This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

Content-Based Image Retrieval Using Error Diffusion Block Truncation Coding Features

Jing-Ming Guo, *Senior Member, IEEE,* Heri Prasetyo, and Jen-Ho Chen

Department of Electrical Engineering
National Taiwan University of Science and Technology
Taipei, Taiwan

E-mail: jmguo@seed.net.tw, heri_inf_its_02@yahoo.co.id, noisee@ms42.hinet.net

ABSTRACT

This paper presents a new approach to index color images using the features extracted from the 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. The distance computed from CHF and BHF can be utilized to measure the similarity between two images. As documented in experimental result, the proposed indexing method outperforms the former BTC-based image indexing and the other existing image retrieval schemes with natural and textural datasets. Thus, the proposed EDBTC is not only examined with good capability for image compression, but it also offers an effective way to index images for the content-based image retrieval (CBIR) system.

Index Terms: Content-Based Image Retrieval, image indexing, Error Diffusion Block Truncation Coding, Vector Quantization.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

I. INTRODUCTION

Many former schemes have been developed to improve the retrieval accuracy in the CBIR system. One type of them is to employ image features derived from the compressed data stream [11-14, 21-23, 60]. As opposite to the classical approach which extracts an image descriptor from the original image, this retrieval scheme directly generates image features from the compressed stream without firstly performing the decoding process. This type of retrieval aims to reduce the time computation for feature extraction/generation since most of multimedia images are already converted to compressed domain before they are recorded in any storage devices. In [11-14, 60], 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, texture similarity, etc.

The concept of the BTC [1] is to look for a simple set of representative vectors to replace the original images. Specifically, the 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 sub-images 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. The traditional BTC method does not improve the image quality or compression ratio compared to JPEG or JPEG 2000. However, the BTC schemes achieves much lower computational complexity compared to that of these

techniques. Some attempts have been addressed to improve the BTC reconstructed image quality, compression ratio, and also to reduce the time computation [2].

Even though the BTC scheme needs low computational complexity, it often suffers from blocking effect and false contour problems, making it less satisfactory for human perception. The halftoning-based BTC, namely Error Diffussion Block Truncation Coding (EDBTC) [3-4], is proposed to overcome the two above disadvantages of the BTC. Similar to the BTC scheme, EDBTC looks for a new representation (i. e. two quantizer and bitmap image) for reducing the storage requirement. The EDBTC bitmap image is constructed by considering the quantized error which diffuses to the nearby pixels to compensate the overall brightness, and thus effectively remove the annoying blocking effect and false contour, while maintaining the low computational complexity. The low-pass nature of Human Visual System (HVS) is employed in [3, 4] to access the reconstructed image quality, in which the continuous image and its halftone version are perceived similarly by human vision when these two images viewed from a distance. The EDBTC method divides a given image into multiple non-overlapped image blocks, and each block is processed independently to obtain two extreme quantizers. This unique feature of independent processing enables the parallelism scenario. In bitmap image generation step, the pixel values in each block are thresholded by a fixed average value in the block with employing error kernel to diffuse the quantization error to the neighboring pixels during the encoding stage.

Some applications have been proposed in literature triggered by the successfulness of EDBTC such as image watermarking [3, 8-9], inverse halftoning [5], data hiding [6], image security [7], halftone classification [10], etc. The EDBTC scheme performs well in those areas with the promising result as reported in [3-10]. Since the EDBCT provides better reconstructed image quality than that of the BTC scheme. In this research, the concept of the EDBTC compression is catered to the CBIR domain, in which the image feature descriptor is constructed from the EDBTC compressed data stream. In this scheme, the compressed data stream which is already stored in database is not necessary decoded to obtain the image feature descriptor. The descriptor is directly derived from EDBTC color quantizers and bitmap image in compressed domain by involving the Vector Quantization (VQ) for the indexing. The

similarity criterion between the query and target image is simply measured using the EDBTC feature descriptor. This new CBIR system with the EDBTC feature can also be extended for video indexing and searching, in which the video is viewed and processed as a sequence of images. The EDBTC feature descriptor can also be adopted as an additional feature for object tracking, background subtraction, image annotation, image classification, and segmentation, etc. The EDBTC feature offers a competitive performance compared to that of the LBP-based feature, and thus the EDBTC feature can substitute the LBP-based feature for image processing and computer vision application with even faster processing efficiency.

A new image retrieval system has been proposed for the color image [69]. Three feature descriptors, namely Structure Element Correlation (SEC), Gradient Value Correlation (GVC), and Gradient Direction Correlation (GDC) are utilized to measure the similarity between the query image and the target images in database. This indexing scheme provides a promising result in big database and outperforms the former existing approaches as reported in [69]. The method in [70] compresses a grayscale image by combining the effectiveness of fractal encoding, Discrete Cosine Transform (DCT), and standard deviation of an image block. An auxiliary encoding algorithm has also been proposed to improve the image quality and to reduce the blocking effect. As reported in [70], this new encoding system achieves a good coding gain as well as the promising image quality with very efficient computation. In [71], a new method for tamper detection and recovery is proposed by utilizing the DCT coefficient, fractal coding scheme, and macted-block technique. This new scheme yields a higher tampering detection rate, and achieves good restored image quality as demonstrated in [71]. The method in [72] combines the fractal image compression and wavelet transform to reduce the time computation in image encoding stage. This method produces a good image quality with a fast encoding speed as reported in [72]. The fast and efficient image coding with the no-search fractal coding strategies have been proposed in [73, 74]. Both methods employ the modified gray-level transform to improve the successful matching probability between the range and domain block in the fractal coding. Two gray-level transforms on quadtree partition are used in [73] to achieve a fast image coding and to improve the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

decoded image quality. The method [74] exploits a fitting plane method and a modified gray-level transform to speed up the encoding process. The fractal image coding presented in [74] accelerates the image encoding stage, reduces the compression ratio, and simultaneously improves the reconstructed image quality. A fast fractal coding is also proposed in [75] which utilizes the matching error threshold. This method firstly reduces the codebook capacity, and takes advantage of matching error threshold to shorten the encoding runtime. The method [75] can achieve a similar or better decoded image with the fast compression process compared to the conventional fractal encoding system with full search strategy.

The contributions can be summarized as follow: 1) extending the EDBTC image compression technique for the color image, 2) proposing two feature descriptors, namely Color Hitogram Feature (CHF) and Bit Pattern Histogram Feature (BHF), which can be directly derived from the EDBTC compressed data stream without performing decoding process, and 3) presenting a new low complexity joint content-based image retrieval system and color image compression by exploiting the superiority of EDBTC scheme. The rest of this paper is organized as follows. A brief introduction of EDBTC is provided in Section II. Section III presents the proposed EDBTC image retrieval including the image feature generation and accuracy computation. Extensive experimental results are reported at section IV. Finally, the conclusions are drawn at the end of this paper.

