A Review on Feature Extraction Techniques for CBIR with different BTC methods

[1] R.Sahaya Jeya Sutha, [2] Dr.D.S.Mahendran, [3] Dr.S.John Peter

[1] Research Scholar, Manonmaniam Sundaranar University, Tirunelveli, India,
[2] Principal, Aditanar College of Arts and Science, Tiruchendur, India
[3] Asso.Professor & Head, Dept. of Computer Science, St.Xavier's College, Palayamkottai, India

Abstract*— Content Based Image Retrieval (CBIR) is one of the fastest growing research areas to search the relevant images from the large image collections by analyzing the content of the user required query image. The retrieval time in CBIR is reduced by extracting the image features from the compressed image instead of the original image. The Block Truncation Coding (BTC) is a very efficient compression technique which requires least computational complexity and it can also effectively employed to index images in database for CBIR applications. Various improved versions of BTC methods are available for compression which is suitable for CBIR. This paper gives the review of feature extraction techniques for CBIR with different BTC methods such as Ordered Dithering BTC(ODBTC), Error Diffusion BTC (EDBTC), Dot Diffusion (DDBTC).

Index Terms—***Bitmap feature, Block truncation coding (BTC), Color feature, Content based image retrieval (CBIR).***

I. INTRODUCTION

In today's electronic world, the development of internet and the inventions of efficient image capturing devices is growing exponentially. To utilize these high resolution digital image databases in an effective way, efficient image retrieval techniques are required.

The aim of image retrieval is to determine the related images from a collection of images in the database. The Content Based Image Retrieval (CBIR) allows the user to search by providing an example image (query image).

The CBIR systems operate in pixel (spatial) domain and compressed domain. In the pixel domain approach, low level features like color, shape and texture are extracted directly from images [24]. These feature extractions are very time consuming while comparing with all images in the database. But in the compressed domain approach, features extracted directly from the compressed image without decompressing it. Nowadays for reducing the storage space most of the images are stored in the storage devices in compressed formats. So, CBIR in compressed domain is becoming an important issue [7]-[14].

Block Truncation Coding (BTC) is one of the best, simple and efficient image compression method and play an important role in image retrieval also. Delp and Mitchell in 1979, introduced the BTC technique for image compression purpose, as in [1]. Many improved methods of BTC compressions like Error Diffusion BTC (EDBTC), Ordered Dither Block Truncation Coding (ODBTC), Dot diffusion BTC (DDBTC) developed for image compressions which are used for CBIR, as reported in [1]-[6], [11]–[33]. The advantage of BTC based CBIR is that the feature vector of image is independent of the image dimensions (no need of all database images to be of same size to that of query image), as in [18], [33].

This paper gives a brief explanation about the general steps of BTC based compression techniques and how features are extracted directly from that compressed image without decompressing the image for CBIR. And also this paper review various feature extraction techniques for CBIR with different BTC methods. The BTC based CBIR using CCF, CHF, BPF, etc. for extracting color and bitmap features are discussed briefly in this paper.

II. CBIR USING BTC BASED COMPRESSION METHODS

The general steps for BTC compression based image retrieval schemes involve three phases: compression, indexing and searching. In the compression phase, RGB color image is compressed using one of the BTC method and it yields two color quantizer and one bitmap image. The indexing phase extracts the image features from the quantizers and bitmap image got in the previous phase which is later stored in database as feature vector. In the searching phase, the feature vector is extracted from the user submitted query image like the same method is matched with the feature vectors in the database by similarity assessments. The image retrieval system finally returns a set of related images to the user.

The Fig.1 illustrates the schematic diagram of compression phase. Consider the original image F is the $M \times N$ size RGB color space and f_R, f_G, f_B is the red, green, and blue color channel of that image. Now divide the image into several non-overlapping image blocks of size $m \times n$. Suppose $f(i,j) = \{f_R(x,y), f_G(x,y), f_B(x,y)\}$ be an image block with index (i,j), where $i = 1, 2, \ldots, \frac{M}{m}$ and $j = 1, 2, \ldots, \frac{N}{n}$.

