Scientific Journal Impact Factor (SJIF): 1.711
International Journal of Modern Trends in Engineering and Research
IJMTER
ISSN (ONLINE) : 2349 - 9745
ISSN (PRINT) : 2393 - 8161

Truncated DCT Image BTC for Image Retrieval

Ravindra C. Patil [1], NST Sai [2], Shailesh Sangle[3]
[1,3]*Assistant professor, Computer Engineering Dept.*
Thakur College of Engineering and Technology, Mumbai, Maharashtra, India-400101
[2] *Tech Mahindra Limited, Mumbai, Maharashtra, India*

Abstract: Histogram based color features vector size is large which require more memory to save it. To make a feature vector compact and effective in terms of retrieval performance Block Truncation Coding (BTC) for the color images are proposed by the researchers. To calculate the mean color value of each color plane whole pixels of the color planes are used which is also time consuming. Time complexity to calculate the mean color value of color planes are reduced by using the smaller size of image. In this paper we discuss proposed two techniques of Content Based Image Retrieval (CBIR) to improve color feature using Truncated Discrete Cosine Transform (DCT) image Block Truncation Coding . We discuss the comparative study of full image BTC, truncated DCT image BTC and truncated DCT image sub block BTC using RGB, YCbCr, YUV, CXY, CIEL*a*b*, R'G'I color planes. Transformed DCT image low frequency coefficients are used to generate the truncated DCT image. In this paper we take up to fifth level DCT truncation and each level truncated DCT image BTC performance is compared with the full image BTC. Proposed techniques are tested over the image database which includes 1200 images having 15 classes. Similarity between query image feature vector and database feature vector measured here using Euclidean distance and Bray Curtis distance similarity. Performance of the proposed approaches measured by using overall average precision and recall cross over point.

Key words: SVD; Precision; Recall; Euclidean Distance; Bray Curtis Distance.

I. INTRODUCTION

In this paper we use truncated DCT image to calculate the upper and lower mean color values of image. Truncated DCT image is obtained on the low frequency DCT coefficients, which contain more information of image. Feature vector which include upper mean and lower mean color value of this truncated DCT image improves the result. We truncate DCT image up to the 5^{th} level to compare the retrieval result. This paper organized in the following sections: Section 2, review of BTC. Section 3, DCT truncation. Section 4, Color Planes Considered for the Proposed Methods. Section 5,Feature Vector Extraction. Section 6, Implementation. Section 7, Feature Vector Matching, Section 8, Performance of CBIR, Section 9, Result Analysis and Discussion. Section 10, Conclusion future work.

II. BLOCK TRUNCATION CODING

BTC method divides the image into R, G, and B components. Then computes inter-band average image (IBAI) which is the average of R, G and B components and mean of inter-band average image is taken as threshold. The bitmap is then created by comparing each pixel with this threshold value. If a pixel in the inter-band average image is greater than or equal to the threshold, the corresponding pixel position of the bitmap will have a value '1' otherwise '0'. Two mean colors one for the pixels greater than or equal to the threshold and other for pixels smaller than threshold are calculated. In the modified BTC threshold for each color is computed and upper mean and lower mean colors are calculated as follows threshold for RED color plane, GREEN color plane and BLUE color planes are

Volume 02, Issue 12, [December – 2015] ISSN (Online):2349–9745 ; ISSN (Print):2393-8161

computed as mean of each color plane MR,MG,MB.

$$MR = \frac{1}{(m \times n)} \sum_{i=1}^{m} \sum_{j=1}^{n} RED(i, j) \quad (1)$$

$$MG = \frac{1}{(m \times n)} \sum_{i=1}^{m} \sum_{j=1}^{n} GREEN(i, j) \quad (2)$$

$$MB = \frac{1}{(m \times n)} \sum_{i=1}^{m} \sum_{j=1}^{n} BLUE(i, j) \quad (3)$$

