x_{t_m}}$$

Similarly for image I_q geometric mean $\bar{x}_{gm_{q_i}}$ and $\bar{y}_{gm_{q_i}}$ will be

$$\bar{x}_{gm_{q_i}} = \sqrt[car(GS_{q_i})]{\prod_{\forall x_{q_i} \in GS_{q_i}} x_{q_l}}$$

$$\bar{y}_{gm_{q_i}} = \sqrt[car(GS_{q_i})]{\prod_{\forall y_{q_i} \in GS_{q_i}} x_{q_m}}$$

Where $car(GS_{t_i})$ and $car(GS_{q_i})$ are the cardinality of index sub cluster GS_{t_i} and GS_{q_i} respectively.

For each and every unsplitable cluster calculate the geometric standard deviation (GSTD) with respect to geometric mean of quarry image pixel cluster and target image pixel cluster that are considering as a sub clusters of the cluster S_i. Geometric Standard Deviation $GSTD_{t_i}$ for I_t will be

$$GSTD_{t_i} = \sqrt{\sum_{\forall x_{t_l} \in GS_{t_i}} \left(x_{t_l} - \bar{x}_{gm_{t_i}}\right)^2 + \sum_{\forall y_{t_m} \in GS_{t_i}} \left(y_{t_m} - \bar{y}_{gm_{t_i}}\right)^2}$$

and $GSTD_{q_i}$ for I_q

$$GSTD_{q_i} = \sqrt{\sum_{\forall x_{q_l} \in GS_{q_i}} \left(x_{q_l} - \bar{x}_{gm_{q_i}}\right)^2 + \sum_{\forall y_{q_m} \in GS_{q_i}} \left(y_{q_m} - \bar{y}_{gm_{q_i}}\right)^2}$$

Than calculate the difference between geometric standard deviation of the sub cluster. Do this for all unsplitable clusters and by absolutely adding their difference that show the geometrical color discrepancy.

$$GD_i = \left\{1 - \frac{Dis(S_i)}{car(S_i)}\right\} * \left\{abs\left(GSTD_{t_i} - GSTD_{q_i}\right)\right\}$$

$$GD_{diff} = \sum_i GD_i$$

Adding this Geometrical color discrepancy and color difference of the image will be the total difference between the quarry image and target image. The Geometrical color discrepancy can be represented as a Geometrical color difference or in simple word Geometrical difference represented by G_m, thus

$$Total\ Difference = Color\ diffrence + \alpha . Gm$$

The value of is α is selected between zero and one.

In this method geographical information of pixel are also used to classify images more effectively and results are more accurate.

Algorithm

Input Parameter: Quarry Image I_q, Target Image I_t

Output: Difference between Quarry Image I_q and Target Image I_t

Step 1 : Combined all three matrix plane (Red, Green, Blue) of Quarry Image I_q and Target Image I_t row wise and make new matrix I.

Step 2 : Arrange each plane of I in one column index form such that entries of column sequentially fill up by rows.

Step 3 :
Step (i): Find a splitable subcluster whose Variance is maximum.

Step (ii): Split the subcluster into two new subclusters using Binary Clustering.

Step 4 : If the Variance of selected cluster is not greater than Threshold Variance than it is not splitable and mark it.

Step 5 : Corresponding to each cluster, generate Index Cluster that contain Index Value for each pixel in the cluster.

Step 6 : Repeat step 4, 5 and 6 until the Variance in each subcluster is below from Threshold Variance.

Step 7 : For all marked unsplitable cluster Calculate Discrepancy of cluster.

Step 8 : Calculate Color Difference.

Step 9 : In the Index Cluster convert each index value in the *(x,y)* co-ordinate.

Step 10: Calculate Geometric Mean for each *x* and *y* co-ordinate in Quarry and Target Image.

Step 11: Calculate the Geometric Standard Deviation (GSTD) with respect to Geometric Mean of Quarry Image Pixel Cluster and Target Image Pixel Cluster.

Step 12: Calculate Geometric Color Discrepancy for each Cluster.

Step 13: Calculate Geometric Difference (GD)

Step 14: Calculate Total Difference by adding all the Color Difference and Geometric Discrepancy of colour.

