Augmentation of Block Truncation Coding based Image Retrieval by using Even and Odd Images with Sundry Colour Spaces

Dr. H.B.Kekre
Senior Professor
Computer Engg. Dept.,
MPSTME,
SVKM's NMIMS University,
Mumbai, India

Sudeep D.Thepade
Associate Professor &
Ph.D.Research Scholar,
Computer Engg. Dept.,
MPSTME,
SVKM's NMIMS University,
Mumbai, India

Varun Lodha, Pooja Luthra,
Ajoy Joseph, Chitrangada
Nemani
B.Tech (IT) Students, MPSTME,
SVKM's NMIMS University,
Mumbai, India

ABSTRACT
The augmentation to block truncation coding (BTC) based image retrieval techniques using Even and Odd images with ten different colour spaces is the theme of work given in the paper. Here the original image is reflected across vertical axis to obtain the flip image, then even and odd images are obtained respectively by addition of original with flip and subtraction of flip from original. The BTC is applied on original image, even image and odd image to get seven different combinational feature sets for content based image retrieval (CBIR) techniques like original, even, odd, original & even, original & odd, even & odd and original & even & odd. Use of ten sundry colour spaces results into total seventy CBIR methods, For experimentation the generic image database having 1000 images spread across 11 categories is used. For each proposed CBIR technique 55 queries (5 per category) are fired on the generic image database. To compare the performance of image retrieval techniques average precision and recall are computed of all queries. The results have shown the performance improvement (higher precision and recall values) with these proposed colour-BTC methods. Instead of using just 6 feature vector in BTC, if we perform the image retrieval using the flipping technique wherein the feature vector is increased to 12 and 18,the performance also increases except in the case of normalized rgb colour space. Image flipping helps to improve the performance in all of luminance-chromaticity colour spaces (YUV, YIQ, LUV, Kekre's YCgCb, YCbCr) as well as non-luminance based colour spaces (XYZ,HSI,RGB,HSV) in comparison of BTC applied on original image. Also overall YUV colour space proves to be the best in all colour spaces for proposed image flipping techniques. The second best performance is given by Kekre's YCGCb colour space.

Keywords
CBIR, BTC, Colour Space, RGB, HSV, XYZ, HSI, rgb, Kekre's LUV, YCbCr, YUV, YIQ, Kekre's YCgCb.

1. INTRODUCTION

The large numbers of images which are being generated from a variety of sources (digital camera, digital video, scanner, the internet etc.) have posed technical challenges to computer systems to store/transmit and index/manage image data effectively to make such collections easily accessible. Image compression deals with the challenge of storage and transmission, where significant advancements have been made [1,4,5]. The challenge to image indexing is studied in the context of image database [2,6,7,10,11], has become one of the promising and important research area for researchers from a wide range of disciplines like computer vision, image processing and database areas. The thirst for better and faster image retrieval techniques is increasing day by day.

Problems with traditional methods of image indexing have led to the rise in techniques for retrieving images on the basis of automatically derived features such as colour, texture and shape- a technology now referred as Content-Based Image Retrieval (CBIR).

Some of important applications for CBIR technology could be identified as art galleries [12,14], museums, archaeology [3], architecture design [8,13], geographic information systems [5], weather forecast [5,22], medical imaging [5,18], trademark databases [21,23], criminal investigations [24,25], image search on the Internet [9,19,20].

1.1 Content Based Image Retrieval

A Content Based Image Retrieval (CBIR) is an interface between a high level system (the human brain) and a low level system (a computer). The human brain is capable of performing complex visual perception, but is limited in speed while a computer is capable of limited visual capabilities at much higher speeds. In a CBIR, features are used to represent the image content. The features are extracted automatically and there is no manual intervention, thus eliminating the dependency on humans in the feature extraction stage. These automated approaches to object recognition are computationally expensive, difficult and tend to be domain specific.

The typical CBIR system performs two major tasks [16,17]. The first one is feature extraction (FE), where a set of features, called feature vector, is generated to accurately represent the content of each image in the database. The second task is similarity measurement (SM), where a distance between the query image and each image in the database using their feature vectors is used to retrieve the top "closest" images [16,17,26]. For feature extraction in CBIR there are mainly two approaches [5] feature extraction in spatial domain and feature extraction in transform domain. The feature extraction in spatial domain includes the CBIR techniques based on histograms [5], BTC [1,2,16], VQ [21,25,26]. The transform domain methods are widely used in image compression, as they give high energy compaction in transformed image [17,24]. So it is obvious to use images in transformed domain for feature extraction in CBIR [23]. But taking transform of image is time consuming and also needs all images of database to be of same size to get similar feature vectors. This limitation is overcome in CBIR using Block Truncation Coding (BTC).

2. BLOCK TRUNCATION CODING (BTC)

Block truncation coding (BTC) is a relatively simple image coding technique developed in the early years of digital imaging more than 29 years ago. Although it is a simple technique, BTC has played an important role in the history of digital image coding in the sense that many advanced coding techniques have been developed based on BTC or inspired by the success of BTC [1]. Block Truncation Coding (BTC) was first developed in 1979 for grayscale image coding. This method first divides the image to be coded into small non-overlapping image blocks (typically of size 4× 4 pixels to achieve reasonable quality.) The small blocks are coded one at a time. For each block, the original pixels within the block are coded using a binary bit-map the same Upper Mean Colour as the original blocks and two mean pixel values. In the original implementation the block mean and the variance of the pixels are used to preserve the first and second moment of the blocks. The descriptors here follow a later version of BTC, which was shown to give better performance [1].

