A Joint Compression and Retrieval Process Using Halftoning-Based BTC Technique

[1]M.Sai Pranavi, [2]Dr. Giri Babu Kande

[1,2]Dept of ECE, Vasireddy Venkatadri Institute of Technology, Guntur, AP, India

Abstract: *This paper presents a new approach for compression and Retrieval. To index color images using the features extracted from the EDBTC it stands for error diffusion block truncation coding. The EDBTC produces a bitmap image and two color quantizers, which processed using vector quantization (VQ) to generate the image feature descriptor. Hear we computed color histogram feature (CHF) and bit pattern histogram feature (BHF), to measure the similarity between a query image and database image. The CHF and BHF are computed from the VQ-indexed color quantizer and VQ-indexed bitmap image, respectively. The distance computed from formula, the proposed indexing method outperforms the former BTC based image indexing and the other existing image retrieval schemes with natural and textural data sets. Thus, the proposed EDBTC is not only examined with good capability for image compression but also offers an effective way to index images for the content based image retrieval system.*

1. INTRODUCTION

Many former schemes have been developed to improve the retrieval accuracy in the content-based image retrieval (CBIR) system. One type of them is to employ image features derived from the compressed data stream. As opposite to the classical approach that extracts an image descriptor from the original image, this retrieval scheme directly generates image features from the compressed stream without first performing the decoding process. This type of retrieval aims to reduce the time computation for feature extraction/generation since most of the multimedia images are already converted to compressed domain before they are recorded in any storage devices. The image features are directly constructed from the typical block truncation coding (BTC) or halftoning-based BTC compressed data stream without performing the decoding procedure. These image retrieval schemes involve two phases, indexing and searching, to retrieve a set of similar images from the database. The indexing phase extracts the image features from all of the images in the database which is later stored in database as feature vector. In the searching phase, the retrieval system derives the image features from an image submitted by a user (as query image), which are later utilized for performing similarity matching on the feature vectors stored in the database. The image retrieval system finally returns a set of images to the user with a specific similarity criterion, such as color similarity and texture similarity.

2. EXISTING SYSTEM

The concept of the BTC is to look for a simple set of representative vectors to replace the original images.BTC compresses an image into a new domain by dividing the original image into multiple non overlapped image blocks, and each block is then represented with two extreme quantizers (i.e., high and low mean values) and bitmap image. Two subimages constructed by the two quantizers and the corresponding bitmap image are produced at the end of BTC encoding stage, which are later transmitted into the decoder module through the transmitter. To generate the bitmap image, the BTC scheme performs thresholding operation using the mean value of each image block such that a pixel value greater than the mean value is regarded as 1 (white pixel) and vice versa.

2.1. Drawbacks

BTC method does not improve the image quality or compression ratio compared with JPEG or JPEG 2000. It often suffers from blocking effect and false contour problems. Making it less satisfactory for human perception.

3. PROPOSED SYSTEM

3.1. Error Diffusion Block Truncation Coding (EDBTC)

The EDBTC produces two color quantizers and a bitmap image, which are further processed using vector quantization (VQ) to generate the image feature descriptor. Herein two features are introduced, namely, color histogram feature (CHF) and bit pattern histogram feature (BHF), to measure the similarity between a query image and the target image in database. The CHF and BHF are computed from the VQ-indexed color quantizer and VQ-indexed bitmap image, respectively.

Advantages

The halftoning - based BTC, namely, Error Diffusion BTC (EDBTC) is proposed to overcome the three above disadvantages of the BTC.EDBTC scheme produces better image quality compared with the classical BTC approaches.
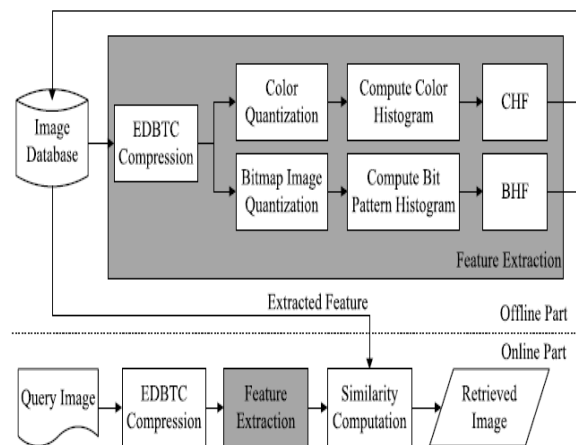