II. Error Diffusion Block Truncation Coding For Color Image

This section presents a review of the EDBTC with its extension to color image compression. The EDBTC compresses an image in an effectively 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 to the classical BTC approaches. The detail explanation and comparison between EDBTC and BTC-based image compression can be found at [3, 4].

BTC and EDBTC have the same characteristic in which the bitmap image and the two extreme

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

$$\left(\frac{1}{16}\right)\begin{bmatrix} & * & 7 \\ 3 & 5 & 1 \end{bmatrix}$$

(a)

$$\left(\frac{1}{42}\right)\begin{bmatrix} & & * & 8 & 4 \\ 2 & 4 & 8 & 4 & 2 \\ 1 & 2 & 4 & 2 & 1 \end{bmatrix}$$

(b)

$$\left(\frac{1}{32}\right)\begin{bmatrix} & & * & 5 & 3 \\ 2 & 4 & 5 & 4 & 2 \\ 0 & 2 & 3 & 2 & 0 \end{bmatrix}$$

(c)

$$\left(\frac{1}{32}\right)\begin{bmatrix} & & * & 8 & 4 \\ 2 & 4 & 8 & 4 & 2 \end{bmatrix}$$

(d)

$$\left(\frac{1}{48}\right)\begin{bmatrix} & & * & 7 & 5 \\ 3 & 5 & 7 & 5 & 3 \\ 1 & 3 & 5 & 3 & 1 \end{bmatrix}$$

(e)

$$\left(\frac{1}{200}\right)\begin{bmatrix} 12 & & 26 & & * & 30 & 32 & & 16 \\ & 12 & & 26 & & & 12 & & \\ 5 & & 12 & & 12 & & & & 5 \end{bmatrix}$$

(f)

Fig. 1. Error diffusion kernel: (a) Floyd-Steinberg [61], (b) Stucki [62], (c) Sierra [65], (d) Burkers [65], (e) Jarvis [63], and (f) Stevenson filter [64].

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 non-overlapping 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 $x = 1,2,\dots,m$ and $y = 1,2,\dots,n$. 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. Figure 1 shows the error diffusion kernels for Floyd-Steinberg, Stucki, Sierra, Burkers, Jarvis, and Stevenson. Different error kernels yield different bit pattern/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 kernel, 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 inter-band 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 $\bar{f}(x,y)$ denote the original and inter-band average value, respectively. The inter-band average value can be computed as:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

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

The $f_R(x,y)$, $f_G(x,y)$, and $f_B(x,y)$ denote the image pixels in the red, green, and blue 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 firstly need to compute the minimum, maximum, and mean value of the inter-band average pixels as follows:

$$x_{min} = \min_{\forall x,y} \bar{f}(x,y), \quad (2)$$

$$x_{max} = \max_{\forall x,y} \bar{f}(x,y), \quad (3)$$

$$\bar{x} = \sum_{x=1}^{m} \sum_{y=1}^{n} \bar{f}(x,y). \quad (4)$$

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

$$h(x,y) = \begin{cases} 1, & \text{if } \bar{f}(x,y) \geq \bar{x}; \\ 0, & \text{if } \bar{f}(x,y) < \bar{x}. \end{cases} \quad (5)$$

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

$$o(x,y) = \begin{cases} x_{max}, & \text{if } h(x,y) = 1; \\ x_{min}, & \text{if } h(x,y) = 0. \end{cases} \quad (6)$$

The residual quantization error of EDBTC can be computed as:

$$e(x,y) = \bar{f}(x,y) - o(x,y), \quad (7)$$

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 un-processed pixels. The value $\bar{f}(x,y)$ of un-processed yet pixel is updated using the following strategy:

$$\bar{f}(x,y) = \bar{f}(x,y) + e(x,y) * \epsilon, \quad (8)$$

where ϵ 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], Stevenson [64], etc, as illustrated in Fig.1. The reason of choosing the extreme values to represent an image block is to generate a dithered result

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

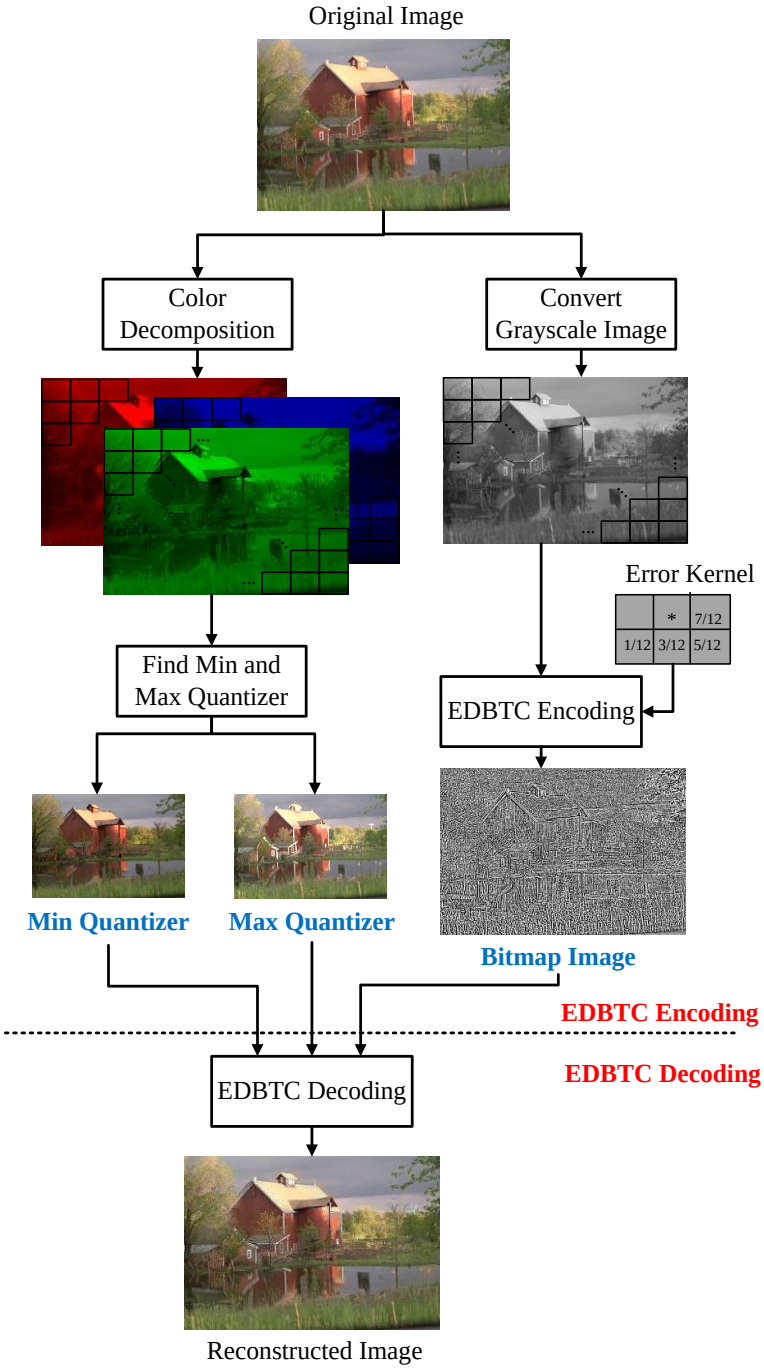


Fig. 2. Schematic diagram of EDBTC processing for color image.