The method first computes the mean pixel value of the whole block and then each pixel in that block is compared to the block mean. If a pixel is greater than or equal to the block mean, the corresponding pixel position of the bitmap will have a value of 1 otherwise it will have a value of 0. Two mean pixel values one for the pixels greater than or equal to the block mean and the other for the pixels smaller than the block mean are also calculated. At decoding stage, the small blocks are decoded one at a time. For each block, the pixel positions where the corresponding bitmap has a value of 1 is replaced by one mean pixel value and those pixel positions where the corresponding bitmap has a value of 0 is replaced by another mean pixel value.

It was quite natural to extend BTC to multi-spectral images such as colour images. Most colour images are recorded in RGB space, which is perhaps the most well-known colour space. As described previously, BTC divides the image to be coded into small blocks and code them one at a time. For single bitmap BTC of colour image, a single binary bitmap of the same size as the block is created and two colours are computed to approximate the pixels within the block. To create a binary bitmap in the RGB space, an inter band average image (IBAI) is first created and a single scalar value is found as the threshold value. The bitmap is then created by comparing the pixels in the IBAI with the threshold value [1].

2.1 CBIR using BTC-RGB

In original BTC we divide the image into R, B, and G components and compute the interband average image (IBAI) which is the average of all the components(R, G, and B) and mean of interband average image is taken as threshold. But the disadvantage of this method is that if one of the component is prominent than the other component then that component dominates the threshold value, reducing the effect of other component. A more general approach is by using three independent R, G and B components of colour images to calculate three different thresholds and then apply BTC to each individual R, G and B planes. Let the thresholds be TR, TG and TB, which could be computed as per the equations given below.

$$TR = \frac{1}{m*n} \sum_{i=1}^{m} \sum_{j=1}^{n} R(i,j) \tag{1}$$

$$TG = \frac{1}{m*n} \sum_{i=1}^{m} \sum_{j=1}^{n} G(i,j) \tag{2}$$

$$TB = \frac{1}{m*n} \sum_{i=1}^{m} \sum_{j=1}^{n} B(i,j) \tag{3}$$

Here three binary bitmaps will be computed as BMr, BMg and BMb. If a pixel in each component (R, G, and B) is greater than or equal to the respective threshold, the corresponding pixel position of the bitmap will have a value of 1 otherwise it will have a value of 0.

$$BMr(i,j) = \begin{cases} 1, if \ldots R(i,j) >= TR \\ 0, \ldots if \ldots R(i,j) < TR \end{cases} \quad (4)$$

$$BMg(i,j) = \begin{cases} 1, if \ldots G(i,j) >= TG \\ 0, \ldots if \ldots G(i,j) < TG \end{cases} \quad (5)$$

$$BMb(i,j) = \begin{cases} 1, if \ldots B(i,j) >= TB \\ 0, \ldots if \ldots B(i,j) < TB \end{cases} \quad (6)$$

Two mean colours one for the pixels greater than or equal to the threshold and other for the pixels smaller than the threshold are also calculated [15]. The upper mean colour UM (Rm1, Gm1, Bm1) is given as follows.

$$Rm1 = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMr(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} BMr(i,j) * R(i,j) \quad (7)$$

$$Gm1 = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMg(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} BMg(i,j) * G(i,j) \quad (8)$$

$$Bm1 = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMb(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} BMb(i,j) * B(i,j) \quad (9)$$

And the Lower Mean LM= (Rm2, Gm2, Bm2) is computed as following equations

$$Rm2 = \frac{1}{m*n - \sum_{i=1}^{m}\sum_{j=1}^{n} BMr(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMr(i,j)\} * R(i,j) \quad (10)$$

$$Gm2 = \frac{1}{m*n - \sum_{i=1}^{m}\sum_{j=1}^{n} BMg(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMg(i,j)\} * G(i,j) \quad (11)$$

$$Bm2 = \frac{1}{m*n - \sum_{i=1}^{m}\sum_{j=1}^{n} BMb(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMb(i,j)\} * B(i,j) \quad (12)$$

These Upper Mean and Lower Mean together will form a feature vector or signature of the image. For every image stored in the database these feature vectors are computed and stored in feature vector table.

2.2 CONSIDERED COLOUR SPACES

As applied to RGB colour space the Block Truncation Coding (BTC) can also applied to other colour spaces for obtaining feature vector to be used in CBIR.