Fig4. *Schematic diagram of the proposed image retrieval framework.*

3.2. EDBTC for Color Images

This section presents a review of the EDBTC with its extension to color image compression. The EDBTC compresses an image in an effective way by incorporating the error diffusion kernel to generate a bitmap image. Simultaneously, it produces two extreme quantizers, namely, minimum and maximum quantizers. The EDBTC scheme offers a great advantage in its low computational complexity in the bitmap image and two extreme quantizers generation. In addition, EDBTC scheme produces better image quality compared with the classical BTC approaches. The detail explanation and comparison between EDBTC and BTC-based image compression can be found at [3] and [4]. BTC and EDBTC have the same characteristic in which the bitmap image and the two extreme values are produced at the end of the encoding stage. In BTC scheme, the two quantizers and its image bitmap are produced by computing the first moment, second moment, and variance value causing a high computational burden. Suppose a color image of size $M \times N$ is partitioned into multiple nonoverlapping image blocks of size $m \times n$. Let $f(x, y) = \{f_R(x, y), f_G(x, y), f_B(x, y)\}$ be an image block, where $x = 1, 2, \ldots, m$ and $y = 1, 2, \ldots, n$.

For each image block, the EDBTC produces a single bitmap image $bm(x, y)$ and two extreme (color) quantizers (q_{min} and q_{max}). The bitmap image size is identical to that of the original image size. EDBTC employs the error kernel to generate the representative bitmap image. Fig. 1 shows the error diffusion kernels for Floyd–Steinberg, Stucki, Sierra, Burkers, Jarvis, and Stevenson. Different error kernels yield different bit/halftoning patterns. The EDBTC exploits the dithering property of the error diffusion to overcome the false contour problem normally occurred in BTC compression. Moreover, the blocking effect can also be eased by its error Bkernel since the quantization error on one side of the boundary can be compensated by the other side of the boundary. The correlation on both sides of a boundary between any pair of resulting image blocks can be maintained. The EDBTC bitmap image can be obtained by performing thresholding of the interband average value with the error kernel. In a block-based process, the raster-scan path (from left to right and top to bottom) is applied to process each pixel in a given image. Suppose that $f(x, y)$ and $f^-(x, y)$ denote the original and interband average value, respectively. The inter-band average value can be computed as

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

The $f_R(x, y)$, $f_G(x, y)$, and $f_B(x, y)$ denote the image pixels in the red, green, and blue (RGB) color channels, respectively. The inter-band average image can be viewed as the grayscale version of a color image.

The EDBTC performs the thresholding operation by incorporating the error kernel. We first need to compute the minimum, maximum, and mean value of the inter-band average pixels as

$$x_{min} = \min_{\forall x, y} \bar{f}(x, y)$$
$$x_{max} = \min_{\forall x, y} \bar{f}(x, y)$$
$$\bar{x} = \sum_{x=1}^{m} \sum_{y=1}^{n} \bar{f}(x, y).$$

The bitmap image h(x, y) is generated using the following rule:

$$h(x, y) = \begin{cases} 1, & \text{if } \bar{f}(x, y) \geq \bar{x} \\ 0, & \text{if } \bar{f}(x, y) < \bar{x}. \end{cases}$$

The intermediate value o(x, y) is also generated at the same time with the bitmap image generation. The value o(x, y) can be computed as

$$o(x, y) = \begin{cases} x_{max}, & \text{if } h(x, y) = 1 \\ x_{min}, & \text{if } h(x, y) = 0. \end{cases}$$