(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 block should 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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

Fig. 3. Reconstruction image quality comparison of BTC (first column) and EDBTC (second column) with Floyd-Steinberg error kernel over image block sizes (from first to fourth row) 4×4, 8×8, 16×16, and 32×32, respectively.

computed by looking for the minimum and maximum of all image pixels in each image block as:

$$q_{min}(i,j) = \{\min_{\forall x,y} f_R(x,y), \min_{\forall x,y} f_G(x,y), \min_{\forall x,y} f_B(x,y)\}, \quad (9)$$

$$q_{max}(i,j) = \{\max_{\forall x,y} f_R(x,y), \max_{\forall x,y} f_G(x,y), \max_{\forall x,y} f_B(x,y)\}. \quad (10)$$

Figure 2 illustrates 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.

At the end of the EDBTC encoding process, two color quantizers and a bitmap image are sent to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

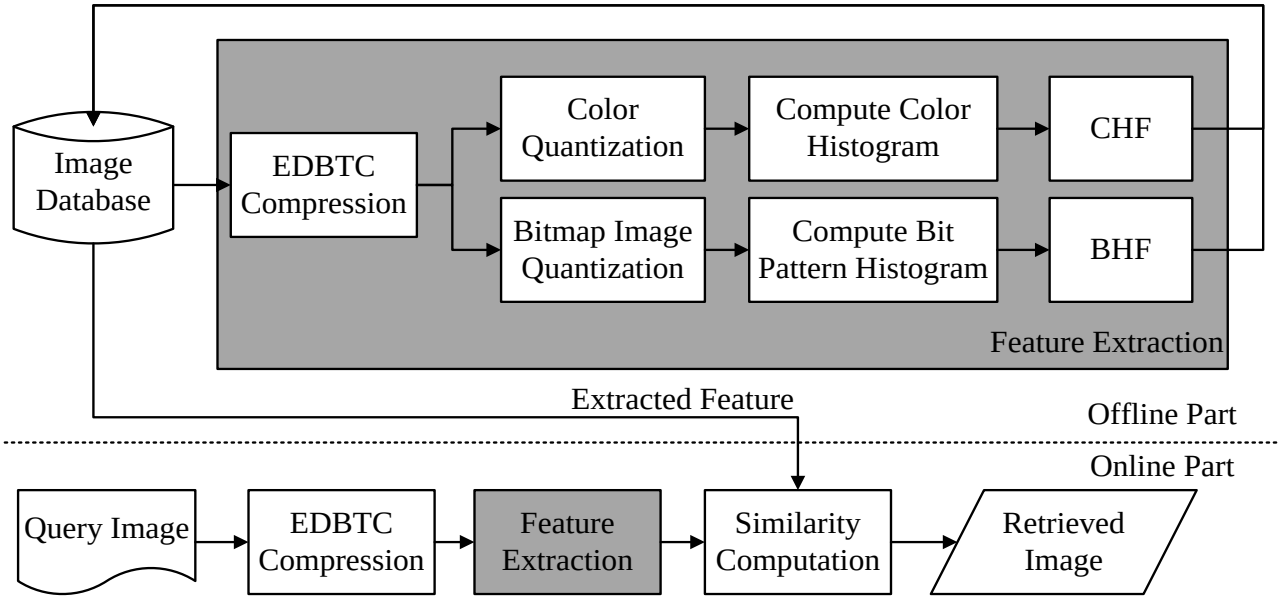


Fig. 4. Schematic diagram of the proposed image retrieval framework.

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. Figure 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 countur artifacts producing in the EDBTC image reconstructed.

III. EDBTC Image Indexing

The proposed EDBTC image indexing scheme is presented in this section. Figure 4 illustrates the block diagram of the proposed image retrieval system. First, the image is encoded using EDBTC to obtain the two representative quantizers (quantization levels) and the bitmap image. An image feature descriptor representing the image content is then constructed from the two quantizers and bitmap image. There are two features employed in the proposed method to characterize the image content, namely Color Histogram Feature (CHF) and Bit Histogram Feature (BHF). The CHF is derived from the two color

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

quantizers (maximum and minimum), and the BHF is constructed from the bitmap image.

A. Vector Quantization

Vector quantization (VQ) compresses an image in lossy mode based on block coding principle. It is a fixed-to-fixed length algorithm. The VQ finds a codebook by iteratively partitioning a given source vector with its known statistical properties to produce the codevectors with the smallest average distortion when a distortion measurement is *a priori* given. Let $C = \{c_1, c_2, \dots, c_{N_c}\}$ be the color codebook generated using VQ consisting N_c codewords. The VQ needs many images involved as the training set. The vector c_k contains RGB color (or grayscale) information which is identical to the two EDBTC quantizers. Except for generating the color codebook, it is also required to construct the bit pattern codebook $B = \{B_1, B_2, \dots, B_{N_b}\}$ consisting of N_b binary codewords. For generating the bit pattern codebook, bitmap images are demanded as input/training set in the VQ process. The bi-level value in bitmap image is treated as a non-integer value which later is fed into VQ. At the end of the training stage, the hard binarization is performed for all codevectors as the final result using the soft centroid principle [24].

Given the color codebook $C = \{c_1, c_2, \dots, c_{N_c}\}$, the VQ indexes the EDBTC minimum and maximum quantizers using the following formula:

$$\tilde{i}_{min}(i,j) = \underset{k=1,2,\dots,N_c}{\operatorname{argmin}} \left\| q_{min}(i,j), c_k^{min} \right\|_2^2, \tag{11}$$

$$\tilde{i}_{max}(i,j) = \underset{k=1,2,\dots,N_c}{\operatorname{argmin}} \left\| q_{max}(i,j), c_k^{max} \right\|_2^2 . \tag{12}$$

for all $i = 1,2,\dots,\frac{M}{m}$, and $j = 1,2,\dots,\frac{N}{n}$. The symbol i and j denote the index of image block. The VQ reduces the bit required for storing the color minimum and maximum quantizers by performing the indexing process as indicated at (11) and (12). The entropy coding with lossless or lossy approach can be applied to further reduce the bits required before the transmission process.