Then bitmap images for each color is computed as

$$BM\_R(i, j) = \begin{cases} 1\ if & RED(i, j) > MR \\ 0\ if & RED(i, j) \le MR \end{cases} \quad (4)$$

$$BM\_G(i, j) = \begin{cases} 1\ if & GREEN(i, j) > MG \\ 0\ if & GREEN(i, j) \le MG \end{cases} \quad (5)$$

$$BM\_B(i, j) = \begin{cases} 1\ if & BLUE(i, j) > MB \\ 0\ if & BLUE(i, j) \le MB \end{cases} \quad (6)$$

After creation of bitmap image three upper mean colors and three lower mean colors are computed as

$$Upper\_col\_red1 = \frac{1}{\sum_{i=1}^{n} \sum_{j=1}^{m} BM\_R(i, j)} \sum_{i=1}^{n} \sum_{j=1}^{m} BM\_R(i, j) \times RED(i, j) \quad (7)$$

$$Upper\_col\_green1 = \frac{1}{\sum_{i=1}^{n} \sum_{j=1}^{m} BM\_G(i, j)} \sum_{i=1}^{n} \sum_{j=1}^{m} BM\_G(i, j) \times GREEN(i, j) \quad (8)$$

$$Upper\_col\_blue1 = \frac{1}{\sum_{i=1}^{n} \sum_{j=1}^{m} BM\_B(i, j)} \sum_{i=1}^{n} \sum_{j=1}^{m} BM\_B(i, j) \times BLUE(i, j) \quad (9)$$

$$Lower\_col\_red1 = \frac{1}{m \times n - \sum_{i=1}^{n} \sum_{j=1}^{m} BM\_R(i, j)} \sum_{i=1}^{n} \sum_{j=1}^{m} \{1 - BM\_R(i, j)\} \times RED(i, j) \quad (10)$$

$$Lower\_col\_green1 = \frac{1}{m \times n - \sum_{i=1}^{n} \sum_{j=1}^{m} BM\_G(i, j)} \sum_{i=1}^{n} \sum_{j=1}^{m} \{1 - BM\_G(i, j)\} \times GREEN(i, j) \quad (11)$$

$$Lower\_col\_blue1 = \frac{1}{m \times n - \sum_{i=1}^{n} \sum_{j=1}^{m} BM\_B(i, j)} \sum_{i=1}^{n} \sum_{j=1}^{m} \{1 - BM\_B(i, j)\} \times BLUE(i, j) \quad (12)$$

These upper mean color and lower mean color together used as feature vector for each image in the database. Proposed approaches are

1. DCT truncated image BTC
2. Block based DCT truncated image BTC

These two approaches are compared with the full color image BTC[2,34].

III. DCT TRUNCATION

Discrete Cosine Transform (DCT) is a technique, which converts signal into the frequency components . Some of the researchers have presented image retrieval methods in the DCT domain for the JEPG image format. Even for the MPEG compression. Some of the methods use partial DCT coefficients for the feature vector calculation [4, 5]. 2-D DCT for an image computed by using 1-D DCT on each row followed by each column. Two dimensional DCT is calculated mathematically as follows:

For forward DCT:

$$F(u,v)=\alpha(u)\alpha(v)\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i,j)\cos[\frac{(2i+1)u\pi}{2N}]\cos[\frac{(2j+1)v\pi}{2N}] \quad (13)$$

For u,v=0,1,2,……………..N-1

and for inverse DCT:

$$f(i,j)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\alpha(u)\alpha(v)F(u,v)\cos[\frac{(2i+1)u\pi}{2N}]\cos[\frac{(2j+1)v\pi}{2N}] \quad (14)$$

For i,j=0,1,2,…………N-1

Where

$$\alpha(j)=\begin{cases}\sqrt{1/N} & if \quad j=0\\ \sqrt{2/N} & if \quad j=1,2\ldots \quad N-1\end{cases}$$

F(u,v) are the DCT coefficients and f(i,j) is the given image pixel value.