The residual quantization error of EDBTC can be computed as

$$e(x, y) = \bar{f}(x, y) - o(x, y).$$

The EDBTC thresholding process is performed in a consecutive way. One pixel is only processed once, and the residual quantization error is diffused and accumulated into the neighboring unprocessed pixels. The value $\bar{f}(x, y)$ of unprocessed yet pixel is updated using the following strategy:

$$\bar{f}(x, y) = \bar{f}(x, y) + e(x, y) * \epsilon$$

where is the error kernel to diffuse the quantization residual into its neighboring pixels which have not yet been processed in the EDBTC thresholding. The symbol $*$ denotes the convolution operation. Several error kernels can be used to perform the diffusion operation, such as Jarvis error kernel [63], Burkers [65], Floyd–Steinberg [61], Sierra [65], Stucki [62], and Stevenson [64], as shown in Fig. 1. The reason of choosing the extreme values to represent an image block is to generate a dithered result (bit pattern illusion) to reduce the annoying blocking effect or false contour inherently existing in BTC images. Notably, the error at the boundary of an image blockshould be diffused to its neighboring blocks, thus the blocking effect can be significantly eased in the EDBTC reconstructed image. The two extreme quantizers consist of RGB color information obtained by searching the minimum and maximum of value in an image block for each RGB color space. Two EDBTC color quantizers are computed by looking for the minimum and maximum of all image pixels in each image block as

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

Fig. 2 shows the schematic diagram of the EDBTC image compression system. The EDBTC is not only able to compress an image but also able to index an image in a CBIR system.
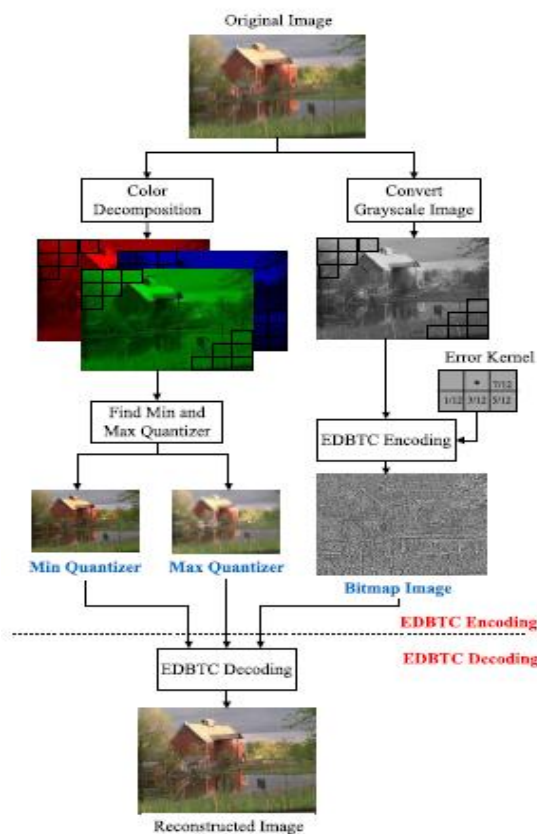

Fig2. *Schematic diagram of EDBTC processing for color image.*

At the end of the EDBTC encoding process, two color quantizers and a bitmap image are sent to the decoder via a transmission channel. The decoder simply replaces the bitmap image which has value 1 with the maximum quantizer, while the value 0 is substituted with the minimum quantizer. There is no computation needed in the decoder side, making it very attractive in the real-time application. Fig. 3 shows an image quality comparison of the EDBTC reconstructed image under Floyd–Steinberg, Stucki, and Stevenson error kernels over various image block sizes as 4×4, 8×8, 16×16, and 32×32. Compared with the BTC scheme, the EDBTC overcomes the blocking effect and false contour artifacts producing in the EDBTC image reconstructed.

Color Histogram Feature

The CHF is derived from the two EDBTC color quantizers, while BHF is computed from EDBTC bitmap image. In this paper, the CHFmin and CHFmax are developed from the color minimum and maximum quantizers, respectively. The CHFmin and CHFmax capture color information from a given image. These features represent the combination of pixel brightness and color distribution in an image. The CHFmin and CHF max features can be computed using

$$\mathrm{CHF}_{\min}(k) = \Pr\left\{\tilde{i}_{\min}(i, j) = k \,\middle|\, i = 1, 2, \ldots, \frac{M}{m}; \right.$$

$$\left. j = 1, 2, \ldots, \frac{N}{n} \right\}$$

$$\mathrm{CHF}_{\max}(k) = \Pr\left\{\tilde{i}_{\max}(i, j) = k \,\middle|\, i = 1, 2, \ldots, \frac{M}{m}; \right.$$

$$\left. j = 1, 2, \ldots, \frac{N}{n} \right\}.$$

In a nutshell, the CHFmin and CHFmax are the VQ-indexed histogram from the color minimum and maximum quantizers, respectively. It only calculates the occurrence of certain color codewords appeared in an image. Fig. 5 shows the CHF computation of the proposed EDBTC image retrieval system.

