Improving Performance of Multileveled BTC Based CBIR Using Sundry Color Spaces

Dr. H. B. Kekre hbkekre@yahoo.com
Senior Professor,
MPSTME, SVKM's NMIMS University,
Mumbai-56, India

Sudeep Thepade sudeepthepade@gmail.com
Associate Professor and *Ph.D. Research Scholar,*
MPSTME,SVKM's NMIMS University,
Mumbai-56, India

Shrikant P. Sanas shrikant_sanas@yahoo.co.in
M-tech Student
MPSTME,SVKM's NMIMS University,
Mumbai-56, India

Abstract

The paper presents an extension of content based image retrieval (CBIR) techniques based on multilevel Block Truncation Coding (BTC) using nine sundry color spaces. Block truncation coding based features is one of the CBIR methods proposed using color features of image. The approach basically considers red, green and blue planes of an image to compute feature vector. This BTC based CBIR can be extended as multileveled BTC for performance improvement in image retrieval. The paper extends the multileveled BTC using RGB color space to other nine color spaces. The CBIR techniques like BTC Level-1, BTC Level-2, BTC Level-3 and BTC Level-4 are applied using various color spaces to analyze and compare their performances. The CBIR techniques are tested on generic image database of 1000 images spread across 11 categories. For each CBIR technique, 55 queries (5 per category) are fired on extended Wang generic image database to compute average precision and recall for all queries. The results have shown the performance improvement (ie., higher precision and recall values) with BTC-CBIR methods using luminance-chrominance color spaces (YCgCb, Kekre's LUV, YUV, YIQ, YCbCr) as compared to non-luminance (RGB, HSI, HSV, rgb , XYZ) Color spaces. The performance of multileveled BTC-CBIR increases gradually with increase in level up to certain extent (Level 3) and then increases slightly due to voids being created at higher levels. In all levels of BTC Kekre's LUV color space gives best performance.

Keywords: Content Based Image Retrieval (CBIR), BTC, Color Spaces.

1. INTRODUCTION

From ancient era, images play an important role in human communication. It is basic and common way to express the information. Today with advancement in information and communication technology most of the information is digitized. Large amount of digital data is generated, transmitted, stored, analyzed and accessed. Mostly, information is in the form of multimedia such as digital images, audio, video, graphics [6]-8. Large numbers of images are generated from various sources on a daily basis. Such images occupy lot of space and are very challenging to search and retrieve from very large image pool. The need for efficient retrieval of images has been recognized by managers and users of large image collections. Efficient indexing techniques for the retrieval of best matching image from a huge database of images are being developed. Content based image retrieval gives efficient solution to these problems [18,19]. Content Based Image Retrieval (CBIR) is used to provide a high percentage of relevant images in response to the query image [12]. The goal of an image retrieval system is to retrieve a set of matching images from an image database [21].

A Content Based Image Retrieval (CBIR) technique takes an image as an input to query and outputs number of matching images to the query image [11]. In CBIR technique, features are used to represent the image content. The features are extracted automatically and there is no manual intervention, and thus eliminating the dependency on humans in the feature extraction stage [10]. The typical CBIR system performs two major tasks. The first one is feature extraction (FE), where a set of features, forming feature vector, is generated to accurately represent the content of each image in the database. A feature vector is much smaller in size than the original image. The second task is similarity measurement (SM), where a distance between the query image and each image in the database using their feature vectors (signatures) is computed so that the top “closest” images retrieved [3], [13], [14], [15].

Many current CBIR system use Euclidean distance [5] on the extracted feature set as a similarity measure. The Direct Euclidian distance between image P and query image Q can be given as equation. where Vpi and Vqi are the feature vectors of image P and query image Q respectively with size ‘n’.

$$ED = \sqrt{\sum_{i=1}^{n} \left(Vpi - Vqi\right)^{2}} \quad (1)$$

Some of important applications for CBIR technology could be identified as art galleries, museums, archaeology, architecture design, geographic information systems, weather forecast, medical imaging , trademark databases, criminal investigations, image search on the Internet. The thirst of a better and faster image retrieval technique is increasing day by day. The paper discusses the performance improvement in multileveled BTC based CBIR techniques [1] using various color spaces. In all ten different color spaces including RGB color space are considered here with four different levels of BTC for feature extraction resulting into total 40 CBIR methods.