A. Kekre's LUV Colour Space

Here we have used Kekre's LUV colour Space, which is special case of Kekre Transform. Where L gives luminance and U and V gives chromaticity values of colour image. Positive value of U indicates prominence of red component in colour image and negative value of V indicates prominence of green component. This needs the conversion of RGB to LUV components. The RGB to LUV conversion matrix given in equation 13 gives the L, U, V components of colour image for respective R, G, B components.

$$\begin{vmatrix} L \\ U \\ V \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ -2 & 1 & 1 \\ 0 & -1 & 1 \end{vmatrix} . \begin{vmatrix} R \\ G \\ B \end{vmatrix} \quad (13)$$

The LUV to RGB conversion matrix given in equation 14 gives the R, G, B components of colour image for respective L, U, V components.

$$\begin{vmatrix} R \\ G \\ B \end{vmatrix} = \begin{vmatrix} 1 & -2 & 0 \\ 1 & 1 & -1 \\ 1 & 1 & 1 \end{vmatrix} . \begin{vmatrix} L/3 \\ U/6 \\ V/2 \end{vmatrix} \quad (14)$$

B. YCbCr Colour Space

We have used YCbCr colour Space. Where Y gives luminance and Cb and Cr gives chromaticity values of colour image. To get YCbCr components we need the conversion of RGB to YCbCr components. The RGB to YCbCr conversion matrix given in equation 15 gives the

Y, Cb, Cr components of colour image for respective R, G, and B components.

$$\begin{vmatrix} Y \\ Cb \\ Cr \end{vmatrix} = \begin{vmatrix} 0.2989 & 0.5866 & 0.1145 \\ -0.1688 & -0.3312 & 0.5000 \\ 0.5000 & -0.4184 & -0.0816 \end{vmatrix} . \begin{vmatrix} R \\ G \\ B \end{vmatrix} \quad (15)$$

The YCbCr to RGB conversion matrix given in equation 16 gives the R, G, B components of colour image for respective Y, Cb, and Cr components.

$$\begin{vmatrix} R \\ G \\ B \end{vmatrix} = \begin{vmatrix} 1 & -0.0010 & 1.4020 \\ 1 & -0.3441 & -0.7140 \\ 1 & 1.7718 & 0.0010 \end{vmatrix} . \begin{vmatrix} Y \\ Cb \\ Cr \end{vmatrix} \quad (16)$$

C. YUV Colour Space

Here we have used YUV colour Space. The YUV model defines a colour space in terms of one luminance (brightness) and two chrominance (colour) components. The YUV colour model is used in the PAL, NTSC, and SECAM composite colour video standards. Previous black-and-white systems used only luminance (Y) information and colour information (U and V) was added so that a black-and-white receiver would still be able to display a colour picture as a normal black and white pictures. YUV models human perception of colour in a different way than the standard RGB model used in computer graphics hardware. The human eye has fairly little colour sensitivity: the accuracy of the brightness information of the luminance channel has far more impact on the image discerned than that of the other two. The RGB to YUV conversion matrix given in equation 17 gives the Y, U, V components of colour image for respective R, G, B components.

$$\begin{vmatrix} Y \\ U \\ V \end{vmatrix} = \begin{vmatrix} 0.299 & 0.587 & 0.144 \\ -0.14713 & -0.22472 & 0.436 \\ 0.615 & -0.51498 & 0.10001 \end{vmatrix} . \begin{vmatrix} R \\ G \\ B \end{vmatrix} \quad (17)$$

The YUV to RGB conversion matrix given in equation 18 gives the R, G, B components of colour image for respective Y, U, V components.

$$\begin{vmatrix} R \\ G \\ B \end{vmatrix} = \begin{vmatrix} 0.7492 & -0.50901 & 1.1398 \\ 1.0836 & -0.22472 & -0.5876 \\ 0.97086 & 1.9729 & -0.000015 \end{vmatrix} . \begin{vmatrix} Y \\ U \\ V \end{vmatrix} \quad (18)$$

D. YIQ Colour Space

The YIQ colour space is derived from YUV colour space and is optionally used by the NTSC composite colour video standard. The 'I' stands for in phase and 'Q' for quadrature, which is the modulation method used to transmit the colour information.

$$\begin{vmatrix} Y \\ I \\ Q \end{vmatrix} = \begin{vmatrix} 0.299 & 0.587 & 0.144 \\ 0.595716 & -0.274453 & -0.321263 \\ 0.211456 & -0.522591 & 0.31135 \end{vmatrix} . \begin{vmatrix} R \\ G \\ B \end{vmatrix} \quad (19)$$

The interconversion equations for YIQ to RGB colour space are given as per the equations (19) and (20).

$$\begin{vmatrix} R \\ G \\ B \end{vmatrix} = \begin{vmatrix} 1 & 0.9563 & 0.6210 \\ 1 & -0.2721 & -0.6474 \\ 1 & -1.107 & 1.7046 \end{vmatrix} . \begin{vmatrix} Y \\ I \\ Q \end{vmatrix} \quad (20)$$

E. YCgCb Colour Space

The new colour space YCbCg is introduced here. Interconversion equations for RGB to YCgCb colour space can be given as below in equations 21 and 22.

$$\begin{vmatrix} Y \\ Cg \\ Cb \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{vmatrix} . \begin{vmatrix} R \\ G \\ B \end{vmatrix} \quad (21)$$

$$\begin{vmatrix} R \\ G \\ B \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{vmatrix} . \begin{vmatrix} Y/3 \\ Cg/2 \\ Cb/2 \end{vmatrix} \quad (22)$$

F. XYZ Colour Space

Conversion equations for RGB to XYZ colour space and XYZ to RGB can be given as given in equations 23 and 24 below.

$$\begin{vmatrix} X \\ Y \\ Z \end{vmatrix} = \begin{vmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.71160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{vmatrix} . \begin{vmatrix} R \\ G \\ B \end{vmatrix} \quad (23)$$

$$\begin{vmatrix} R \\ G \\ B \end{vmatrix} = \begin{vmatrix} 3.240479 & -1.537150 & -0.498535 \\ -0.969256 & 1.875992 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{vmatrix} . \begin{vmatrix} X \\ Y \\ Z \end{vmatrix} \quad (24)$$