- F(0,0) is the DC coefficient which represent average luminance.
- Remaining coefficients are the AC coefficients which represent the intensity change in the pixel value.
- The low frequency coefficients concentrate at the upper left corner of the transformed plane and high frequency coefficients are the remaining part.

As per above discussion when we apply DCT on given N × N image we get N × N transformed image DCT coefficients. Not all the DCT coefficients are used as a feature vector. To truncate the DCT coefficients we select the upper left corner low frequency coefficients [5,6]. This is N/2 × N/2 upper part of NXN image as shown in Fig. 1

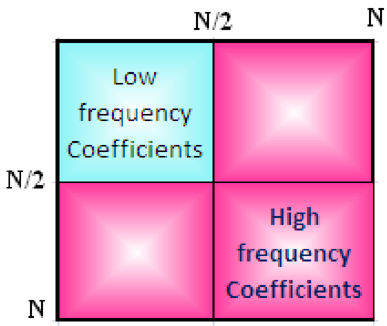

Fig. 1 DCT low frequency coefficients and high frequency coefficients.

Selected part is low frequency coefficients having size N/2 × N/2.Now when we apply inverse DCT on this selected part then we get same image having size N/2 × N/2. So these image coefficients are used to compute feature vector. This is the first stage truncation of DCT. For the second stage apply DCT on the first stage upper part low frequency coefficient having size N/2 × N/2 image .We will get N/2 × N/2 DCT coefficients in which N/4 × N/4 upper corner low frequency coefficient and remaining high frequency coefficients. Select only low frequency coefficients and apply the same method which is applied for the first stage. As shown in Fig.2. As like this we can apply this truncation further. We use this truncation up to the fifth level (on image having size 256x256) and find out the impact of this truncation on the retrieving results for each truncation level.

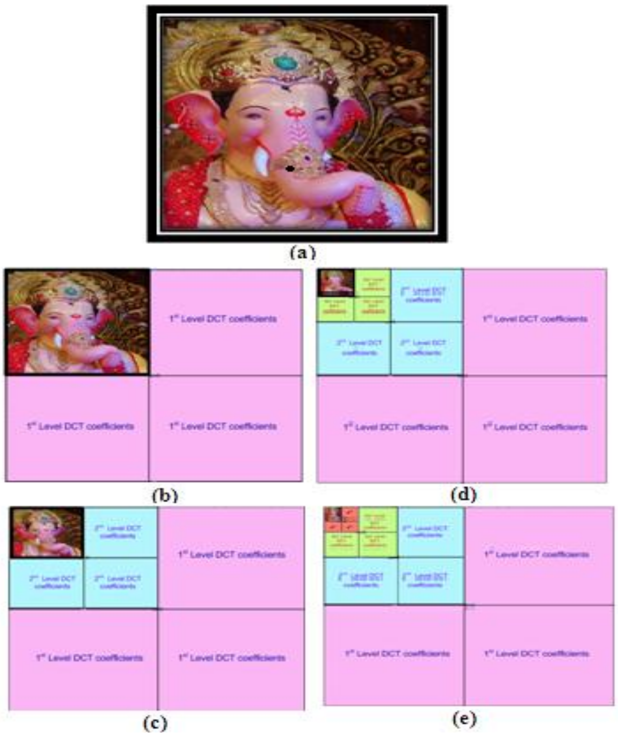

Fig. 2 (a) Given Image,(b)(c)(d)and(e) are first level, second level, third level and fourth level DCT truncation of the given image respectively.

IV. COLOR PLANES CONSIDERED FOR THE PROPOSED METHODS

Compact and effective Color feature is the aim of this research. To find out which color plane performing better for the proposed algorithms different color planes are used. So comparative study of the different color planes are explained in this paper.

4.1 YCbCr color plane [3]

We have used YCbCr color plane. Where Y gives luminance and Cb and Cr gives chromaticity values of color 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 color image for respective R,G, and B components.