At the same time, the VQ method indexes a EDBTC bitmap image $bm(i,j) = \{h(x,y) | x = 1,2,\dots,m; y = 1,2,\dots,n\}$ with the bit pattern codebook $B = \{B_1, B_2, \dots, B_{N_b}\}$ using the following

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

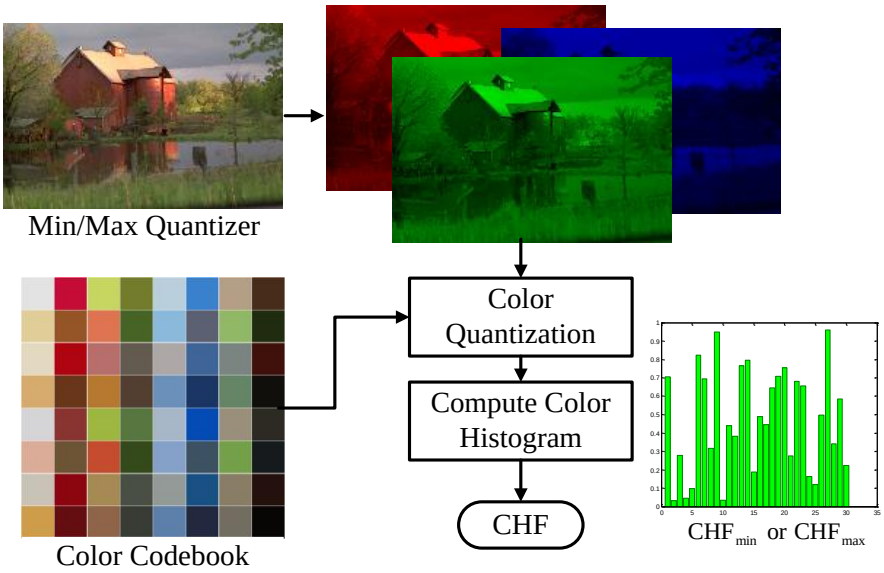


Fig. 5. Illustration of CHF computation.

procedure:

$$\tilde{b}(i,j) = \underset{k=1,2,\ldots,N_b}{\operatorname{argmin}} \delta_H\{bm(i,j), B_k\}, \tag{13}$$

for all $i = 1,2,\ldots,\frac{M}{m}$, and $j = 1,2,\ldots,\frac{N}{n}$. The symbol $\delta_H\{\cdot,\cdot\}$ denotes the Hamming distance between two binary patterns (vectors). The VQ performs indexing for both color and bitmap image which may further reduce the bit required in the ODBTC compression scheme. Based on our observation, the VQ-indexed information is not only useful for image compression, but it also can index an image in effectively and efficiently for a CBIR system.

B. Color Histogram Feature (CHF)

The CHF is derived from the two EDBTC color quantizers, while BHF is computed from EDBTC bitmap image. In this study, the CHF_{min} and CHF_{max} are developed from the color minimum and maximum quantizer, respectively. The CHF_{min} and CHF_{max} capture color information from a given image. These features represent the combination of pixel brightness and color distribution in an image. The CHF_{min} and CHF_{max} features can be computed using the following equations:

$$CHF_{min}(k) = \Pr\left\{\tilde{i}_{min}(i,j) = k \middle| i = 1,2,\ldots,\frac{M}{m}; j = 1,2,\ldots,\frac{N}{n}\right\}, \tag{14}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

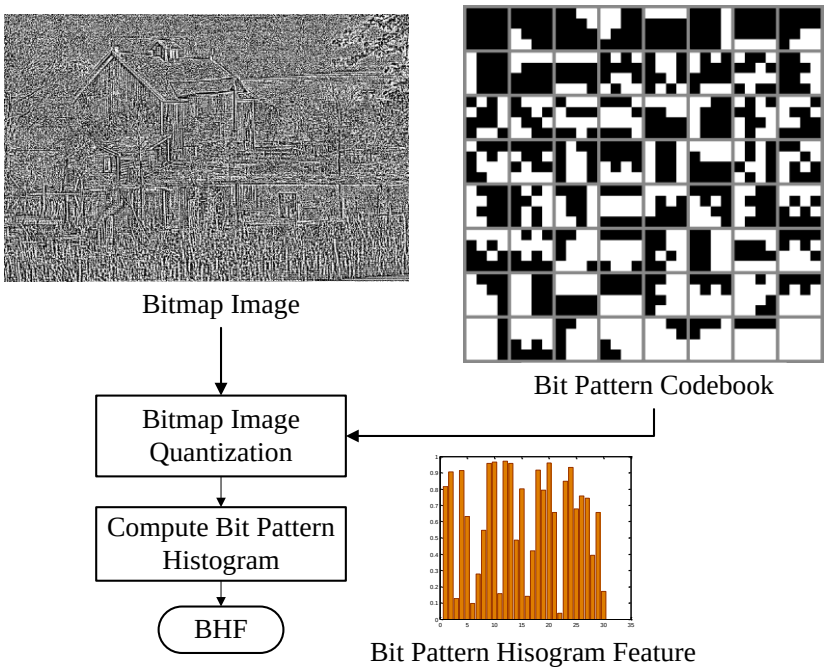


Fig. 6. Illustration of BHF computation.

$$\mathrm{CHF}_{\max}(k) = \Pr\left\{\tilde{i}_{\max}(i,j) = k \middle| i = 1,2,\dots,\frac{M}{m}; j = 1,2,\dots,\frac{N}{n}\right\}, \quad (15)$$

In a nutshell, the $\mathrm{CHF}_{\min}$ and $\mathrm{CHF}_{\max}$ 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. Figure 5 illustrates the CHF computation of the proposed EDBTC image retrieval system.

C. Bit Pattern Histogram Feature (BHF)

Another feature generated from a VQ-indexed EDBTC datastream 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 the following equation:

$$\mathrm{BHF}(k) = \Pr\left\{\tilde{b}(i,j) = k \middle| i = 1,2,\dots,\frac{M}{m}; j = 1,2,\dots,\frac{N}{n}\right\}, \quad (16)$$

Figure 6 shows the flowchart of the BHF computation. The BHF can be viewed as the histogram of the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

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

D. 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 follows:

$$\delta(\text{query},\text{target}) = \alpha_1 \sum_{k=1}^{N_c} \frac{|\text{CHF}_{\text{min}}^{\text{query}}(k) - \text{CHF}_{\text{min}}^{\text{target}}(k)|}{\text{CHF}_{\text{min}}^{\text{query}}(k) + \text{CHF}_{\text{min}}^{\text{target}}(k) + \varepsilon} + \alpha_2 \sum_{k=1}^{N_c} \frac{|\text{CHF}_{\text{max}}^{\text{query}}(k) - \text{CHF}_{\text{max}}^{\text{target}}(k)|}{\text{CHF}_{\text{max}}^{\text{query}}(k) + \text{CHF}_{\text{max}}^{\text{target}}(k) + \varepsilon} + \alpha_3 \sum_{k=1}^{N_b} \frac{|\text{BHF}^{\text{query}}(k) - \text{BHF}^{\text{target}}(k)|}{\text{BHF}^{\text{query}}(k) + \text{BHF}^{\text{target}}(k) + \varepsilon}, \quad (17)$$

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 $\text{CHF}^{\text{query}}$ and $\text{BHF}^{\text{query}}$ denote the color and bit pattern feature descriptors of the query image, respectively, while the symbols $\text{CHF}^{\text{target}}$ and $\text{BHF}^{\text{target}}$ represent the image descriptors of the target image in database.