To find the minimum quantizer for each image block, combine the minimum pixel value over red, green, and blue color spaces as [27]-[29],

$$q_{min}(i,j) = \{min_{\forall x,y}\, f_R(x,y), min_{\forall x,y}\, f_G(x,y), min_{\forall x,y}\, f_B(x,y)\}$$

In the same manner, the maximum quantizer also derived.

$$q_{max}(i,j) = \{max_{\forall x,y}\, f_R(x,y), max_{\forall x,y}\, f_G(x,y), max_{\forall x,y}\, f_B(x,y)\}$$

where $x = 1, 2, \ldots, m$ and $y = 1, 2, \ldots, n$. The color quantizer size reduced as $\frac{M}{m} \times \frac{N}{n}$ from the original image size $M \times N$. Here the maximum quantizer is looking brighter than that of the minimum quantizer.

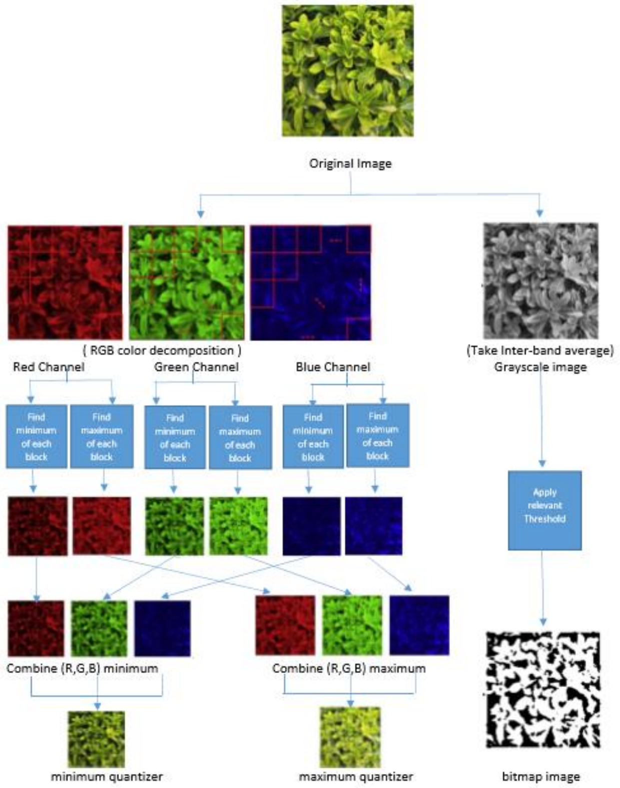

Fig. 1 Schematic diagram of Compression phase

On the other side, convert the color image into grayscale image by its inter-band average value as

$$\bar{f}(x,y) = \frac{1}{3}[f_R(x,y) + f_G(x,y) + f_B(x,y)]$$

Then construct the bitmap image from the grayscale image by applying suitable threshold technique. If a pixel value is lesser than the threshold then it is turned to 0 (black pixel) otherwise it turns to 1 (white pixel). Table 1 shows different BTC based compression methods using various threshold techniques.

The traditional BTC approach using a single threshold value obtained from the mean value of the pixels in an image block. The ODBTC employs the void-and-cluster dither matrix of the same size as an image block to generate the bitmap image. Different threshold value applied for every pixel in a region. The EDBTC employs the error kernel (diffused matrix) to generate bitmap image. Floyd-Steinberg method is one of the most famous error diffusion method. The error computed from the difference between the input pixel value and the output value is dispersed to nearby neighbors. The DDBTC for color image performs the thresholding operation by incorporating the diffused matrix and class matrix concurrently to generate the bitmap image. The output bitmap image having the same number of pixels as in the source image.

The indexing phase extracting the feature vector (compact code) from the outcome of first phase and stored in a feature database. For indexing two types of features extracted: Color and Bit Pattern feature. The color feature is derived from the two color quantizers, and the Bit Pattern feature is derived from the bitmap image. The Fig. 2 illustrates the schematic diagram of indexing phase for the BTC based CBIR.