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

4.2 YUV Color plane [5]

In YUV color plane Y represent the luminance and U, V represent the chrominance information of given color image. Color conversion of RGB to YUV is given by equation (16).

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ -0.14713 & -0.22472 & 0.436 \\ 0.615 & -0.51498 & 0.10001 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{16}$$

4.3 HSV Color plane

H (Hue), S(Saturation) and V(Value) is considered as Tint, Shade and Tone by artists. Value represent the intensity of color .The Hue and Saturation components are intimately related to the way human eye perceives color resulting in image processing algorithms with physiological basis. Conversion formula from RGB to HSV is given in the equation (5)(6)(7).

$$H = \begin{cases} \theta, & \text{if } B \le G \\ 360 - \theta, & \text{if } B > G \end{cases} \tag{17}$$

Where

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

$$S = 1 - \frac{3}{R+G+B}\left[\min(R,G,B)\right] \tag{18}$$

$$V = \frac{1}{3}(R+G+B) \tag{19}$$

4.4 R'G'I Color plane [49]

This Color plane can be used to separate low and high frequencies in the image without losing any information from the image. This in turn allows both distinguishing possible ROIs and retrieving their proper color for further ROI analysis. To get R'G'I components the conversion of RGB to R'G'I components is necessary. The RGB to R'G'I conversion matrix given in equations (2) gives the R',G',I components of image for respective R, G, B components.

$$R' = \frac{R \times 256}{(R+G+B)} \tag{20}$$

$$G' = \frac{G \times 256}{(R+G+B)} \tag{21}$$

$$I = \frac{(R+G+B)}{3} \tag{22}$$

4.5 CXY Color Plane

Conversion of RGB to CXY color plane is given in the equation (3)

$$\begin{bmatrix} C \\ X \\ Y \end{bmatrix} = \begin{bmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.14 \\ 0.000 & 0.006 & 1.116 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{23}$$

4.6 CIEL*a*b* Color Plane

CIEL*a*b* (Brightness, red-green and yellow blue content) system gives quantitative expression to the Munsell system of color classification [38]. The transform from RGB can be made as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.490 & 0.310 & 0.200 \\ 0.177 & 0.813 & 0.011 \\ 0.000 & 0.010 & 0.990 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (24)$$

$$L^* = 25\left(\frac{100Y}{Y_0}\right) - 16, \quad 1 \le 100Y \le 100 \quad (25)$$

$$a^* = 500\left[\left(\frac{X}{X_0}\right)^{1/3} - \left(\frac{Y}{Y_0}\right)^{1/3}\right] \quad (26)$$

$$b^* = 200\left[\left(\frac{Y}{Y_0}\right)^{1/3} - \left(\frac{Z}{Z_0}\right)^{1/3}\right] \quad (27)$$

$X_0 Y_0$ & Z_0 Mean tristimulus values for reference white.

V. FEATURE VECTOR EXTRACTION

5.1 Full DCT Truncated Image BTC

Feature vector computed of a given image is explained in Fig.3 We use seven color planes for comparison of result of proposed methods. Upper mean color and lower mean color for each DCT truncated image is computed as explained in above section 3. Feature vector computations steps for these seven color planes are same and given as.

a. For DCT truncation read image having size NXN color image.
b. Apply 2D-DCT on the given image.
c. c. Select upper left corner low frequency coefficients having size N/2 × N/2 for 1st level(L=1), N/4 × N/4 for 2nd level(L=2), N/8 × N/8 for 3rd level(L=2) and so on up to the L=5 level.
d. Apply inverse DCT on selected part to obtain the original image at that particular L level.
e. Compute BTC feature vector.
f. Repeat the steps c to e on the above first level DCT truncated image to get L level DCT truncation.

In this proposed approach L=5. That means up to the 5th level DCT truncation is used. For each level BTC feature (upper mean and lower mean color) values are different but feature vector size is same for each level. For each color space feature vector is calculated and feature vector of each color space concatenated to form feature vector of color image.