E. Performance Measurement

The successfulness of the proposed EDBTC retrieval system is measured with the precision, recall, and Average Retrieval Rate (ARR) value. These values indicate the percentage of relevant image returned

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

by a CBIR system with a specific number of retrieved images L. The precision ($P(q)$) and recall ($R(q)$) values are defined as:

$$P(q) = \frac{n_q}{L}, \tag{18}$$

$$R(q) = \frac{n_q}{N_q}, \tag{19}$$

where n_q and N_q denote the number of relevant images against a query image q, and the number of all relevant images against a query image q in database. A higher value in precision and recall exhibits the better retrieved result.

The other metric employed to measure the retrieval performance is the ARR value which can be formally defined as:

$$ARR = \frac{1}{|DB|}\sum_{i=1}^{|DB|} R(I_i, n)\Big]_{n\geq 16}, \tag{20}$$

$$R(I_i, n) = \frac{n_q}{N_q}, \tag{21}$$

where the $|DB|$ denotes the total number of images in the database. Similar to the precision and recall rates, a greater ARR value indicates that the image retrieval system performs well/better in retrieving a set of similar images. Normally, a set of returned image is more preferable to a user.

IV. 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 image are utilized in this experiment to have an in-depth investigate 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 image and target image 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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

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.

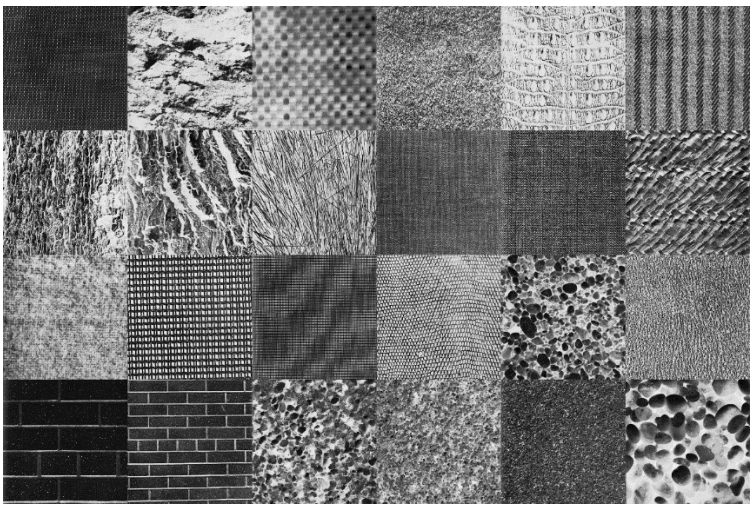
A. Database and Codebook Generation

The natural and textural images are incorporated to investigate the successfulness of the proposed method. The first experiment utilizes the natural color images consisting two databases, i.e., Corel 1000 and Corel 10,000 databases. The experiments were carried out with the commercially available Corel Photo Collections [25], consisting of 1000 color images of size 384×256. All images in database are grouped into ten classes, and each class consists of 100 images with different semantic categories such as people, beach, building, bus, dinosaur, elephant, flower, horse, mountain, and food. The images in the same class or semantic category are regarded as similar images. For the second experiment, the proposed method and the former schemes are fairly investigated using $10{,}000$ natural images from the Corel dataset as in [30, 31]. The database consists of 100 categories with different semantic names as beach, car, fish, door sunset, etc. Each category contains 100 images. The performance accuracy of the proposed method and the former schemes for Corel 1000 and Corel 10,000 is compared based on their average precision rate.

For the other experiment, a comparison is made between the proposed EDBTC indexing with the former schemes under the textural image. Herein, both of the grayscale and color textural image databases, namely DB1 and DB2, as similar in [41] are involved. The DB1 and DB2 consist of 116 and 40 different texture classes, respectively. The DB1 contains 109 grayscale images collected from the Brodatz texture photographic album [47] and 7 grayscale images from USC database [48]. On the other hand, the DB2 has 40 different color texture images assembled from the Vistex texture dataset in [49]. Each image is of size 512×512 which is further divided into 16 non-overlapping images of size 128×128. Thus, the DB1 and DB2 consist of $1856\ (116 \times 16)$ and $640\ (40 \times 16)$ images, respectively. The main purpose of using the DB1 and DB2 is to make a fair comparison between the proposed EDBTC image retrieval with the former methods. Additional databases, namely DB3 and DB4

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

(a)



(b)

(c)

Fig. 7. Image sample from the test database: (a) Corel 1000, (b) Brodatz, and (c)Vistex database.

of sizes 416 and 640, respectively, are also employed to further investigate the superiority of the proposed indexing method compared with the fractal-based image retrieval. Figure 7 shows some image

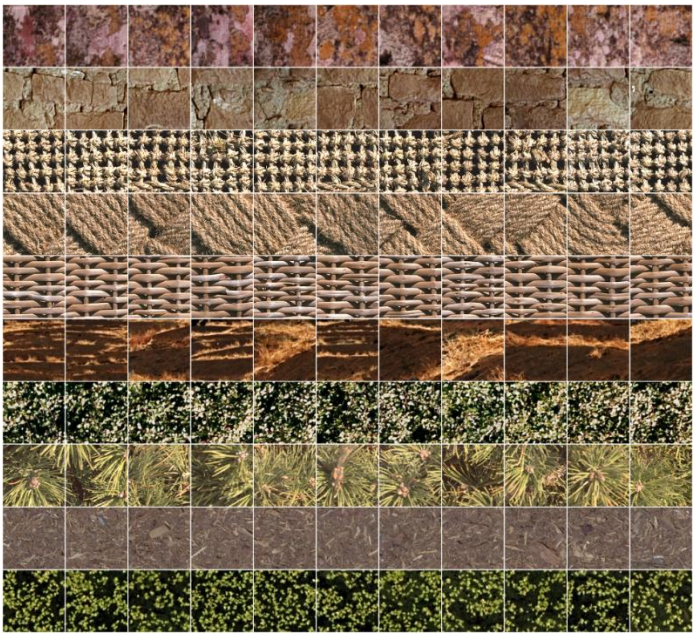
samples from several databases such as Corel 1000, Brodatz, and Vistex image databases. In the textural image database, the successfulness of the proposed method is measured in terms of the ARR value in which a higher ARR value indicates the better retrieval result. The retrieval performance of the proposed method is investigated not only for natural image, but also for the textural image as well as in the color and grayscale modes.

We firstly perform color and bit pattern quantization by means of VQ to obtain corresponding codebooks for the CHF and BHF. The color codebook is simply generated using the common LBG-VQ, while the bit pattern codebook is computed using the binary LBG-VQ with soft centroid strategy [24]. In the bit pattern codebook generation, the common LBG-VQ computes and searches the representative codebook from the training data in binary form by treating the binary values as real values. The image block in binary values is firstly converted into real values before bit pattern codebook generation. At the codebook generation, all components of a codevector may have intermediate values between zero (black pixel) and one (white pixel), and not necessary in binary values. It can be seen that the training vectors are firstly initialized at the corners of the hypercube space, and then the codevectors are later updated and calculated by shifting inside the cube during the training process. At the end of the training stage, the hard binarization is performed for all codevectors to obtain the final result, i.e., trained bit pattern codebook.