G. Normalized rgb Colour Space

In order to eliminate the influence of illumination intensity, colour information (R, G and B) can be normalized to get rgb colour space [29,30] where,

$$r = \frac{R}{R+G+B} \quad (25)$$

$$g = \frac{G}{R+G+B} \quad (26)$$

$$b = \frac{B}{R+G+B} \quad (27)$$

H. HSV Colour Space

The HSV stands for the Hue, Saturation, and Value based on the artists (Tint, Shade, and Tone). The Value represents intensity of a colour, which is decoupled from the colour information in the represented image. The Hue and Saturation components are intimately related to the way human eye perceives colour resulting in image processing algorithms with physiological basis. Conversion formula from RGB to HSV is as follows:

$$H = \cos^{-1}\left[\frac{\frac{1}{2}[(R-G)+(R-B)]}{\sqrt{[(R-G)^2+(R-B)(G-B)]}}\right] \quad (28)$$

$$S = 1 - \frac{3}{R+G+B}[min(R,G,B)] \quad (29)$$

$$V = \frac{1}{3}(R+G+B) \quad (30)$$

Conversion from HSV space to RGB space is more complex. And, given to the nature of the hue information, we will have a different formula for each sector of the colour triangle.

<u>Red-Green Sector:</u>
for $0° < H \le 120°$

$$b = \frac{1}{3}(1-S)$$

$$r = \frac{1}{3}\left[1 + \frac{S\cos H}{\cos(60° - H)}\right] \quad (31)$$

$$g = 1 - (r+b)$$

<u>Green-Blue Sector:</u>
for $120° < H \le 240°$

$$r = \frac{1}{3}(1-S)$$

$$g = \frac{1}{3}\left[1 + \frac{S\cos H}{\cos(60° - H)}\right] \quad (32)$$

$$b = 1 - (r+b)$$

<u>Blue-Red Sector:</u>
for $240° < H \le 360°$

$$g = \frac{1}{3}(1-S)$$

$$b = \frac{1}{3}\left[1 + \frac{S\cos H}{\cos(60° - H)}\right] \quad (33)$$

$$r = 1 - (r+b)$$

I. HSI Colour Space

To convert RGB to HSI [29,30], first we convert RGB to 'normalized rgb' using equations given in section 2.2.G. Each normalized H, S and I are then obtained using following equations.

$$h = \cos^{-1}\left\{\frac{\frac{1}{2}[(r-g)+(r-b)]}{\left[(r-g)^2+(r-g)(g-b)\right]^{1/2}}\right\}, h \in [0,\Pi] \quad , for b \le \quad (34)$$

$$h = 2\Pi - \cos^{-1}\left\{\frac{\frac{1}{2}[(r-g)+(r-b)]}{\left[(r-g)^2+(r-g)(g-b)\right]^{1/2}}\right\}, h \in [\Pi, 2\Pi] \quad , for, b \rangle \quad (35)$$

$$s = 1 - 3.\min(r,g,b) \quad , s \in [0,1] \quad (36)$$

$$i = (R+G+B)/(3.255) \quad , i \in [0,1] \quad (37)$$

For convenience h,s and i values are converted in the ranges of [0,360],[0,100] and [0,255] respectively using following equation 38.

$$H = h \times 180/\Pi, \quad S = s \times 100, \quad I = i \times 255 \quad (38)$$

3. FLIPPING OF IMAGE

A flipped image or reversed image, the more formal term, is a static or moving image that is generated by a mirror-reversal of an original across a horizontal axis or vertical axis which is shown in Fig 1. There are two ways to generate even or odd images as Horizontal axis and Vertical axis. Take an image and flip it along the horizontal axis or vertical axis. In the flipping along horizontal axis the original image is flipped along the X-axis. While flipping along the X-axis what we are basically doing is the pixel values of the images are interchanged wherein the 1st element of the 1st row is interchanged with the last element of the 1st row, the 2nd element of the 1st row is interchanged with the second last element of the 1st row and so on till all the pixel values of all the rows are interchanged. In flipping along vertical axis, the original image is flipped along the Y-axis. While flipping along the Y-axis what we are basically doing is the pixel values of the images are interchanged wherein the 1st element of the 1st column is interchanged with the last element of the 1st column, the 2nd element of the 1st column is interchanged with the second last element of the 1st column, the 1st element of the 2nd column is interchanged with the last element of the 2nd column , the 2nd element of the 2nd column is interchanged with the second last element of the 2nd column. and so on till all the pixel values of all the columns are interchanged.

3.1 Even Image and Odd Image

An even image is generated by adding all the pixel values of the original image and the flipped image and dividing them by 2 which is shown in Fig 1. An odd image is generated by subtracting all the pixel values of the original image and the flipped image and dividing them by 2 which is shown in Figure 1.