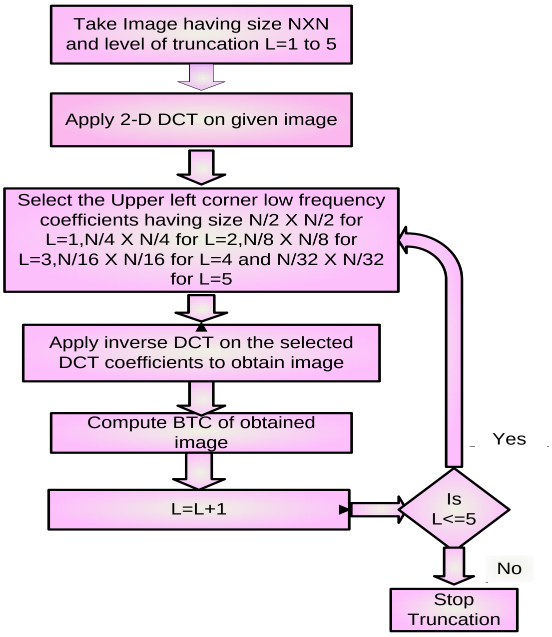

Fig. 3 DCT truncation and feature vector computation

5.2 DCT Truncated Image sub block BTC

As explained in above section at each level we get three upper mean color and three lower mean color value as a feature vector. To boost the retrieval efficiency each level DCT truncated image divided into 8x8 sub block .For each sub block upper mean and lower mean color value is computed and stored as a feature vector. The feature vector size using this method increases which is for 256x256 color image at first level is 1536,at 2nd level it is 384,at 3rd level it is 96 ,at 4th level it is 24 and at the 5th level it is 6. As result retrieval efficiency also increases.

VI. IMPLEMENTATION

The implementation of the discussed 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 augmented image[8] database of 1200 variable size images spread across 15 categories of tribal people, motor bike, elephant, horse, flower, historical mountain, beach, sunset, mountain, flower lawn etc. The categories and distribution of the images is shown in table 1. Sample images are shown in Fig.4.

Fig. 4 Sample images from database.

Table 1. Category-Wise Distribution of Image Database

Name of Category	No. of Images	Name of Category	No. of Images	Name of Category	No. of Images
Motorbikes	100	Beaches	100	Buses	100
Elephants	100	Flowers	100	Tribal Peoples	68
Flying Birds	63	Flower lawn	48	Butterfly Scenery	52
Dinosaurs	100	Mountains	62	Guitar	59

VII. FEATURE VECTOR MATCHING

When a user submits a query image, we need to compute the feature vector as before. After that query image and database image matching is done using similarity measures. In this paper two similarity measures are used Euclidean Distance(ED) and Bray Curtis Distance (BCD) for the comparison. Minkowski (Euclidean distance when r=2) distance is computed between each database image & query image on feature vector to find set of images falling in the class of query image.

$$Ed(Q,I) = \left(\sum_{M=0}^{M-1} \left| H_Q - H_I \right|^r \right)^{1/r} \tag{28}$$

Where Q-Query image
I- Database image.
H_Q-Feature vector query image.
H_I-Feature vector for database image.
M-Total no of component in feature vector.

Bray Curtis Distance is computed between query image and database image using Eq. 29

$$Bd(Q,I) = \frac{\sum_{k=1}^{n} \left| H_{Qk} - H_{Ik} \right|}{\sum_{k=1}^{n} (H_{Qk} - H_{Ik})} \tag{29}$$

Where n-Total no of component in feature vector.
Q-Query image
I-Database image
H_{QK}-Feature vector query image.
H_{Ik}-Feature vector for database image.