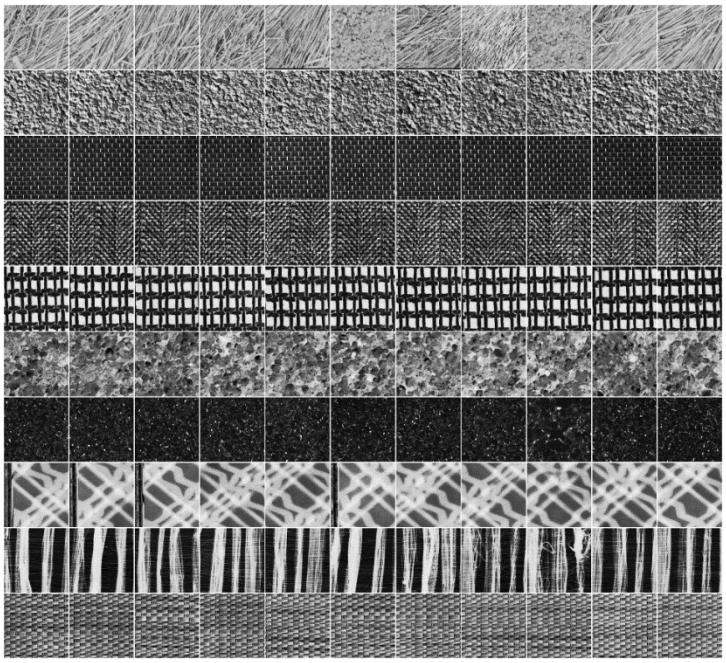
Five images of each class are randomly chosen from the Corel 1000 database as the training set to generate color and bit pattern codebook by applying the color LBG-VQ and soft centroid binary VQ training, respectively. The color and bit pattern codebook are generated only for once, and then used for all image databases such as Corel 1000, Corel 10,000, Vistex, etc. In other words, these codebooks can be regarded as predetermined data for image retrieval.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

(a)



(b)



(c)

Fig. 8. Examples of image retrieval results using (a) Corel 1000, (b) Vistex, and (c) Brodatz databases.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

B. Effectiveness of the CCF and BHF

Some experiments are additionally conducted to further investigate the effectiveness of the proposed feature descriptor. First, EDBTC encodes all images in the database over various image block sizes, i.e., 4×4, 8×8, 16×16, and 32×32, to obtain an image feature descriptor. Two color quantizers and the bitmap image are produced at the end of the EDBTC encoding. The color quantizers are fed into the color VQ module, while the bit pattern is for the bit pattern VQ processing. The color quantization is performed by incorporating the trained codebook in which the CHF is sequentially produced after the VQ indexing process. The CHF is simply generated by computing the histogram of the VQ-indexed color. Simultaneously, the bit pattern indexing is also processed using the trained bit pattern codebook to obtain the BHF. The CHF and BHF are then stored in the database for similarity matching when a query image is turned against the targeted images in the database. In this experiment, the color and bit pattern codebook sizes are determined as $N_c = 64$ and $N_b = 256$ to yield a similar feature dimension as that of the former compared schemes.

Figure 8 shows some retrieved examples using the proposed image retrieval scheme under Corel 1000, Vistex, and Brodatz image databases. Among these, the query image is placed on the left hand side for each row, whereas the retrieved image set is shown subsequently from left to right based on their similarity distance ordered in descending manner. From these results, it can be seen that the proposed EDBTC retrieval system is very effective to index an image for CBIR application.

We further investigate the successfulness of the proposed EDBTC feature descriptor under Corel 1000 and DB1 databases. All images in database are turned as query image successively, and the retrieval accuracy is computed based on the average precision and recall value for the two image databases. In our retrieval strategy, a set of retrieved images is returned by the system and is ordered in descending manner based on the visual similarity score. The accuracy is observed under the number of the retrieved images $L = \{10, 20, \dots, 100\}$ and $L = \{4, 8, \dots, 16\}$ for Corel 1000 and DB1, respectively. The similarity constants are simply set at $\alpha_i = \{0,1\}$ for $i = 1, \dots, 3$, where the value 1 indicates that the corresponding feature is used for computing the similarity distance, and vice versa. For example, when