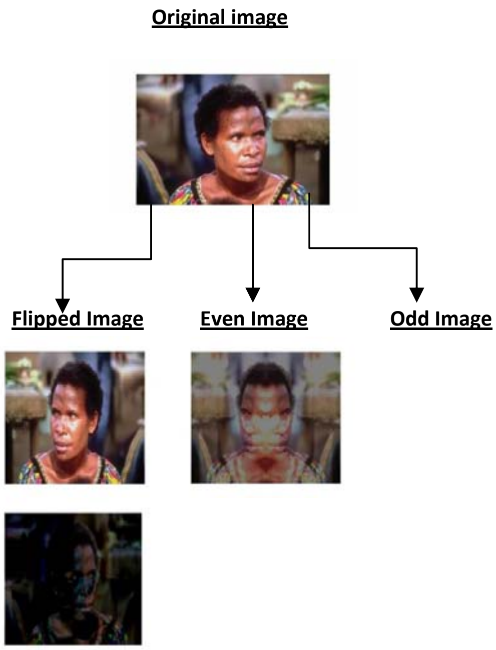

Figure 1: Original, flipped, even and odd image

4. PROPOSED CBIR TECHNIQUES

a. Simple BTC

In simple BTC we calculate the feature vector using the simple BTC technique.

b. Even BTC

In even BTC we retrieve the images using the 6 feature vectors consisting of only the even part.

c. Odd BTC

In even BTC we retrieve the images using the 6 feature vectors consisting of only the odd part.

d. Even and Simple BTC

Here we retrieve the images by using a combination of both the even part as well as the simple BTC thus increasing the feature vector to size 12.

e. Odd and Simple BTC

Here we retrieve the images by using a combination of both the odd part as well as the simple BTC thus increasing the feature vector to size 12.

f. Even and Odd BTC

Here we retrieve the images by using a combination of both the even part as well as the odd part thus increasing the feature vector to size 12.

g. Even, Odd and Simple BTC

Here we retrieve the images by using a combination of the even part as well as the odd part as well as the simple BTC thus increasing the feature vector to size 18.

5. IMPLEMENTATION

The implementation of the CBIR techniques is done in MATLAB 7.0 using a computer with Intel Core 2 Duo Processor T8100 (2.1GHz) and 2 GB RAM. The CBIR techniques are tested on the image database [15] of 1000 variable size images spread across 11 categories of human being, animals, natural scenery and manmade things. The categories and distribution of the images are Tribes (85), Buses (99), Beaches (99), Dinosaur (99), Elephants (99), Roses (99), Horses (99), Mountains (61), Airplanes (100), Monuments (99) and Sunrise (61). The table below will summarize the categories. Figure 2 gives the sample images from generic image database.

Table 1: Generic Image Database: Category-wise Distribution

Category	**Tribes**	**Buses**	**Beaches**
No of images	85	99	99
Category	Horses	Mountains	Airplanes
No of images	99	61	100
Category	Dinosaurs	Elephants	Roses
No of images	99	99	99
Category	Monuments	Sunrise	
No of images	99	61	

$$Precision = \frac{Number\_of\_relevant\_images\_retrieved}{Total\_number\_of\_images\_retrieved} \quad (39)$$

$$Recall = \frac{Number\_of\_relevant\_images\_retrieved}{Total\_number\_of\_relevent\_images\_in\_database} \quad \text{“}(40)$$

Figure 2: Sample images of Generic Image Database [Image database contains total 1000 images with 11 categories]

For testing the performance of each proposed CBIR technique first five images from each category are fired on the database as queries. The average precision and average recall are computed by grouping the number of retrieved images.

Figure 3 gives performance comparison of BTC-CBIR techniques based on image flipping for respective colour spaces. In general we could conclude that the performance of image retrieval improves when considering the proposed CBIR technique i.e. considering odd, even parts and original image and various permutations and combination of these parts applied on various non-luminance colour spaces like HSV, XYZ, HIS, Normalized RGB as well as luminance-chromaticity based colour spaces like Kekre‟s LUV, YCbCr, YUV, YIQ and Kekre‟s YCgCb.

6. RESULTS AND DISCUSSIONS

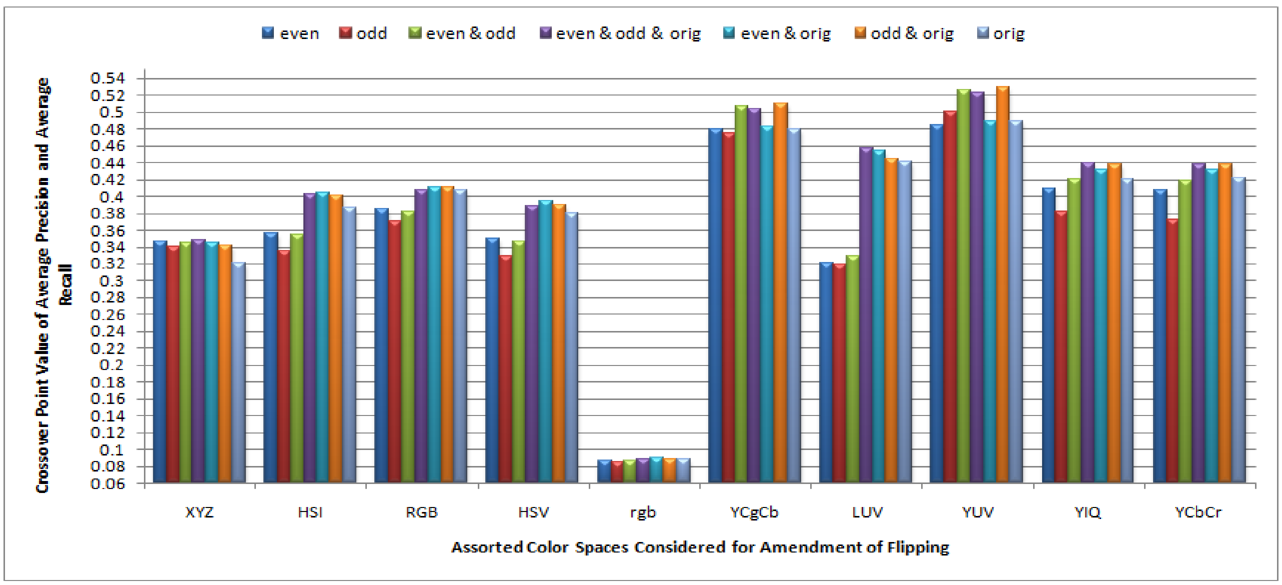

Figure 3: Comparison of performance of BTC-CBIR using image flipping techniques for respective colour spaces