Step 15: Return Total Difference between I_q and I_t.

In Figure 4.4 show block diagram of image retrieval using CBIRC. Query image and target image form database resize in the same size. After that they combine in specific manner and make a whole image and extract the color feature and other index information from combined image.

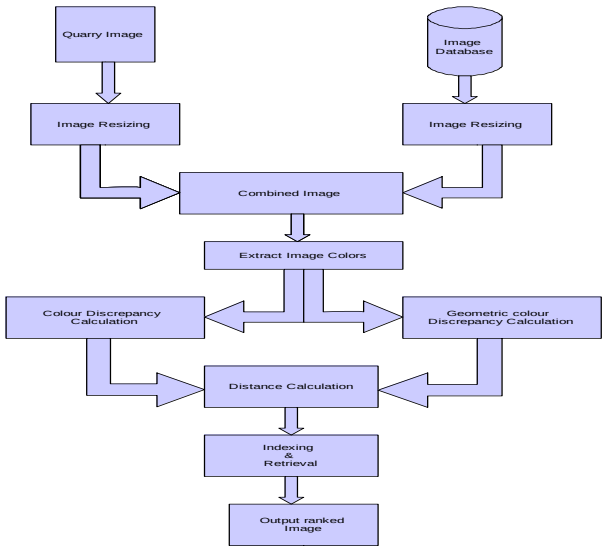

Figure 4.4 Image Retrieval Using CBIRC

Then find colour discrepancy and difference of target and object image with the geometric colour discrepancy .All images form database taken one by one and count their corresponding distance. On the basis of their Colour difference and Geometric difference indexing is perform in descending order. In next stage images are montage as a thumbnail according to indexing.

5. EXPERIMENT AND RESULTS

We perform our experimental result on Intel Core 2 Duo CPU 2.10 GHz, 2GB RAM and Windows XP 32 bit operating system. We develop our coding in MATLAB 7.6.0.324(R 2008 a). In our experiment we take 600 different landscape images from Google search engine. We divide all images in six appropriate categories manually. Each category has 100 images. So each image belongs to specific class. These classes are sunset, sea, farm, large stone, desert and night. Types of image and there categorization are important because precision graph based on image category. So due to this reason we take landscape images such that, there is sufficient visual difference between different categories of images. Typical images are shows in Figure 5.1.

Figure 5.1 Categories of Images

We compare visual and graphical result between ACE (for variable count i.e.VC) and CBIRC. Precision graph of different queries may have too much fluctuation , so to get the perfect concluding result we use average precision graph .We examine both process ACE & CBIRC for separate query of different class. Result show that CBIRC return good result in 90% cases with ACE (VC).

Figure 5.2 show a query image of class "desert". According to query image figure 5.3 shows a retrieval result for ACE variable count method in which 42 same

Figure 5.2 Quarry image

Class images are retrieve out of 80 images. So retrieval efficiency is (42/80*100) =52.5%.The distance of retrieved images from the quarry image are shown in Table 5.1 for ACE method

Figure 5.3 Result of ACE (VC)

Table 5.1 Distance of retrieved image from the quarry image respectively for ACE (VC)

S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance
1	0.0000	2	0.5020	3	0.5536	4	0.5542	5	0.5585	6	0.5654	7	0.5669	8	0.5689
9	0.5804	10	0.5857	11	0.5917	12	0.5956	13	0.5962	14	0.6089	15	0.6399	16	0.6429
17	0.6552	18	0.6622	19	0.6631	20	0.6634	21	0.6675	22	0.6690	23	0.6817	24	0.6841
25	0.6871	26	0.6874	27	0.7004	28	0.7033	29	0.7074	30	0.7166	31	0.7205	32	0.7278
33	0.7296	34	0.7296	35	0.7362	36	0.7363	37	0.7365	38	0.7377	39	0.7392	40	0.7394
41	0.7498	42	0.7512	43	0.7532	44	0.7552	45	0.7585	46	0.7588	47	0.7591	48	0.7592
49	0.7602	50	0.7606	51	0.7615	52	0.7618	53	0.7620	54	0.7623	55	0.7641	56	0.7646
57	0.7660	58	0.7675	59	0.7697	60	0.7702	61	0.7702	62	0.7702	63	0.7702	64	0.7715
65	0.7732	66	0.7733	67	0.7736	68	0.7751	69	0.7780	70	0.7791	71	0.7803	72	0.7816
73	0.7825	74	0.7831	75	0.7923	76	0.7926	77	0.7931	78	0.7940	79	0.7958	80	0.7974