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 [2]. Block Truncation Coding (BTC) was first developed in 1979 for grayscale image coding [2]. 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.

3. CBIR USING MULTILEVELED BLOCK TRUNCATION CODING

The block truncation coding (BTC) technique can be extended to higher levels by considering multiple threshold values to divide the image pixels into higher (upper) and less than or equal to (lower) threshold. The image pixel data is thus divided in to multiple clusters and per cluster the mean value is taken as part of feature vector. At BTC Level 1 only one threshold value is used to divide the pixel data to get two clusters and respective means of these clusters as upper mean and lower mean are computed, resulting in to feature vector of size six (two values per color plane). In next level each cluster can be further divided into two parts with respect to it's mean value resulting into total four clusters per color plane to get feature vector of size twelve (four per plane). Thus BTC can be extended to multiple levels to get BTC Level 2, BTC Level 3, etc. The feature vector extraction for CBIR using multileveled BTC with RGB color space is explained in section A, B and C here.

A) CBIR using BTC-RGB-Level-1 (BTC-RGB-6) [3,4,9,17]

In original BTC we divide the image into R, B, and G components and compute the mean value of each color component as individual color. Let I(m,n)=[R(m,n), G(m,n), B(m,n)] be the color image image of size mxn. Let the thresholds be MR, MG and MB, which could be computed as per the equations 2, 3 and 4.

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

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

$$MB = \frac{1}{m*n}\sum_{i=1}^{m}\sum_{j=1}^{n}B(i,j) \tag{4}$$

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'.

$$BMr(i,j) = \{ \begin{matrix} 1, if \ldots R(i,j) > MR \\ 0, \ldots if \ldots R(i,j) <= MR \end{matrix} \tag{5}$$

$$BMg(i,j) = \{ \begin{matrix} 1, if \ldots G(i,j) > MG \\ 0, \ldots if \ldots G(i,j) <= MG \end{matrix} \tag{6}$$

$$BMb(i,j) = \{ \begin{matrix} 1, if \ldots B(i,j) > MB \\ 0, \ldots if \ldots B(i,j) <= MB \end{matrix} \tag{7}$$

$$UR = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n}BMr(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n}BMr(i,j)*R(i,j) \tag{8}$$

$$UG = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMg(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} BMg(i,j) * G(i,j) \quad (9)$$

$$UB = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMb(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} BMb(i,j) * B(i,j) \quad (10)$$

Two mean colors 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 color UM(UR, UG, UB) is given as equations 8, 9 and 10.

And the Lower Mean LM= (LR, LG, LB) is computed as following equations 11, 12 and 13.

$$LR = \frac{1}{m*n - \sum_{i=1}^{m}\sum_{j=1}^{n} BMr(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMr(i,j)\} * R(i,j) \quad (11)$$

$$LG = \frac{1}{m*n - \sum_{i=1}^{m}\sum_{j=1}^{n} BMg(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMg(i,j)\} * G(i,j) \quad (12)$$

$$LB = \frac{1}{m*n - \sum_{i=1}^{m}\sum_{j=1}^{n} BMb(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMb(i,j)\} * B(i,j) \quad (13)$$

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.