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

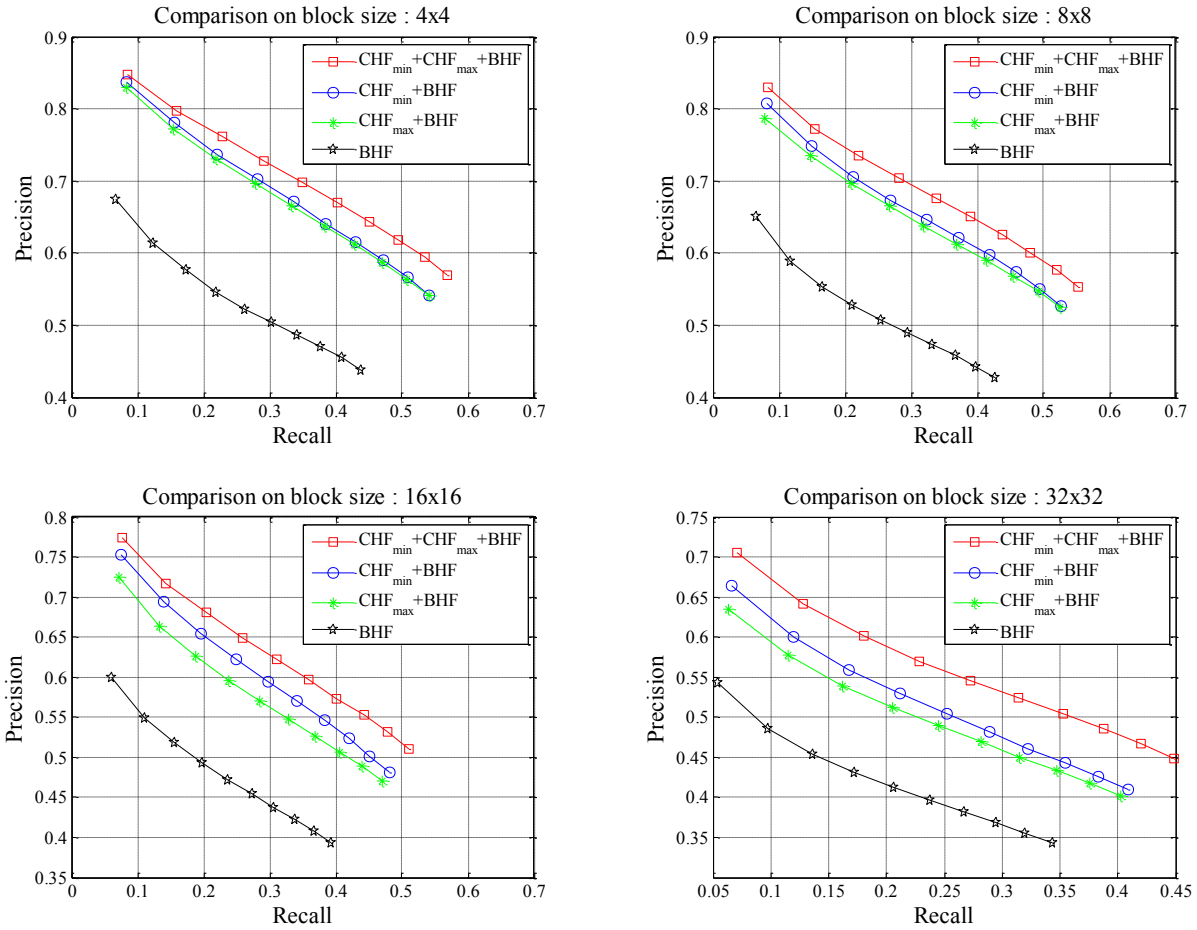


Fig. 9. Effectiveness of the proposed feature under Corel 1000 database.

BHF is employed, the similarity weighting constants are set as $\{\alpha_1 = 0, \alpha_2 = 0, \alpha_3 = 1\}$. In this experiment, the effectiveness of EDBTC feature descriptor is also investigated over several block sizes such as 4×4, 8×8, 16×16, and 32×32.

Figure 9 shows the average precision recall values of proposed EDBTC feature descriptor under Corel 1000 database. The image feature descriptor can be employed individually (such as only using $\mathrm{CHF}_{\mathrm{min}}$, $\mathrm{CHF}_{\mathrm{max}}$, or BHF) or combined among several features (such as $\mathrm{CHF}_{\mathrm{min}} + \mathrm{BHF}$, $\mathrm{CHF}_{\mathrm{min}} + \mathrm{CHF}_{\mathrm{max}}$, etc). For the Corel 1000 database, the combination of all features yields the best average precision and recall over all image block sizes. The same fact can also be found for DB1 as reported in Fig. 10. The usage of all features produces the highest average precision and recall rates over all image block sizes and all numbers of retrieved images, since all features can completely capture the color distribution as well as the edge and image content which are very important in the image retrieval system.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

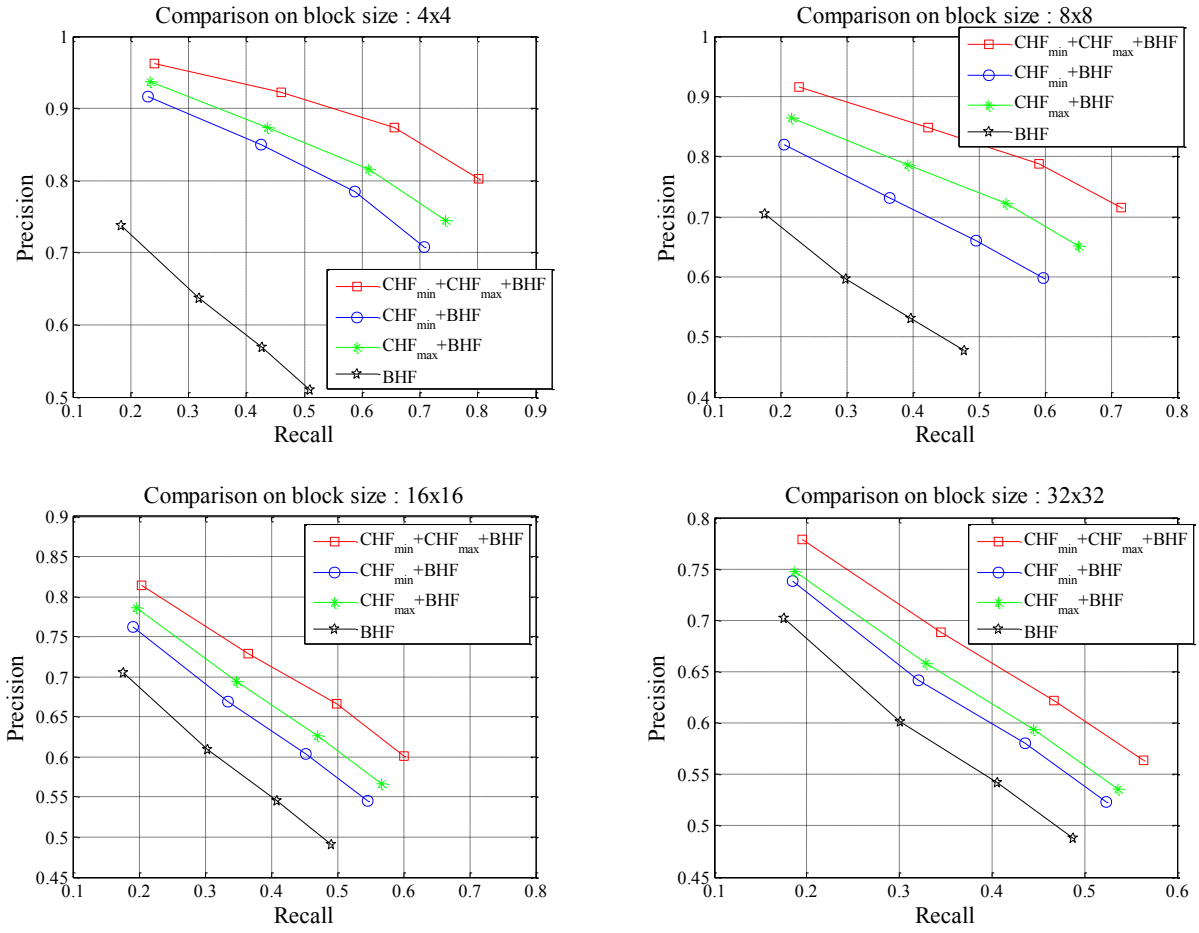


Fig. 10. Effectiveness of the proposed feature under Brodatz database.

An additional experiment were also carried out to further investigate the effect of the EDBTC error kernels on the proposed image retrieval system. Table I shows the average precision rates over different block sizes, when different EDBTC error kernels are employed in the feature extraction stage. From this result, we can see that the Stevenson error filter yields the best retrieval result compared with the other error kernels under 4×4 image block size. It is noteworthy that the differences among the average precision values for all kernels are not significant (the difference is only about 1%), meaning that the EDBTC is stable and the kernel choice leads to slight difference in the image retrieval performance.

C. Comparison with former existing methods

This experiment aims at making some comparisons among the proposed method and former existing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

TABLE I. EFFECT OF EDBTC KERNELS IN IMAGE RETRIEVAL.