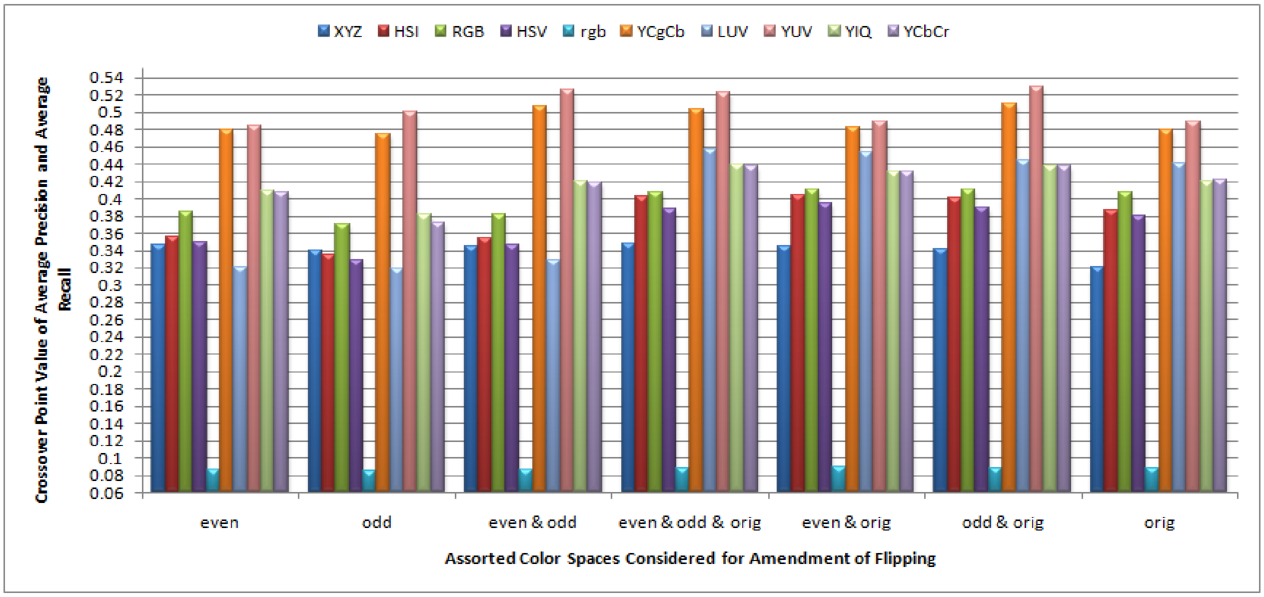

Figure 4 Comparison of performance of BTC-CBIR using assorted colour spaces for respective image flipping techniques

The performance comparison based on image flipping methods for BTC-CBIR techniques using assorted colour spaces is shown in figure 4. Here in even flipping based BTC-CBIR, it can be observed that the performance of luminance-chromaticity based colour spaces (YCbCr, YIQ, YUV, Kekre's YCgCb) is higher than other colour spaces. Similarly while considering odd flipping, combination of even and odd, even& odd & original, even& original, odd& original the performance of luminance-chromaticity based colour spaces (YCbCr, YIQ, YUV, Kekre' s YCgCb) is higher than other colour spaces. In general, the performance of YUV and Kekre's YCgCb is best in all proposed flipping techniques.

7. CONCLUSION

So far many CBIR techniques have been proposed, but till the researchers are craving for better and faster image retrieval solutions. The paper presented the exhaustive comparison of image retrieval techniques based on augmented colour-BTC using 10 assorted colour spaces and even and odd image parts. The techniques are tested on image database of 1000 generic images spread across 11 different categories. Image flipping helps to improve the performance in all luminance-chromaticity colour spaces (YUV, YIQ, LUV, Kekre's YCgCb, YCbCr) as well as non-luminance based colour spaces (XYZ,HSI,RGB,HSV) in comparison of BTC applied on original image at the cost of increased feature vector size. Overall YUV colour space proves to be the best in all proposed image flipping based CBIR techniques among the proposed extensions to BTC-CBIR.

8. REFERENCES

[1] H.B.Kekre, Sudeep D. Thepade, "Boosting Block Truncation Coding using Kekre"s LUV Colour Space for Image Retrieval", WASET International Journal of Electrical, Computer and System Engineering (IJECSE), Volume 2, Number 3, pp. 172-180, Summer 2008. Available online at http://www.waset.org/ijecse/v2/v2-3-23.pdf.