B) CBIR using BTC- RGB- Level 2 (BTC-RGB-12)[1]

In BTC-RGB-Level 2 the image data is divided into 12 parts using the six means obtained in BTC-RGB-Level 1. Here the bitmap are prepared using upper and lower mean values of individual color components. For red color component the bitmap 'BMUR' and 'BMLR' are generated as given in equations 14 and 15. Similarly for green color component 'BMUG' & 'BMLR' and for blue colour components 'BMUB' & 'BMLB' can be generated.

$$BMUR(i,j) = \{ \begin{matrix} 1, if....R(i,j) > UR \\ 0,...if...R(i,j) <= UR \end{matrix} \quad (14)$$

$$BMLR(i,j) = \{ \begin{matrix} 1, if....R(i,j) > LR \\ 0,...if...R(i,j) <= LR \end{matrix} \quad (15)$$

Using this bitmap the two mean colour per bitmap one for the pixels greater than or equal to the threshold and other for the pixels smaller than the threshold are calculated [15]. The red plane of the image is divided into two parts as upper red image and lower red image as given by equations 16 and 17.

$$Iur(i,j) = R(i,j) * BMr(i,j),..for.i = 1...m, and.j = 1...n. \quad (16)$$

$$Ilr(i,j) = R(i,j) * [1 - BMr(i,j)],..for.i = 1...m, and.j = 1...n. \quad (17)$$

The upper mean color UM (UUR, ULR, UUG, ULG, UUB, ULB) are given as follows.

$$UUR = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMUR(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} BMUR(i,j) * Iur(i,j) \tag{18}$$

$$ULR = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMLR(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} BMLR(i,j) * Ilr(i,j) \tag{19}$$

And the first two components of Lower Mean LM= (LUR, LLR, LUG, LLG, LUB, LLB) are computed using following equations

$$LUR = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMUR(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMUR(i,j)\} * Iur(i,j) \tag{20}$$

$$LLR = \frac{1}{\sum_{i=1}^{m}\sum_{j=1}^{n} BMLR(i,j)} * \sum_{i=1}^{m}\sum_{j=1}^{n} \{1 - BMLR(i,j)\} * Ilr(i,j) \tag{21}$$

These Upper Mean and Lower Mean values together will form a feature vector for BTC-12. For every image stored in the database these feature vectors are computed and stored in feature vector table.

C) CBIR using BTC-RGB-Level 3 (BTC-RGB-24)[1] and BTC-RGB-Level 4 (BTC-RGB-48)

Similarly the feature vector for BTC-24 can be found by extending the BTC till level 3. Each plane will give the 8 elements of feature vector. For Red plane we get (UUUR, LUUR, ULUR, LLUR, UULR, LULR, ULLR, LLLR). Also feature vector of size 48 can be generated by extending this process of BTC to next level.

4. COLOR SPACES IN CBIR USING MULTILEVELED BTC

Just as discussed in section 3 for RGB color space, the CBIR using multileveled BTC can be used with other color spaces. Here in all ten color spaces like RGB, HSV[20], XYZ[20], rgb[20], HSI[20], Kekre's LUV [3], YCbCr[17], YUV[9], YIQ[20], Kekre's YCgCb[20] are considered. The ten color spaces along with BTC extended to four levels resul into total 40 CBIR methods. These color spaces can mainly be divided into two categories as luminance-chrominance color spaces (Kekre's LUV, YCbCr, YUV, YIQ, Kekre's YCgCb) and nom-luminance color spaces (RGB, HSI, HSV, XYZ and rgb).

5. IMPLEMENTATION

The implementation of these CBIR techniques is done using MATLAB 7.0. The CBIR techniques are tested on the augmented Wang [15] image database of 1000 variable size images spread across 11 categories of human beings, animals, natural scenery and man-made things. To compare various techniques in various color space, performance is evaluated based on precision and recall. The efficiency of CBIR technique is evaluated based on accuracy, stability and speed. To assess the retrieval effectiveness, we have used the precision and recall as statistical comparison parameters for the BTC-6, BTC-12, BTC-24 and BTC-48 techniques of CBIR on all color spaces. The standard definitions of these two measures are given by following equations.

$$\text{Precesion} = \frac{\text{Number\_of\_relevant\_images\_retrieved}}{\text{Total\_number\_of\_images\_retrieved}} \tag{22}$$

$$\text{Recall} = \frac{\text{Number\_of\_relevant\_images\_retrieved}}{\text{Total\_number\_of\_relevent\_images\_in\_database}} \quad (23)$$

6. RESULTS AND DISCUSSION

The methods BTC-6, BTC-12, BTC-24 and BTC-48 are applied to the image database of 1000 images using various color spaces. The resulting 40 CBIR methods are tested using 55 random queries (5 per image category). The average precision and recall values of all these queries per CBIR methods are computed and considered for performance comparison.