Figure 5.4 shows a, retrieval result for CBIRC in which 55 same class images are retrieved out of 80 images. So retrieval efficiency is (55/80*100)= 68.75%.So for given specific query CBIRC is 16.25% better than ACE (VC).The distance of retrieved images from quarry image are shown in Table 5.2 for CBIRC .

Figure 5.4 Result of CBIRC

Table 5.2 Distance of retrieved image from the quarry image respectively for CBIRC

S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance	S No	Distance
1	0	2	8198	3	8216	4	8932	5	9158	6	9264	7	9434	8	9550
9	9820	10	9922	11	10000	12	10416	13	10492	14	10666	15	10742	16	10812
17	10948	18	10876	19	10936	20	11004	21	11008	22	11188	23	11392	24	11494
25	11424	26	11428	27	11432	28	11472	29	11522	30	11602	31	11604	32	11616
33	11830	34	11882	35	11906	36	12192	37	12256	38	12278	39	12324	40	12330
41	12350	42	12366	43	12404	44	12442	45	12484	46	12550	47	12568	48	12620
49	12684	50	12758	51	12788	52	12826	53	12902	54	12966	55	13006	56	13008
57	13202	58	13336	59	13360	60	13372	61	13390	62	13394	63	13442	64	13452
65	13514	66	13524	67	13534	68	13610	69	13728	70	13740	71	13750	72	13764
73	13788	74	13790	75	13808	76	13822	77	13822	78	13886	79	13908	80	14016

5.1 Average precision graph comparison

The performance of a color extraction method is measured in terms of average retrieval precision. Each time a query image taken from database and retrieve k best matched image from database. The average retrieval precision η is defined according to [9] as follows

$$\eta = \frac{\sum_{i=1}^{j} n_i}{\sum_{i=1}^{j} k}$$

Where j is the no images in the database and n_i is the number of returned image falling into correct image class of quarry image i. Figure 5.5 shows the precision comparison

graph of CBIRC and ACE in case of VC. We compare ACE and CBIRC on variances 500, 1000, 1500, 2000, 2500. ACE shows corresponding precision values 0.7062, 0.7024, 0.6911, 0.6663, 0.5812 and for CBIRC 0.8662, 0.8524, 0.8211, 0.8073, 0.7012. Here CBIRC show 14 % better result from ACE.

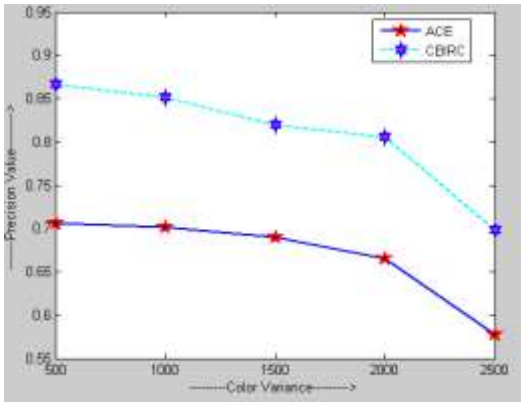

Figure 5.5 Precision comparisons between CBIRC and ACE (VC)

Our Proposed method CBIRC gives good result as compare to ACE. The main region of increased retrieval result`s performance is that in the CBIRC, we cluster all the colors of both image at the time of image color extraction, on the other hand in ACE clusters of color feature for each separate image are made by CHIC. After that CHIC find discrepancies between all clusters and add all discrepancy. Color cluster perform more robust clustering comparing to feature color clustering, because features are representative color, not true color. This implies summation of discrepancies of color cluster is best estimation of similarity measure of image compare to, summation of discrepancies of color feature cluster. In CBIRC geometric color discrepancy also used to calculate the difference between images. So above fact implies the performance of CBIRC increases as with number of colors increases in images.