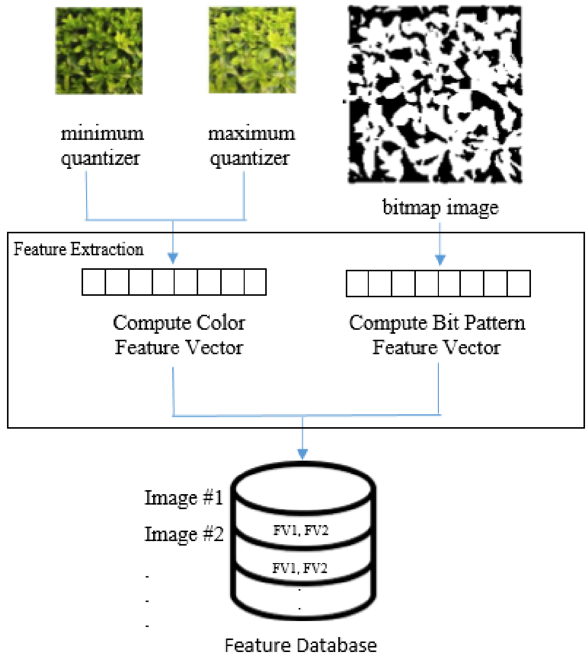

Fig. 2 Schematic diagram of indexing phase

TABLE I

VARIOUS THRESHOLD TECHNIQUES USED WITH DIFFERENT BTC BASED COMPRESSION METHODS

Traditional BTC	ODBTC	EDBTC	DDBTC
Mean value of pixels in a block	void-and-cluster Dither Matrix	Diffused matrix	Diffused matrix and Class matrix

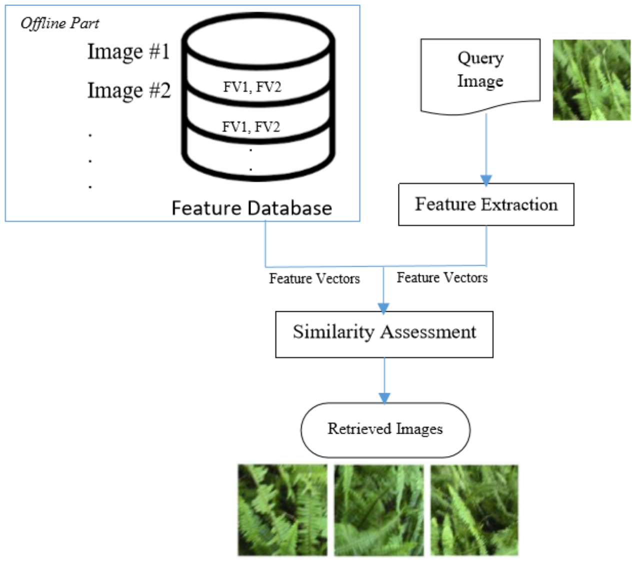

Fig. 3 Schematic diagram of searching phase

During the searching phase, user gives the query image for finding the relevant images. The feature vector for that query is computed like the previous way. Using a similarity assessments, this feature vector is compared with the existing feature vectors in the feature database. A set of relevant images are retrieved based on the similarity score. The Fig. 3 illustrates the schematic diagram of searching phase.

III. FEATURE EXTRACTION FOR BTC BASED CBIR

Features are used to represent the image content in the CBIR systems. The features are extracted from the image automatically without any manual intervention. These automated approaches are computationally is very expensive, difficult and tend to be domain specific. The set of features in feature extraction (FE) is called image signature or feature vector, is generated to accurately represent the content of each image in the database. The size of the feature vector is much smaller than the original image.

In this paper, some important BTC based feature extraction techniques like Color co-occurrence feature (CCF), Color Histogram Feature (CHF), Bit Pattern Feature (BPF) are discussed.

A. Color co-occurrence feature (CCF)

This section elaborates how to derive a CCF image feature from the two color quantizers of BTC based compression. The Fig. 4 illustrates the schematic diagram CCF construction.

Firstly, the minimum and maximum color quantizers are indexed using a specific color codebook. The LBG Vector Quantization (LBGVQ) generates a codebook. The color codebook can be different or same for converting minimum and maximum color quantizers. The color indexing process on RGB color space can be done by mapping the three (R, G, B) tuples into color codebook. The collection of replaced indexed color returns indexed color matrix.

The color co-occurrence matrix is subsequently constructed from these indexed color matrix, as in [12]. Subsequently, the CCF is as below from the color co-occurrence matrix at the end of computation.

$$CCF(t_1,t_2) = Pr\left\{\tilde{\imath}_{min}(i,j) = t_1, \tilde{\imath}_{max}(i,j) = t_2 \mid i = 1,2,\dots,\frac{M}{m}; j = 1,2,\dots,\frac{N}{n}\right\},$$

for $t_1, t_2 = 1,2,\dots,N_c$.