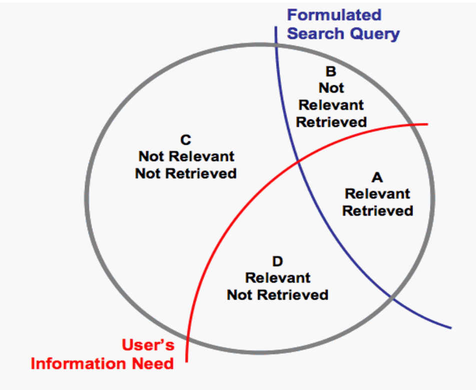

Fig. 5 Evaluation of CBIR

VIII. PERFORMANCE OF CBIR

Performance of image retrieval system can be analyzed by using two parameters precision and recall. As shown in Fig.5. Testing the effectiveness of the image search engine is about testing how well can the search engine retrieve similar images to the query image and how well the system prevents the return results that are not relevant to the source at all in the user point of view. A sample query image must be selected from one of the image category in the database. When the search engine is run and the result images are returned, the user needs to count how many images are returned and how many of the returned images are similar to the query image. The first measure is called Recall. All the relevant images from the database is recall. The Eq. 30 for calculating recall is given below. The second measure is called Precision. It is accuracy of a retrieval system to present relevant as well as non relevant images from the database which is mathematically given in Eq. 31

$$Recall = \frac{Number\ of\ relevant\ images\ retrieved(A)}{Total\ no.of\ relevant\ images\ in\ the\ database(A+D)} \quad (30)$$

$$Precision = \frac{Number\ of\ relevant\ images\ retrieved(A)}{Total\ no.of\ images\ retrieved(A+B)} \quad (31)$$

Precision stipulate the accuracy of the result of CBIR technique and recall specifies the completeness .Higher value of precision and recall indicated better image retrieval. Usually a threshold will be determined on trial and error basis which is use to retrieve the images similar to query having distance less than pre-determined threshold. As this trial and error is time consuming process and never comes up with fix or ideal threshold value, what we do here for retrieval is we are sorting the distances in ascending order and selecting first N number images to retrieve the images relevant to the query image out of it. N is the total number of images of that particular category in the database images. This generates the cross over points [2]-[9]-[2] of precision and recall values for that particular category our results. This means that as average precision goes on decreasing recall goes on increasing with respect to number of images retrieved. The crossover point of the precision and recall play very important role in the performance of analysis of the image retrieval method. As this crossover, point value of precision equal to the recall .In ideal situation the height of precision and recall cross over point should be at the value one.

IX. RESULT ANALYSIS AND DISCUSSION

The proposed approaches for feature vector extraction is tested on the augmented Wang database[23] spread over 1200 variable size image categories.

Table 2. Overall Average Precision And Recall Cross Over Point For Full Image BTC Approach.

Color plane	Overall average precision/recall cross over point using ED	Overall average precision/recall cross over point using BCD
YCbCr	0.60287	0.60287
RGB	0.54423	0.54423
YUV	0.51861	0.51861
R'G'I	0.58635	0.58635
HSV	0.48213	0.48213
CXY	0.5072	0.5072
CIE L*a*b*	0.62563	0.62563

Table 2 shows the overall average precision and recall crossover point for full image BTC using seven color planes with two similarity measures. It is noted that CIE L*a*b* color performance using Euclidean Distance similarity is above 62%.YCbCr color plane performance is also closer to the CIEL*a*b* and it is above 61% using Bray Curtis Distance similarity.

Table 3 shows the overall average precision and recall crossover point for DCT truncated image BTC. In this approach CIEL*a*b* color performance is above 62% using Euclidean Distance similarity at 4th level. It slightly improves compare to the full image BTC. Table IV shows overall average precision and recall cross over point for DCT truncated image sub block BTC .It is noted that for YCbCr color plane performance at 3rd level is above 67% with Bray Curtis Distance similarity.

Table 3. Overall Average Precision And Recall cross over point for DCT truncated image BTC approach.