[2] H.B.Kekre, Sudeep D. Thepade, "Image Retrieval using Augmented Block Truncation Coding Techniques", ACM International Conference on Advances in Computing, Communication and Control (ICAC3-2009), pp. 384-390, 23-

[3] 24 Jan 2009, Fr. Conceicao Rodrigous College of Engg., Mumbai. Is uploaded on online ACM portal.

[4] H.B.Kekre, Sudeep D. Thepade, "Scaling Invariant Fusion of Image Pieces in Panorama Making and Novel Image Blending Technique", International Journal on Imaging (IJI), Volume 1, No. A08, pp. 31-46, Autumn 2008. www.ceser.res.in/iji.html

[5] Hirata K. and Kato T. "Query by visual example – content-based image retrieval", In Proc. of Third International Conference on Extending Database Technology, EDBT"92, 1992, pp 56-71

[6] H.B.Kekre, Sudeep D. Thepade, "Rendering Futuristic Image Retrieval System", National Conference on Enhancements in Computer, Communication and Information Technology, EC2IT-2009, 20-21 Mar 2009, K.J.Somaiya College of Engineering, Vidyavihar, Mumbai-77.

[7] Minh N. Do, Martin Vetterli, "Wavelet-Based Texture Retrieval Using Generalized Gaussian Density and Kullback-Leibler Distance", IEEE Transactions On Image Processing, Volume 11, Number 2, pp.146-158, February 2002.

[8] B.G.Prasad, K.K. Biswas, and S. K. Gupta, "Region –based image retrieval using integrated colour, shape, and location index", International Journal on Computer Vision and Image Understanding Special Issue: Colour for Image Indexing and Retrieval, Volume 94, Issues 1-3, April-June 2004, pp.193-233.

[9] H.B.Kekre, Sudeep D. Thepade, "Creating the Colour Panoramic View using Medley of Grayscale and Colour Partial Images ", WASET International Journal of Electrical, Computer and System Engineering (IJECSE), Volume 2, No. 3, Summer 2008. Available online at www.waset.org/ijecse/v2/v2-3-26.pdf.

[10] Stian Edvardsen, "Classification of Images using colour, CBIR Distance Measures and Genetic Programming", Ph.D. Thesis, Master of science in Informatics, Norwegian university of science and Technology, Department of computer and Information science, June 2006.

[11] H.B.Kekre, Tanuja Sarode, Sudeep D. Thepade, "DCT Applied to Row Mean and Column Vectors in Fingerprint Identification", In Proceedings of International Conference on Computer Networks and Security (ICCNS), 27-28 Sept. 2008, VIT, Pune.

[12] Zhibin Pan, Kotani K., Ohmi T., "Enhanced fast encoding method for vector quantization by finding an optimally-ordered Walsh transform kernel", ICIP 2005, IEEE International Conference, Volume 1, pp I - 573-6, Sept. 2005.

[13] H.B.kekre, Sudeep D. Thepade, "Improving „Colour to Gray and Back‟ using Kekre‟s LUV Colour Space", IEEE International Advanced Computing Conference 2009 (IACC‟09), Thapar University, Patiala, INDIA, 6-7 March 2009. Is uploaded and available online at IEEE Xplore.

[14] H.B.Kekre, Sudeep D. Thepade, "Image Blending in Vista Creation using Kekre's LUV Colour Space", SPIT-IEEE Colloquium and International Conference, Sardar Patel Institute of Technology, Andheri, Mumbai, 04-05 Feb 2008.

[15] H.B.Kekre, Sudeep D. Thepade, "Colour Traits Transfer to Grayscale Images", In Proc.of IEEE First International Conference on Emerging Trends in Engg. & Technology, (ICETET-08), G.H.Raisoni COE, Nagpur, INDIA. Uploaded on online IEEE Xplore.

[16] http://wang.ist.psu.edu/docs/related/Image.orig (Last referred on 23 Sept 2008)

[17] H.B.Kekre, Sudeep D. Thepade, "Using YUV Colour Space to Hoist the Performance of Block Truncation Coding for Image Retrieval", IEEE International Advanced Computing Conference 2009 (IACC‟09), Thapar University, Patiala, INDIA, 6-7 March 2009.

[18] H.B.Kekre, Sudeep D. Thepade, Archana Athawale, Anant Shah, Prathmesh Verlekar, Suraj Shirke,"Energy Compaction and Image Splitting for Image Retrieval using Kekre Transform over Row and Column Feature Vectors", International Journal of Computer Science and Network Security (IJCSNS),Volume:10, Number 1, January 2010, Available at www.IJCSNS.org.

[19] H.B.Kekre, Sudeep D. Thepade, Archana Athawale, Anant Shah, Prathmesh Verlekar, Suraj Shirke, "Walsh Transform over Row Mean and Column Mean using Image Fragmentation and Energy Compaction for Image Retrieval", International Journal on Computer Science and Engineering (IJCSE),Volume 2S, Issue1, January 2010, Available online at www.enggjournals.com/ijcse.

[20] H.B.Kekre, Sudeep D. Thepade, "Image Retrieval using Colour-Texture Features Extracted from Walshlet Pyramid", ICGST International Journal on Graphics, Vision and Image Processing (GVIP), Volume 10, Issue I, Feb.2010, pp.9-18, Available online www.icgst.com/gvip/Volume10/Issue1/P1150938876.html

[21] H.B.Kekre, Sudeep D. Thepade, "Colour Based Image Retrieval using Amendment Block Truncation Coding with YCbCr Colour Space", International Journal on Imaging (IJI), Volume 2, Number A09, Autumn 2009, pp. 2-14. Available online at www.ceser.res.in/iji.html (ISSN: 0974-0627).