N_c denote the color codebook size, $\tilde{\imath}_{min}$ and $\tilde{\imath}_{max}$ are minimum and maximum indexed color tables. Pr is the occurrence probabilities (histogram) of the associated indexed color image form the CCF.

B. Color Histogram Feature (CHF)

The CHF is an effective feature for depicting the brightness and contrast of a color image. Firstly, the minimum and maximum color quantizers are indexed using a specific color codebook. The CHF is simply constructed from these two indexed color quantizers.

The Fig. 5 illustrates the schematic diagram of CHF construction. After the color indexing process, the CHF can be easily derived from the indexed minimum and maximum quantizers to form an image feature vector, as in [29].

$$CHF_{min}(k) = Pr\left\{\tilde{\imath}_{min}(i,j) = k \mid i = 1,2,\dots,\frac{M}{m}; j = 1,2,\dots,\frac{N}{n}\right\}$$

for $k = 1,2,\dots,N_{min}$, and

$$CHF_{max}(k) = Pr\left\{\tilde{\imath}_{max}(i,j) = k \mid i = 1,2,\dots,\frac{M}{m}; j = 1,2,\dots,\frac{N}{n}\right\}$$

for $k = 1,2,\dots,N_{max}$

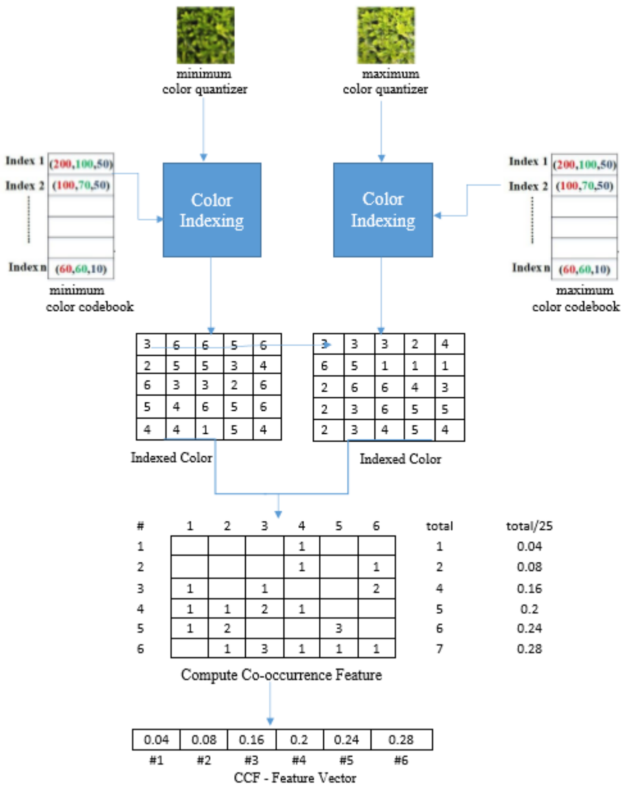

Fig. 4 Schematic diagram of the CCF construction

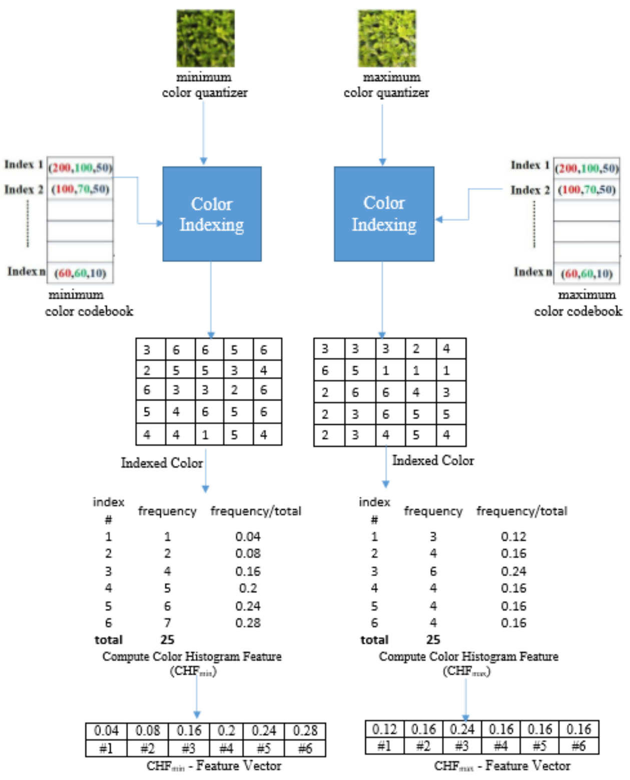

Fig. 5 Schematic diagram of the CHF construction

where N_{min} and N_{max} are same as color code book size. Pr is the occurrence probabilities (histogram) of the associated indexed color image form the CHF.