Figure 1 shows the average precision and recall values plotted against number of retrieved images for RGB-BTC-CBIR at level 1 (RGB-BTC-6), level 2 (RGB-BTC-12), level 3 (RGB-BTC-24) and level 4 (RGB-BTC-48). The conclusion that 'higher the level of BTC used better the performance of CBIR is' can be drawn from the given graph as indicated by higher precision and recall values in BTC 48 and BTC 24 as compared to BTC-12 and BTC-6.

However the distinction in the performance of all these techniques is not very clear. The height of crossover point of precision and recall curves plays very important role in performance comparison of CBIR methods. Ideally this crossover point height should be one. Higher the value of this crossover point better the performance is.

Figure 2a shows the zoomed version of graphs in figure 1 for crossover points of precision and recall curves. Figure 2 b shows the bar graphs indicating the heights of crossover points of precision-recall curves of respective CBIR methods shown in figure 2a. From this figure 2b it can be observed that the best performance (Highest precision-recall crossover point value) is given by BTC-RGB-48. The increase in performance from BTC-RGB-6 to BTC-RGB-24 is gradual in nature while the performances of BTC-RGB-24 and BTC-RGB48 have very slight difference. So the conclusion that multileveled BTC helps to improve the performance of CBIR methods up to BTC-Level 3 (BTC-24) can be drawn from these observations.

Figure 3 gives the comparison of crossover points of the multileveled BTC-CBIR techniques per color space. In all color spaces increasing the BTC level helps in improving the performance of CBIR as indicated by higher crossover point values.

Figure 4 gives the comparison of crossover points for all color spaces per level of BTC-CBIR techniques. In all levels of BTC-CBIR luminance-chrominance color spaces (YCgCb, Kekre's LUV, YUV, YIQ, YCbCr) better than non-luminance (RGB, HSI, HSV, rgb, XYZ) Color spaces and in all Kekre's LUV gives best performance.