Similarity	DCT Truncation level	YCbCr	RGB	YUV	R'G'I	HSV	CXY	CIE L*a*b*
ED	1st level	0.5990	0.5378	0.5169	0.4801	0.4801	0.5009	0.6246
BCD		0.6146	0.5454	0.5282	0.5093	0.5093	0.5053	0.5982
ED	2nd level	0.5973	0.5267	0.5107	0.4821	0.4821	0.4927	0.6255
BCD		0.6110	0.5317	0.5204	0.5101	0.5101	0.4954	0.5982
ED	3rd level	0.5974	0.5227	0.5082	0.4860	0.4860	0.4940	0.6264
BCD		0.6095	0.5285	0.5171	0.5064	0.5064	0.4934	0.5972
ED	4th level	0.5969	0.5207	0.5070	0.4855	0.4855	0.4907	0.6280
BCD		0.6084	0.5323	0.5166	0.5026	0.5026	0.4971	0.5935
ED	5th level	0.5915	0.5187	0.5078	0.4767	0.4767	0.4882	0.6182
BCD		0.6018	0.5255	0.5142	0.4988	0.4988	0.4882	0.5862

Table 4. Overall Average Precision And recall cross over point for DCT truncated image sub block BTC approach.

Similarity	DCT Truncation level	YCbCr	RGB	YUV	R'G'I	HSV	CXY	CIE L*a*b*
ED	1st level	0.5812	0.5591	0.5004	0.5755	0.5770	0.5006	0.6010
BCD		0.6483	0.5568	0.5206	0.6149	0.6019	0.5092	0.5287
ED	2nd level	0.6179	0.5882	0.5393	0.6139	0.6122	0.5315	0.6329
BCD		0.6719	0.5893	0.5541	0.6456	0.6351	0.5408	0.5637
ED	3rd level	0.6232	0.5941	0.5547	0.6271	0.6080	0.5443	0.6473
BCD		0.6725	0.5937	0.5643	0.6505	0.6331	0.5483	0.5860
ED	4th level	0.5834	0.5324	0.5000	0.5665	0.5347	0.4918	0.6246
BCD		0.6303	0.5388	0.5131	0.5940	0.5618	0.4986	0.5906
ED	5th level	0.5915	0.5187	0.5078	0.5713	0.4767	0.4882	0.6182
BCD		0.6018	0.5255	0.5142	0.5800	0.4988	0.4882	0.5862

X. CONCLUSION

In the case of full image BTC CIEL*a*b* color performance is higher than the other color plane. In truncated DCT image BTC CIEL*a*b* color again at fourth level out performing than the other level, other color plane and also than the full image BTC with Euclidean distance similarity. In truncated DCT sub block image BTC YCbCr color plane outperforming at third level truncated DCT than the other level and other color plane with Bray Curtis distance similarity. Second level truncated DCT image sub block BTC performance also close to the third level using Bray Curtis distance (BCD) similarity. For the future we can apply this algorithm for the face recognition in addition with the texture feature.

REFERENCES

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

[2] H.B.Kekre, Sudeep D. Thepade, Shrikant Sanas," Improving Performance of Multileveled BTC Based CBIR Using Sundry Color Spaces ",(IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 07, 2010, 2471-2476 .

[3] 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.

[4] 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.

[5] Akram A. Moustafa and Ziad A. Alqadi," Color Image Reconstruction Using A New R'G'I Model"Journal of Computer Science 5 (4): 250-254, 2009. ISSN 1549-3636 © 2009 Science Publications.

[6] Pdamshree Suresh,RMD Sundaram, Aravindhan Arumugam,” Feature Extraction in Compressed Domain for Content Based Image Retrieval ”,International Conference on Advanced Computer Theory and Engineering.2008.

[7] Hee-Jung Bae, Sung-Hwan Jung ,"Image Retrieval Using Texture Based on DCT "International Conference on Information, Communications and Signal Processing ICICS '97 Singapore,7803-3676-3/1997 IEEE ,9-12 September 1997.

[8] http://wang.ist.psu.edu/docs/related/Image.orig .