C. Bit Pattern Feature (BPF)

The Bit Pattern Feature (BPF) is used to characterize the edge and visual texture pattern of an image. The Fig. 6 illustrates the schematic diagram for constructing the BPF.

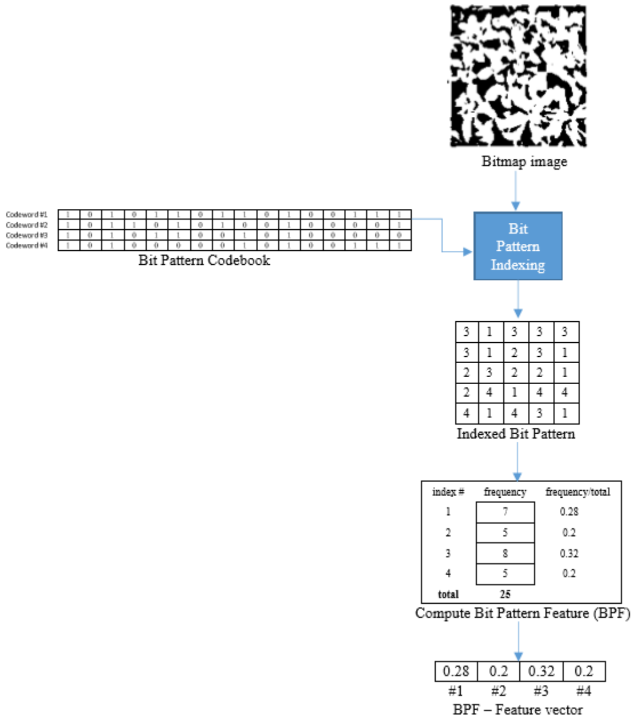

Fig. 6 Schematic diagram for constructing the BPF

The bitmap image produced from the BTC based compression is firstly indexed using the trained bit pattern codebook. The classical LBGVQ is used to generate the representative codebook of the bitmap image.

The bit pattern indexing is performed between the blocks of bitmap image $bm(i,j)$ and the codeword B_i of the bit pattern codebook by using the following criterion:

$$\tilde{b}(i,j) = \arg\min_{k=1,2,\ldots N_b} \delta_H\{bm(i,j), B_k\}$$

where $i = 1,2,\ldots,\frac{M}{m}; j = 1,2,\ldots,\frac{N}{n}$, N_b is bit pattern codebook size and $\delta_H\{.,.\}$ denotes the Hamming distance between the two compared binary patterns. The index value is the minimum distance mapped codeword index number.

Afterwards, the BPF feature vector value derived by computing the occurrence probability (histogram) of the indexed bit pattern defined as in [28].

$$BPF(t) = Pr\left\{\tilde{b}(i,j) = t \middle| i = 1,2,\ldots,\frac{M}{m}; j = 1,2,\ldots,\frac{N}{n}\right\}$$

for $t_1, t_2 = 1,2,\ldots,N_b$ and N_b is the size of the bit pattern codebook. The dimension of the BPF is N_b.

IV. SIMILARITY MEASUREMENT

The important step in CBIR is similarity distance computation between query image and the set of images in the database. The distance metric chosen is the important task for the user because the image retrieval performance is very sensitive to this. Both color feature and bitmap feature vectors are compared in a combined manner. The weightage values can be given for the percentage contribution of color and bitmap feature. Finally, it returns the most relevant images according to the highest similarity.

This paper discuss the distance metrics suitable for BTC based CBIR. The distances between the query and target image can be formally defined under various distance metrics, as in [28]-[33].