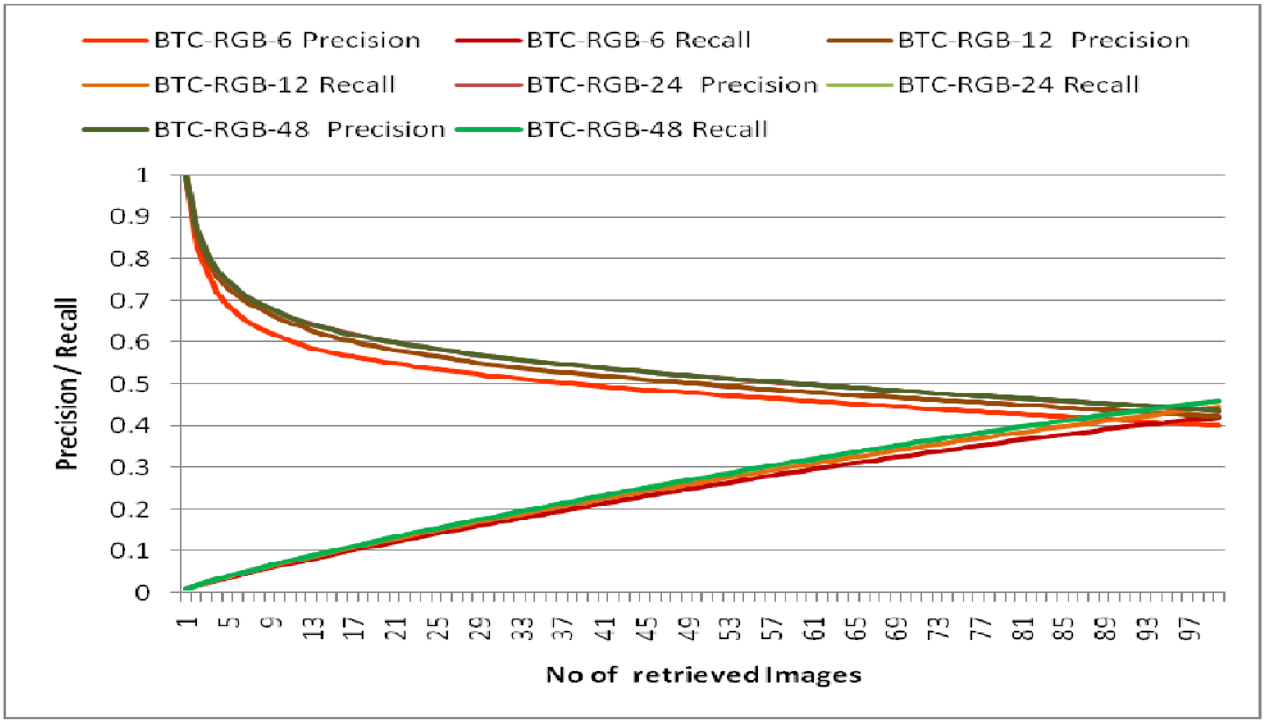

FIGURE 1 : Precision & Recall plotted against for RGB color space.