[22] H.B.Kekre, Tanuja Sarode, Sudeep D. Thepade, "Colour-Texture Feature based Image Retrieval using DCT applied on Kekre‟s Median Codebook", International Journal on Imaging (IJI), Volume 2, Number A09, Autumn 2009,pp. 55-65. Available online at www.ceser.res.in/iji.html

[23] H.B.Kekre, Sudeep D. Thepade, Akshay Maloo "Performance Comparison for Face Recognition using PCA, DCT &WalshTransform of Row Mean and Column Mean", ICGST International Journal on Graphics, Vision and Image Processing (GVIP), Volume 10, Issue II, Jun.2010, pp.9-18, Available at http://209.61.248.177/gvip/Volume10/Issue2/P1181012028.pdf..

[24] H.B.Kekre, Sudeep D. Thepade, "Improving the Performance of Image Retrieval using Partial Coefficients of Transformed Image", International Journal of Information Retrieval, Serials Publications, Volume 2, Issue 1, 2009, pp. 72-79

[25] H.B.Kekre, Sudeep D. Thepade, Archana Athawale, Anant Shah, Prathmesh Verlekar, Suraj Shirke, "Performance Evaluation of Image Retrieval using Energy Compaction and Image Tiling over DCT Row Mean and DCT Column Mean", Springer-International Conference on Contours of Computing Technology (Thinkquest-2010), Babasaheb Gawde Institute of Technology, Mumbai, 13-14 March 2010, The paper will be uploaded on online Springerlink.

[26] H.B.Kekre, Tanuja K. Sarode, Sudeep D. Thepade, Vaishali Suryavanshi,"Improved Texture Feature Based Image Retrieval using Kekre‟s Fast Codebook Generation Algorithm", Springer-International Conference on Contours of Computing Technology (Thinkquest-2010), Babasaheb Gawde Institute of Technology, Mumbai, 13-14 March 2010.

[27] H.B.Kekre, Tanuja K. Sarode, Sudeep D. Thepade, "Image Retrieval by Kekre‟s Transform Applied on Each Row of Walsh Transformed VQ Codebook", (Invited), ACM-International Conference and Workshop on Emerging Trends in Technology (ICWET 2010),Thakur College of Engg. And Tech., Mumbai, 26-27 Feb 2010, The paper is invited at ICWET 2010.

[28] H.B.Kekre, Sudeep D. Thepade, Adib Parkar, "A Comparison of Kekre's Fast Search and Exhaustive Search for various Grid Sizes used for Colouring a Greyscale Image", 2nd International Conference on Signal Acquisition and Processing (ICSAP 2010), IACSIT, Bangalore, pp. 53-57, 9-10 Feb 2010.

9. AUTHOR BIOGRAPHIES

	**Dr. H. B. Kekre** has received B.E. (Hons.) in Telecomm. Engineering. from Jabalpur University in 1958, M.Tech (Industrial Electronics) from IIT Bombay in 1960, M.S.Engg. (Electrical Engg.) from University of Ottawa in 1965 and Ph.D. (System Identification) from IIT Bombay in 1970 He has worked as Faculty of Electrical Engg. and then HOD Computer Science and Engg. at IIT Bombay. For 13 years he was working as a professor and head in the Department of Computer Engg. at Thadomal Shahani Engineering. College, Mumbai. Now he is Senior Professor at MPSTME, SVKM's NMIMS University. He has guided 17 Ph.Ds, more than 100 M.E./M.Tech and several B.E./B.Tech projects. His areas of interest are Digital Signal processing, Image Processing and Computer Networking. He has more than 320 papers in National / International Conferences and Journals to his credit. He was Senior Member of IEEE. Presently He is Fellow of IETE and Life Member of ISTE Recently ten students working under his guidance have received best paper awards and two have been conferred Ph.D. degree of SVKM's NMIMS University. Currently 10 research scholars are pursuing Ph.D. program under his guidance
	Sudeep D. Thepade has Received B.E.(Computer) degree from North Maharashtra University with Distinction in 2003. M.E. in Computer Engineering from University of Mumbai in 2008 with Distinction, currently pursuing Ph.D. from SVKM's NMIMS University, Mumbai. He has about than 07 years of experience in teaching and industry. Currently working as Associate Professor in Computer Engineering at

	Mukesh Patel School of Technology Management and Engineering, SVKM's NMIMS University, Vile Parle(w), Mumbai, INDIA. He is member of IAENG and IACSIT, Singapore. He is reviewer for many international journals and in the international advisory panel for many international conferences. He has worked as member of International Advisory Committee for many International Conferences. His areas of interest are Image Processing and Computer Networks. He has about 80 papers in National/International Conferences/Journals to his credit with a Best Paper Award at Int. Conference SSPCCIN-2008, Second Best Paper Award at ThinkQuest-2009 National Level faculty paper presentation competition and Best Paper Award at Springer Int. Conf. ICCCT-2010 **Chitrangada Nemani, Ajoy Joseph, Pooja Luthra, Varun Lodha** are currently pursuing B.Tech. (IT) from MPSTME, SVKM's NMIMS University, Mumbai. There area of interest is Content Based Image Retrieval in IP