- ***L1 distance***

$$\delta(query, target) = \alpha_1 \sum_{k=1}^{N_{min}} \left|CHF_{min}^{query}(k) - CHF_{min}^{target}(k)\right| + \alpha_2 \sum_{k=1}^{N_{max}} \left|CHF_{max}^{query}(k) - CHF_{max}^{target}(k)\right| + \alpha_3 \sum_{k=1}^{N_b} \left|BPF^{query}(k) - BPF^{target}(k)\right|$$

- ***L2 distance***

$$\delta(query, target) = \alpha_1 \sum_{k=1}^{N_{min}} \left(CHF_{min}^{query}(k) - CHF_{min}^{target}(k)\right)^2 + \alpha_2 \sum_{k=1}^{N_{max}} \left(CHF_{max}^{query}(k) - CHF_{max}^{target}(k)\right)^2 + \alpha_3 \sum_{k=1}^{N_b} \left(BPF^{query}(k) - BPF^{target}(k)\right)^2$$

- ***χ^2 distance***

$$\delta(query, target) = \alpha_1 \sum_{k=1}^{N_{min}} \left(\frac{CHF_{min}^{query}(k) - CHF_{min}^{target}(k)}{CHF_{min}^{query}(k) + CHF_{min}^{target}(k) + \varepsilon}\right)^2 + \alpha_2 \sum_{k=1}^{N_{max}} \left(\frac{CHF_{max}^{query}(k) - CHF_{max}^{target}(k)}{CHF_{max}^{query}(k) + CHF_{max}^{target}(k) + \varepsilon}\right)^2 + \alpha_3 \sum_{k=1}^{N_b} \left(\frac{BPF^{query}(k) - BPF^{target}(k)}{BPF^{query}(k) + BPF^{target}(k) + \varepsilon}\right)^2$$

- ***Fu distance***

$$\delta(query, target) = \frac{L_2\ distance}{|CHF_{min}| + |CHF_{max}| + |BPF|}$$

- ***Modified Canberra distance***

$$\delta(query, target) = \alpha_1 \sum_{k=1}^{N_{min}} \frac{\left|CHF_{min}^{query}(k) - CHF_{min}^{target}(k)\right|}{CHF_{min}^{query}(k) + CHF_{min}^{target}(k) + \varepsilon} + \alpha_2 \sum_{k=1}^{N_{max}} \frac{\left|CHF_{max}^{query}(k) - CHF_{max}^{target}(k)\right|}{CHF_{max}^{query}(k) + CHF_{max}^{target}(k) + \varepsilon} + \alpha_3 \sum_{k=1}^{N_b} \frac{\left|BPF^{query}(k) - BPF^{target}(k)\right|}{BPF^{query}(k) + BPF^{target}(k) + \varepsilon}$$

Where $\alpha 1$, $\alpha 2$ and $\alpha 3$ denote the similarity weighting constants representing the percentage contribution of the CHF and BPF utilized in the proposed image retrieval process. When a constant is set at 1, meaning the corresponding feature is employed in the distance computation, whereas the value 0 means the feature is disable.

V. CONCLUSION

In this paper, we presented the common steps for performing content based image retrieval in the BTC based compressed domain for RGB color images. We reviewed the feature extraction techniques for CBIR which suitable to work with BTC based compressed image. Specifically, feature extraction techniques for traditional BTC, EDBTC, ODBTC, and DDBTC compressions are covered in this paper. The feature vectors are directly extracted from two maximum and minimum color quantizers and its corresponding bitmap image of BTC based compressions, without doing the decoding process. This paper gives a quick overview of important feature extraction techniques, CCF and CHF for obtaining color features and BPF for bitmap feature. We also reviewed the similarity metrics for comparing images.

REFERENCES

[1] E.J. Delp, O.R. Mitchell, “Image compression using block truncation coding”, IEEE Transactions on Communications COMM-27 (September 1979) 1335–1342.

[2] D. E. Knuth, “Digital halftones by dot diffusion”, ACM Trans. Graph., Vol. 6, No. 4, pp. 245–273, Oct. 1987.

[3] J. Huang et al., “Image indexing using color correlogram,” in Proc.CVPR97, 1997, pp. 762–768.

[4] S.F. Chang, A. Eleftheriadis, R. McClintock “Next-generation content representation, creation, and searching for new-media applications in education”, in IEEE Proceedings, Vol.86, 1998.

[5] K. Knox, "Evolution of Error Diffusion", J. Electonic Imaging, Vol. 8, No. 4, pp 422-429, 1999.

[6] Guoping Qiu, “Color Image Indexing Using BTC”, IEEE Transactions on Image Processing, Vol.12, No.1, pp.93-101, Jan. 2003.