6. CONCLUSION AND FUTURE WORK

For Content Based Image retrieval various image feature are used to compare the image. The number of features may vary as per requirements. More the feature are selected there are chances that results are closer and accurate but it may lead to increase complexity. In this thesis a method for content based image retrieval are proposed. Clustering has been used from many years for image segmentation and image retrieval. In this work binary clustering are used for grouping color data .In this method we apply binary clustering simultaneously on target and query images by combining and treating both images as a single image. Binary clustering generates color and their corresponding index cluster. On the basis of these clusters we calculate color and geometric spreadness differences of corresponding colors. Lastly we add these two differences with the appropriate weighted that is the total difference between images. The algorithms are tested on 600 landscape images. We compare proposed method with ACE on all images to calculating average retrieval performance. Experiment result show significant improvement of proposed method over ACE method.

A number of significant issues related to the CBIR have been addressed in this work. However, there are still a number of possible improvements that require further investigation. There are also a number of new directions in which the presented work can be employed.

(1) To improve proposed algorithm by using proper preprocessing of images such as contrast enhancement, noise removal etc.

(2) To speed up the algorithm perform similarity measure on color feature in place of colors.

(3) By getting regional feature into account and combined with color feature to improve algorithm performance.

(4) To get a better performance, the system can automatically pre-classified the database into different semantic images and develop algorithm that are specific for particular semantic image class.

(5) On the place of binary clustering another clustering can be used.

REFERENCES

1. Sharmin Siddique, “A Wavelet Based Technique for Analysis and Classification of Texture Images,” Carleton University, Ottawa, Canada, Project Report 70.593, April 2002.

2. W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, G. Taubin “The QBIC Project: Querying Images By Content Using Color, Texture, and Shape” SPIE Vol. 1908 (1993) , pp : 173-187.

3. John R. Smith, Shih-Fu Chang “Visually Searching the Web for Content” IEEE Multimedia , pp: 12-20, July-September 1997.

4. Nuno Vasconcelos,” From Pixels to Semantic Spaces: Advances in Content-Based Image Retrieval” IEEE Computer Society, pp: 20-26, , July 2007

5. S. Siggelkow, H. Burkhardt,” Fast invariant feature extraction for image retrieval”, In R. C. Veltkamp, H. Burkhardt, and H.-P. Kriegel, editors, Stateof- the-

Art in Content-Based Image and Video Retrieval, Kluwer, pp: 43–68, 2001.

6. Ryszard S. Chora´s," Image Feature Extraction Techniques and Their Applications for CBIR and Biometrics Systems", International Journal Of Biology And Biomedical Engineering, Vol. 1, pp: 6-16, 2007.

7. Zhenhua Zhang, Wenhui Li, Yinan Lu,"Novel Color Feature Representation and Matching Technique for Content-based Image Retrieval" , pp: 118 - 122 IEEE , 2009.

8. Alexandr Andoni , Piotr Indyk , Robert Krauthgamer," Earth Mover Distance over High-Dimensional Spaces, October 11 ,2007 "

9. Wei-Ta Chen, Wei-Chuan Liu, and Ming-Syan Chen, "Adaptive Color Feature Extraction Based on Image Color Distributions" , IEEE Transactions on image processing, vol. 19, no. 8, pp:2005-2016, August 2010.

10. Soo-Chang Pei, Ching-Min Cheng "Color Image Processing by Using Binary Quaternion-Moment-Preserving Thresholding Technique" IEEE transactions on image processing, vol. 8, No. 5, pp: 616-628, May 1999.

11. Jiawei Han, Micheline Kamber," Data Mining: Concepts and Techniques".

12. Ian H. Witten, Eibe Frank," Data Mining, Practical Machine Learning Tools and Techniques".

13. A.K. Jain, M.N. Murty, P.J. Flynn," Data Clustering: A Review" ACM Computing Surveys, Vol. 31, No. 3, pp: 264-323, September 1999.

14. Introduction to Data Mining by Pang-Ning Tan, Michael Steinbach, Vipin Kumar.

15. I. El-Feghi, H. Aboasha, M. A. Sid-Ahmed, M. Ahmadi "Content-Based Image Retrieval Based on Efficient Fuzzy Color Signature", IEEE Conference, pp: 1118 - 1124, 2007.