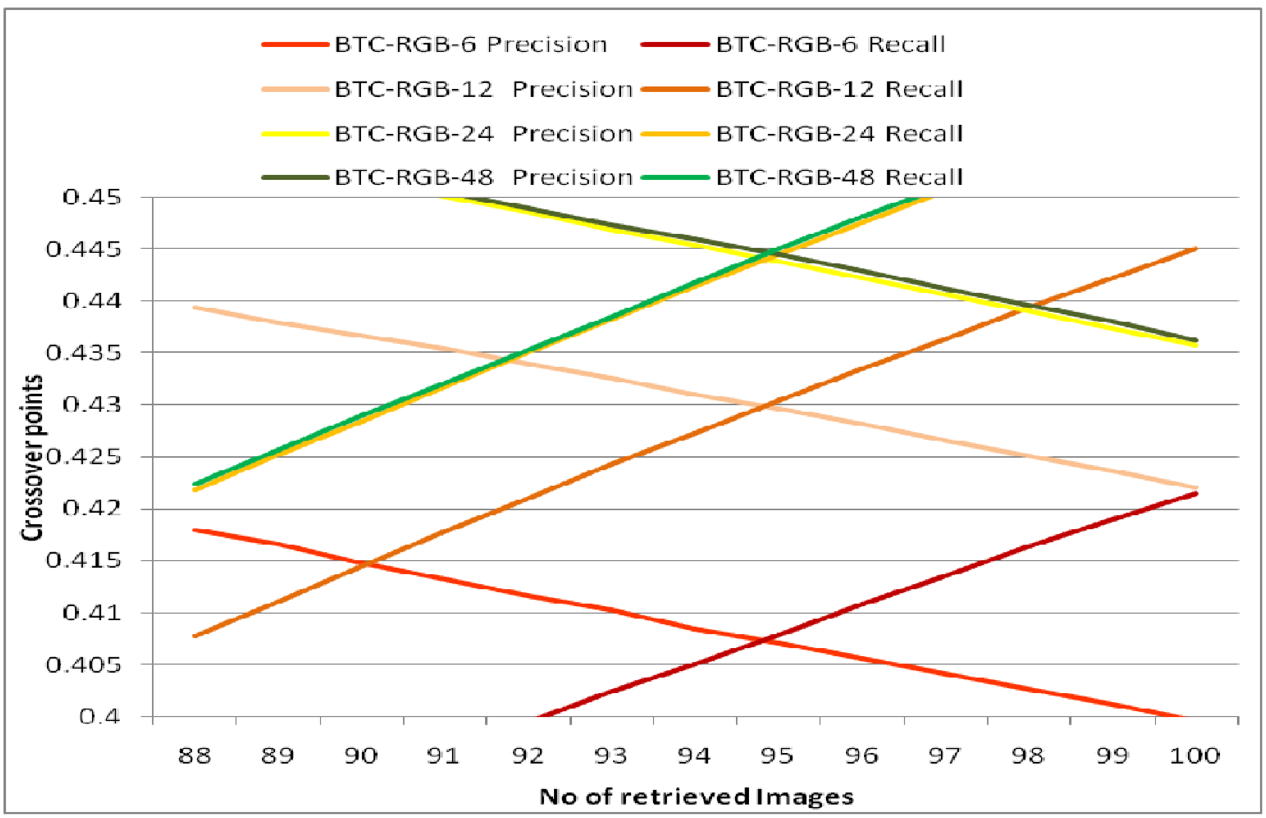

FIGURE2.A: Crossover points of Precision-Recall plotted against number of retrieved images.

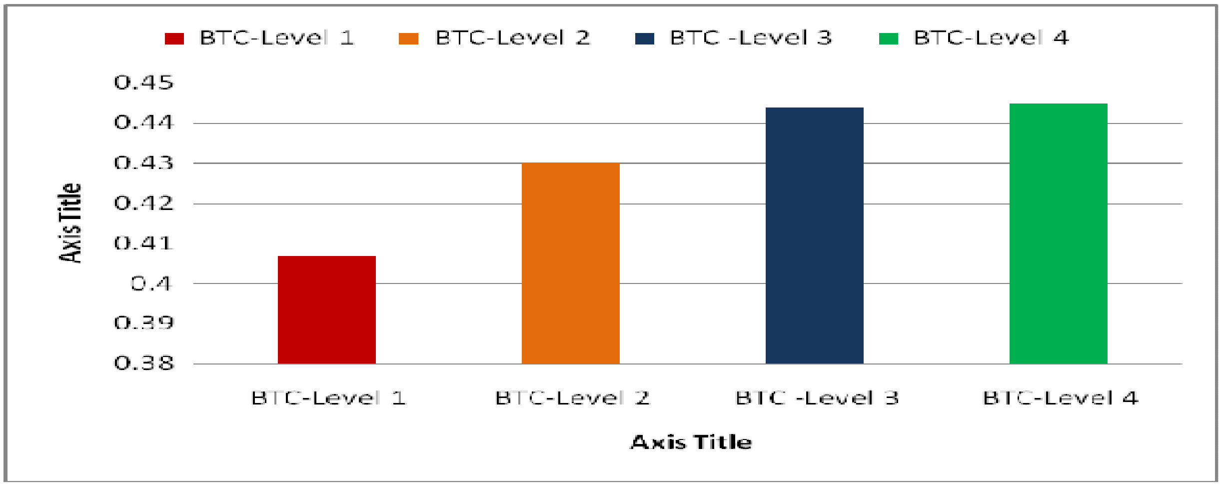

FIGURE 2.B : Performance comparison of discussed CBIR methods using Precision-Recall crossover points for RGB color sapce.