[7] F. Long, H. J. Zhang, D. D. Feng (2003) , ‘Fundamentals of content – based image retrieval’, in: D. D. Feng, W. C. Siu, H. J .Zhang(Eds.), Multimedia Information Retrieval and Management—Technological Fundamentals and Applications, Springer, pp.1–27.

[8] Z. M. Lu, and H. Burkhardt, “Colour image retrieval based on DCT-domain vector quantization index histograms,” Electronics Letters, vol. 41, no. 17, 2005.

[9] Kim, Tae-Su & Kim, Seung-Jin& Lee, Kuhn-Il. (2005). Image Retrieval Based on Co-occurrence Matrix Using Block Classification Characteristics. 3767. 946-956. 10.1007/11581772_83.

[10] Y. Liu et al., “A survey of content based image retrieval with high-level semantics”, Journal of Pattern Recognition, Vol. 40, pp. 262–282, Nov. 2007.

[11] M. R. Gahroudi and M. R. Sarshar, “Image retrieval based on texture and color method in BTC-VQ compressed domain,” in Proc. Int. Symp. Signal Process. Appl., Feb. 2007, pp. 1–4.

[12] J. M. Guo, “Improved block truncation coding using modified error diffusion”, Electron. Lett., Vol. 44, No. 7, Mar. 2008, pp. 462–464.

[13] Dr.H.B.Kekre, Sudeep D. Thepade, “Boosting Block Truncation Coding using Kekre’s LUV Color Space for Image Retrieval”, WASET Int. Journal of Electrical, Computer and System Engineering (IJECSE), Vol. 2, No. 3, pp. 172-180, Summer 2008.

[14] Y. D. Chun, N. C. Kim, and I. H. Jang, Content -based image retrieval using multiresolution color and texture features, IEEE Trans. Multimedia, Vol. 10, No. 6, Pp. 1073 1084, Oct. 2008.

[15] J. M. Guo and M. F. Wu, “Improved Block Truncation Coding Based on the Void-and-Cluster Dithering Approach,” IEEE Trans. Image Processing, vol. 18, no. 1, pp. 211-213, Jan. 2009.

[16] J. M. Guo and Yun-Fu Liu,” Improved Dot Diffusion by Diffused Matrix and Class Matrix Co-Optimization”, IEEE Trans. On Image Processing,, Vol. 18, No. 8, Aug. 2009.

[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), Vol. 2, No. A09, Autumn 2009, pp. 2-14.

[18] Dr. H.B.Kekre et al. “Improved CBIR using Multileveled Block Truncation Coding”, International Journal on Computer Science and Engineering(IJCSE) Vol. 02, No. 07, 2010, 2471-2476.

[19] J. M. Guo, “High efficiency ordered dither block truncation coding with dither array LUT and its scalable coding application,” Digit. Signal Process., Vol. 20, No. 1, pp. 97–110, Jan. 2010.

[20] J. M. Guo and Y. F. Liu, “Improved block truncation coding using optimized dot diffusion,” in Proc. IEEE Int. Symp. Circuits Syst., May–Jun. 2010, pp. 2634–2637.

[21] J. M. Guo and Y.-F. Liu, “Joint compression/watermarking scheme using majority-parity guidance and halftoning-based block truncation coding,” IEEE Trans. Image Process., Vol. 19, No. 8, pp. 2056–2069, Aug. 2010.

[22] Yu, F.-X & Luo, Hao& Lu, Z.-M. (2011). Colour image retrieval using pattern co-occurrence matrices based on BTC and VQ. Electronics Letters. 47. 100 - 101. 10.1049/el.2010.3232.

[23] Dr.H.B.Kekre, Sudeep D. Thepade,” Image Classification using Block Truncation Coding with Assorted Color Spaces”, International Journal of Computer Applications (0975 – 8887) Vol. 44, No.6, April 2012.

[24] P. Poursistani, H. Nezamabadi-pour, R. A. Moghadam, and M. Saeed, “Image indexing and retrieval in JPEG compressed domain based on vector quantization,” Math. and Comp. Modeling, vol. 57, no. 5-6, pp. 1005-1017, 2013.

[25] Jing-Ming Guo, Heri Prasetyo, Huai-Sheng Su. (2013). Image indexing using the color and bit pattern feature fusion. Journal of Visual Communication and Image Representation, Volume 24, Issue 8, November 2013, Pages 1360-1379.