Bit Pattern Histogram Feature (BHF)

Another feature generated from a VQ-indexed EDBTC data stream is the BHF. This feature captures the visual pattern, edge, and textural information in an image. The BHF can be obtained by tabulating the occurrence of a specific bit pattern codebook in an image. The BHF can be generated using

$$\text{BHF}(k) = \Pr\left\{\bar{b}(i, j) = k \mid i = 1, 2, \ldots, \frac{M}{m}; \quad j = 1, 2, \ldots, \frac{N}{n}\right\}.$$

Fig. 6 shows the flowchart of the BHF computation. The BHF can be viewed as the histogram of the indexed bit pattern of the
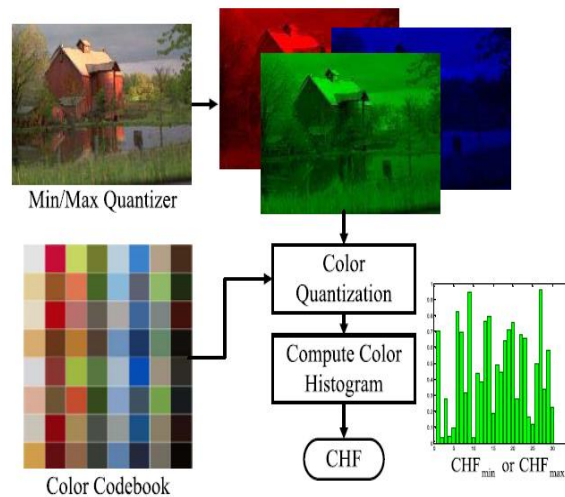

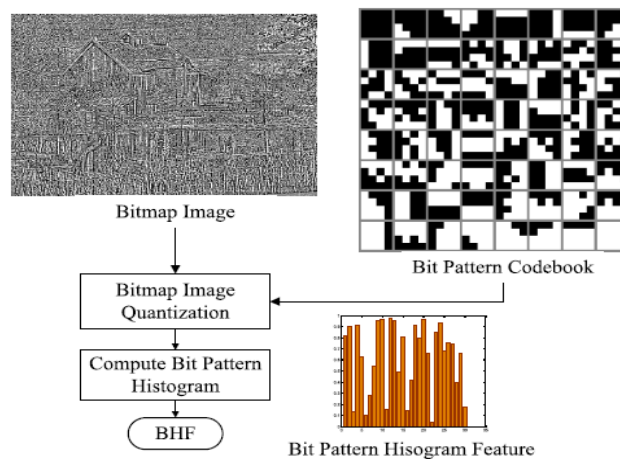**Fig5.** *Illustration of CHF computation.*

Fig6. *Illustration of BHF computation.*

EDBTC bitmap image. The color and bit pattern features can be used individually or together based on the user's preference.

Image Retrieval with EDBTC Feature

The similarity distance computation is needed to measure the similarity degree between two images. The distance plays the most important role in the CBIR system since the retrieval result is very sensitive with the chosen distance metric. The image matching between two images can be performed by calculating the distance between the query image given by a user against the target images in the database based on their corresponding features (CHF and BHF). After the similarity distance computation, the system returns a set of retrieved image ordered in ascending manner based on their similarity distance scores. The similarity distance between the two images, namely, query and target images, can be formally

defined as

$$\delta(\text{query, target}) = \alpha_1 \sum_{k=1}^{N_c} \frac{|\text{CHF}_{min}^{query}(k) - \text{CHF}_{min}^{target}(k)|}{\text{CHF}_{min}^{query}(k) + \text{CHF}_{min}^{target}(k) + \varepsilon}$$