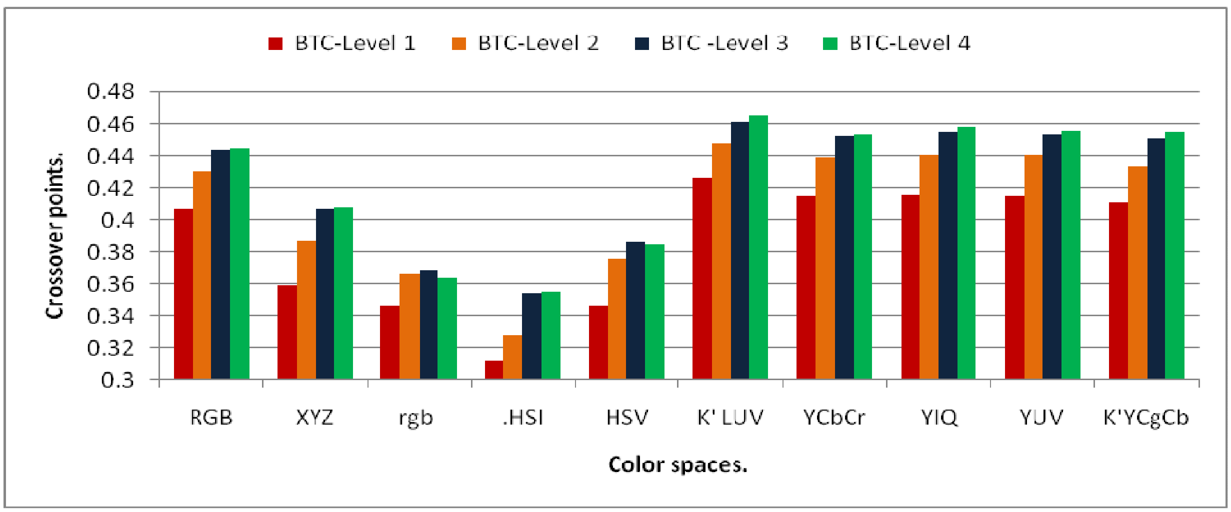

FIGURE 3: Performance Comparison between different levels of BTC for respective color spaces.

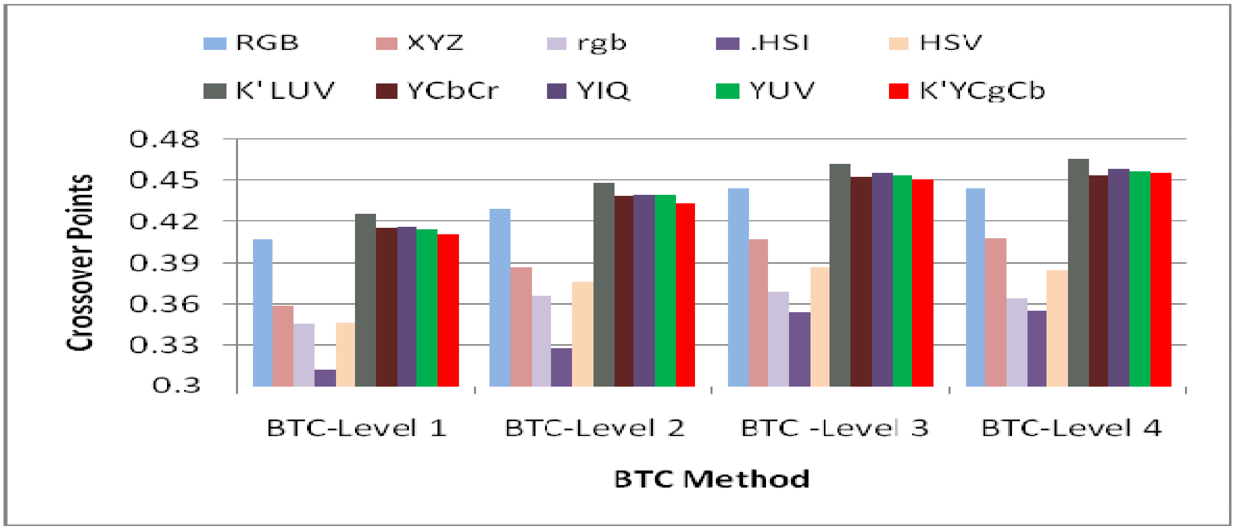

FIGURE 4: Performance Comparison between different color spaces for respective level of BTC.