[26] J. M. Guo and Y. F. Liu, “Improved Block Truncation Coding Using Optimized Dot Diffusion”, IEEE Trans. On Image Processing, Vol. 23, No. 3, Mar. 2014.

[27] Jing, Guo&Prasetyo, Heri& Lee, Hua & Yao, Chen-Chieh. (2015). Image retrieval using indexed histogram of Void-and-Cluster Block Truncation Coding. Signal Processing. 123. 10.1016/j.sigpro.2015.11.009.

[28] J. M. Guo, and H. Prasetyo,“Content-Based Image Retrieval Using Features Extracted From Halftoning-Based Block Truncation Coding”, IEEE Transaction on Image Processing , Vol. 24 , No. 3, Mar. 2015.

[29] J. M. Guo, H. Prasetyo, and Jen-Ho Chen, “Content-Based Image Retrieval Using Error Diffusion Block Truncation Coding Features”, IEEE Transactions On Circuits And Systems For Video Technology, Vol. 25, No. 3, Mar. 2015.

[30] J. M. Guo, H. Prasetyo, and N. J. Wang, “Effective image retrieval system using dot-diffused block truncation coding features,” IEEE Trans. on Multimedia, Vol. 17, No. 9, Sep. 2015.

[31] R. Sahaya Jeya Sutha, D.S. Mahendran, Dr. S. John Peter, Dr. (2017). An Overview of Content Based Image Retrieval Techniques. International Journal of Advanced Research Trends in Engineering and Technology (IJARTET) Vol. 4, Special Issue 4, March 2017.

[32] P. Liu, J. M. Guo, C. Y. Wu and D. Cai, "Fusion of Deep Learning and Compressed Domain Features for Content-Based Image Retrieval," in IEEE Transactions on Image Processing, vol. 26, no. 12, pp. 5706-5717, Dec. 2017.

[33] R Sahaya Jeya Sutha & D S Mahendran, Dr. & S John Peter, Dr. (2017). A survey on Block Truncation Coding based Content Based Image Retrieval. INTERNATIONAL JOURNAL OF COMPUTER APPLICATION. 5. 10.26808/rs.ca.i7v6.03.

[34] Liu, Peizhong& Jing, Guo&Chamnongthai, Kosin&Prasetyo, Heri. (2017). Fusion of Color Histogram and LBP-based

Features for Texture Image Retrieval and Classification. Information Sciences. 390. 10.1016/j.ins.2017.01.025.

[35] Liu, Peizhong& Jing, Guo& Wu, Chi-Yi &Cai, Danlin. (2017). Fusion of Deep Learning and Compressed Domain Features for Content-Based Image Retrieval. IEEE Transactions on Image Processing. PP. 1-1. 10.1109/TIP.2017.2736343.

[36] J. M. Guo and H. Prasetyo "Performance Evaluation of Error Diffusion Block Truncation Coding Feature for Color Image Retrieval" Chapter 4, publically available in the link http://sciencegatepub.com/books/gcsr/gcsr_vol4/GCSR-Vol4-Ch4.pdf

R. Sahaya Jeya Sutha received the B.Sc. (Computer Science), M.C.A. (Computer Applications) and M.Phil. (Computer Science) degrees in 1995, 2000 and 2006, respectively from the Manonmaniam Sundaranar University, Tirunelveli. She has qualified the Tamilnadu State Eligibility Test (SET) Lectureship in 2016. She is currently pursuing Ph.D. degree in Computer Applications at the Manonmaniam Sundaranar University, Tirunelveli. She is a life member of ISTE. Her research interests include image retrieval, video retrieval, data hiding and image compression.

D.S.Mahendran is working as the Principal, Aditanar College of Arts and Science, Tiruchendur. He received the M.Sc. Physics and PGDCA degrees from Madurai Kamaraj University and M.Phil. and Ph.D. degrees in Computer Science from Alagappa University, Karaikudi. His research interest includes Computer algorithms, Ad-Hoc Networks, Network Security and Green Network.

S. John Peter is working as Associate Professor in Computer Science at St.Xavier's College, Palayamkottai, Tirunelveli. He received his M.Sc. and M.Phil. degrees in Computer Science from Bharadhidasan University,Tiruchirappalli. He also earned his Ph.D. degree from Manonmaniam Sundaranar University, Tirunelveli. He has published many research papers on clustering algorithms in various conferences, national and international journals.