$$+ \alpha_2 \sum_{k=1}^{N_c} \frac{|\text{CHF}_{max}^{query}(k) - \text{CHF}_{max}^{target}(k)|}{\text{CHF}_{max}^{query}(k) + \text{CHF}_{max}^{target}(k) + \varepsilon}$$

$$+ \alpha_3 \sum_{k=1}^{N_b} \frac{|\text{BHF}^{query}(k) - \text{BHF}^{target}(k)|}{\text{BHF}^{query}(k) + \text{BHF}^{target}(k) + \varepsilon}$$

where α_1, α_2, and α_3 are the similarity weighting constants representing the percentage contribution of the CHF and BHF in the proposed image retrieval process. The value 1 means that the color or bit pattern feature is catered in the similarity distance, while the value 0 meaning that the color or bit pattern feature is disabled in the distance computation. A small number ε is added into denominator to avoid the mathematic division error. The CHFquery and BHFquery denote the color and bit pattern feature descriptors of the query image, respectively, while the symbols CHFtarget and BHFtarget represent the image descriptors of the target image in database.

Experimental Results

In this section, extensive experiment results are reported to demonstrate the effectiveness of the proposed EDBTC image indexing method. Several image databases consisting of the natural and textural images are utilized in this experiment to have an in-depth investigation of the successfulness of the proposed CBIR system. The proposed image retrieval system extracts the image features from all images in the database using the proposed CHF and BHF EDBTC features. The similarity between the query and target images is measured based on the similarity distance score from their descriptors. A set of retrieved images is returned by the system in ascending order based on the similarity distance values. In this experiment, the retrieval accuracy is measured using the average precision, average recall, or ARR value over all query images. The higher average precision rate and ARR value indicate that the system is able to retrieve a set of returned image which has more similar appearance with the query image.

4. CONCLUSION

New method is proposed in this paper for color image indexing by exploiting the simplicity of the EDBTC method. A feature descriptor obtained from a color image is constructed from the EDBTC encoded data (two representative quantizers and its bitmap image) by incorporating the VQ. The CHF effectively represents the color distribution within an image, while the BHF characterizes the image edge and texture. The experimental results demonstrate that the proposed method is not only superior to the former BTC-based image indexing schemes but also to the former existing methods in the literature related to the CBIR. To achieve a higher retrieval accuracy, another feature can be added into the EDBTC indexing scheme with the other color spaces such as YCbCr, hue–saturation–intensity, and lab. An extension of the EDBTC image retrieval system can be brought to index video by considering the video as a sequence of images. This strategy shall consider the temporal information of the video sequence to meet the user requirement in the CBIR context.

REFERENCES

[1] E. J. Delp and O. R. Mitchell, "Image compression using block truncation coding," IEEE Trans. Commun., vol. 27, no. 9, pp. 1335–1342, Sep. 1979.

[2] Y.-G. Wu and S.-C. Tai, "An efficient BTC image compression technique," IEEE Trans. Consum. Electron., vol. 44, no. 2, pp. 317–325, May 1998.

[3] 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.

[4] J.-M. Guo, "Improved block truncation coding using modified error diffusion," Electron. Lett., vol. 44, no. 7, Mar. 2008, pp. 462–464.

[5] Y.-F. Liu, J.-M. Guo, and J.-D. Lee, "Inverse halftoning based on the Bayesian theorem," IEEE Trans. Image Process., vol. 20, no. 4, pp. 1077–1084, Apr. 2011.

[6] J.-M. Guo and Y.-F. Liu, "High capacity data hiding for error-diffused block truncation coding," IEEE Trans. Image Process., vol. 21, no. 12, pp. 4808–4818, Dec. 2012.

[7] J.-M. Guo and Y.-F. Liu, "Halftone-image security improving using overall minimal-error searching," IEEE Trans. Image Process., vol. 20, no. 10, pp. 2800–2812, Oct. 2011.

[8] J.-M. Guo, S.-C. Pei, and H. Lee, "Watermarking in halftone images with parity-matched error diffusion," Signal Process., vol. 91, no. 1, pp. 126–135, 2011.

[9] S.-C. Pei and J.-M. Guo, "Hybrid pixel-based data hiding and blockbased watermarking for error-diffused halftone images," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 867–884, Aug. 2003.