7. CONCLUSION

The performance of CBIR system depends on the precision and recall. Quite often the crossover point of precision and recall is taken as criteria for judging the performance of CBIR technique on various Color spaces. Kekre's LUV gives better result and the values are 0.4258, 0.448, 0.462, 0.46495 for BTC-6, BTC-12, BTC-24 and BTC-48 respectively. The results have shown that, the performance improvement (i.e. higher precision and recall values) in Luminous & Chrominance Color spaces with BTC-CBIR methods compared to Non-Luminous &Non-Chrominance Color spaces are lower. Performance improves with increased level of BTC. Up to level -3 gradual increases in performance with increasing level is observed while the difference of performance in level-3 and level-4 is negligible due to voids being created at higher levels.

8. REFERENCES

[1] Dr.H.B.Kekre, Sudeep D. Thepade, Shrikant P. Sanas 'Improved CBIR using Multileveled Block Truncation Coding' International Journal of Computer Science and Engineering (IJCSE), Volume 2, Number 7, October 2010 Edition, pp2471-2476. http://www.enggjournals.com/ijcse.

[2] Guoping Qiu, "Color Image Indexing Using BTC," IEEE Transactions on Image Processing, VOL.12, NO.1, pp.93-101, January 2003.

[3] Dr.H.B.Kekre, Sudeep D. Thepade, "Boosting Block Truncation Coding using Kekre's LUV Color 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

[4] Dr.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, 23-24 Jan 2009, Fr. Conceicao Rodrigous College of Engg., Mumbai. Is uploaded on online ACM portal.

[5] Dr.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), www.ceser.res.in/iji.html, Volume 1, No. A08, pp. 31-46, Autumn 2008.

[6] B.G.Prasad, K.K. Biswas, and S. K.Gupta, “Region–based image retrieval using integrated color, shape, and location index”, computer vision and image understanding, Oct 2003.

[7] Minh N. Do, Member, IEEE, and Martin Vetterli, Fellow, IEEE,” Wavelet-Based Texture Retrieval Using Generalized Gaussian Density and Kullback-Leibler Distance,” IEEE Transactions nn Image Processing, Vol.11, No.2, Feb 2002.

[8] Dr.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.

[9] Dr.H.B.Kekre, Sudeep D. Thepade, “Using YUV Color 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.

[10] Dr.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, (ISSN: 1738-7906) Available at www.IJCSNS.org.

[11] Dr.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, (**ISSN: 0975–3397). Available online at www.enggjournals.com/ijcse.**

[12] Dr.H.B.Kekre, Sudeep D. Thepade, “Image Retrieval using Color-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

[13] Khalid Sayood ,” Introduction to Data Compression ,” University of Nebraska-Lincoln , Second Edition , ISBN:1-55860-558-4, by Academic Press,2000.

[14] Stian Edvardsen,”Classification of Images using color, 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.

[15] Rafael C.Gonzalez, Richard E. Woods,” Digital Image Processing,” University of Tennessee, Second Edition, ISBN 81-7808-629-8, Pearson Education Pvt. Ltd.,2002.

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

[17] Dr.H.B.Kekre, Sudeep D. Thepade, “Color Based Image Retrieval using Amendment Block Truncation Coding with YCbCr Color 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).**

[18] Dr.H.B.Kekre, Tanuja Sarode, Sudeep D. Thepade, “Color-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 (ISSN: 0974-0627).**

[19] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: the QBIC system. *IEEE Computer*, 28(9):23–32, 1995.

[20] Dr.H.B.Kekre, Sudeep D. Thepade, A. Athawale, Adib Parkar, "Using Assorted Color Spaces and Pixel Window Sizes for Colorization of Grayscale Images", ACM-International Conference and Workshop on Emerging Trends in Technology (ICWET 2010), Thakur College of Engg. And Tech., Mumbai, 26-27 Feb 2010, uploaded on online ACM Portal.

[21] Dr.H.B.Kekre, Sudeep D. Thepade, Shobhit W., Miti K., Styajit S., Priyadarshini M. "Image Retrieval with Shape Features Extracted using Gradient Operators and Slope Magnitude Technique with BTC", International Journal of Computer Applications (IJCA), Volume 6, Number 8, pp.28-33, September 2010. **Available online at** http://www.ijcaonline.org/volume6/number8/pxc3871430.pdf