Error Kernel	Block Size			
	4×4	8×8	16×16	32×32
Jarvis [63]	0.760	0.768	0.721	0.656
Burkers [65]	0.758	0.761	0.702	0.635
Floyd-Steinberg [61]	0.797	0.772	0.717	0.642
Sierra [65]	0.799	0.775	0.725	0.651
Stucki [62]	0.800	0.774	0.727	0.660
Stevenson [64]	0.802	0.769	0.722	0.656

methods under all image databases. For comparison purpose, the EDBTC image block size is determined at 4×4. For the Corel 1000 database, the color and bit pattern codebook are chosen as $N_c = 64$ and $N_b = 256$, respectively, as identical to that of the BTC-based image indexing schemes [11-14, 60] for a fair comparison. The color and bit pattern codebooks are set with $N_c = 64$ and $N_b = 64$ for Corel 10,000 database to further reduce the time computation for feature extraction and similarity matching. The proposed feature dimensionality is $N_c + N_c + N_b$, while the dimensionalities of the former BTC-based image indexing scheme are $N_c^2 + N_b$ [11], $3N_c + 3N_c + 3N_b$ [12], $2N_c^2$ [13], and $2N_c^2 + N_b$ [60]. The color and bit pattern in [12] are fixed at $N_c = N_b = 256$. In this experiment, the dimensionalities of the proposed image features are $64 + 64 + 256$ and $64 + 64 + 64$ for Corel 1000 and Corel 10,000 image databases, respectively. In the case of textural database, the size of the color and bit pattern codebooks are chosen as identical to the feature dimensionality indicated in Table IV. For Corel 1000 and 10,000 databases, the similarity weighting constant is set at $\alpha_i = 1$ for $i = 1, \ldots, 3$, indicating that all image features are employed.

In this experiment, all images in the Corel 1000 database are turned as query, and the accuracy performance is measured with the average precision rate among all query images with the number of retrieved images $L = 20$. For Corel 10,000 database, the experiments were conducted as similar to that of [31, 34], setting by randomly selecting 20 image categories out of 100 categories. For each category, 10 images are randomly drawn and turned as query images in which the average precision

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

TABLE II. COMPARISONS WITH FORMER SCHEMES UNDER COREL 1000 AND COREL 10,000 DATABASES.

Corel 1000		Corel 10,000	
Method	Average Precision	Method	Average Precision
Gahroudi [12]	0.396	EOAC [46]	0.210
Jhanwar [16]	0.526	CSD3 [44]	0.290
Huang [17]	0.532	MTH [43]	0.310
Chiang [19]	0.533	EHD [45]	0.350
Silakari [14]	0.560	MSD [42]	0.490
Saadatmand [26]	0.565	SED [30]	0.540
Moghaddam [27]	0.566	SEC+GVC+GDC [69]	0.560
Qiu [11]	0.595	ODII [60]	0.760
Z.M. Lu [21]	0.600	Proposed Method	**0.798**
Lu [18]	0.665		
Yu [13]	0.717		
Subrahmanyam [68]	0.725		
Lin [15]	0.727		
ElAlami [23]	0.739		
Subrahmanyam [28]	0.740		
Poursistani [22]	0.743		
ODII [60]	0.773		
Proposed Method	**0.797**		

rate is later computed for all query images. As similar to the experimental setting in [31, 34], the number of retrieved image is set at $L = 10$. Table II exhibits the comparisons among the proposed method and former image indexing schemes under Corel 1000 and Corel 10,000 databases. It can be seen that the proposed method outperforms the former existing CBIR methods under Corel 1000 and Corel 10,000 databases. With the Corel 1000 image database, the MSD [42], SED [30], CSD3 [44], SEC+GVC+GDC [69], and the proposed method yield retrieval rates 0.85, 0.97, and 0.81, 0.979, and 0.984, respectively, in which ten images of each category are turned as query image with the number of retrieved images, $L = 10$. The proposed method is slightly superior to that of the method [69] and the former existing methods under the same experimental setting. Yet, the proposed method outperforms all of the existing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

TABLE III. COMPARISONS WITH FORMER SCHEME UNDER DB1 AND DB2 DATABASES.

Method	Feature Dimension	ARR for DB1	ARR for DB2
GT with GGD&KLD [39]	$4 \times 6 \times 2$	17.19	76.57
DT-CWT [40]	$(3 \times 6 + 2) \times 2$	74.73	80.78
DT-RCWT [40]	$(3 \times 6 + 2) \times 2$	71.17	75.78
DT-CWT+DT-RCWT [40]	$2 \times (3 \times 6 + 2) \times 2$	77.75	82.34
LBP [35]	256	73.26	82.27
LTP [36]	2×256	79.96	82.38
GLBP [37]	$3 \times 4 \times 256$	75.21	84.74
LMEBP [41]	8×512	83.28	87.77
GLMEBP [41]	$3 \times 4 \times 512$	82.01	87.93
LDP [67]	4×59	79.91	87.27
LTrP [50]	13×59	85.3	90.02
GLTrP [50]	13×59	82.04	90.16
GLDP [67]	4×59	79.24	88.18
MCMCM+DBPSP [68]	$9 \times 7 \times 7 + 6$	-	86.17
ODII [60]	$N_{min} = N_{max} = 128, N_b = 128$	85.56	**93.23**
Proposed Method	256 $(N_{min} = 128, N_b = 128)$	84.27	90.47
Proposed Method	$256 (N_{max} = 128, N_b = 128)$	87.4	90.63
Proposed Method	$272 (N_{min} = N_{max} = 128, N_b = 16)$	87.51	89.95
Proposed Method	$384 (N_{min} = N_{max} = N_b = 128)$	**90.09**	92.55
Proposed Method	$192 (N_{min} = N_{max} = N_b = 64)$	80.74	91.77

schemes with the Corel 10,000 image database.

An additional experiment were conducted to compare the accuracy performance among the proposed method and former image retrieval schemes using Local Binary Patterns (LBP) feature and its variants such as Local Binary Pattern (LBP) [35], Local Tenary Pattern (LTP) [36], Local Tetra Pattern (LTrP) [50], Local Derivative Pattern (LDP) [67], Local Maximum Edge Binary Pattern (LMEBP) [41], Gabor LMEBP (GLMEBP) [41], Gabor LBP (GLBP) [37], etc. The experiments were investigated using the DB1 and DB2 databases by considering the ARR value to evaluate the image retrieval accuracy. All images in the databases are in turn considered as the query image. The LBP and its variants (which are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

TABLE IV. COMPARISONS WITH FORMER SCHEMES UNDER DB3 AND DB4 DATABASES.

DB3			DB4		
Method	Feature Dimension	ARR	Method	Feature Dimension	ARR
IH	256	55.50	FC [52]	(M/B)^2	23.4
FDI [51]	$M \times N$	21.40	HWQCS [53]	16	45.1
FC [52]	$4 \times M/B \times N/B$	21.40	JHMS [55]	256	71.2
HWQCS [53]	8	44.50	FIRE [56]	20	51.7
Index-1[54]	64	51.70	FIRE without (xd,yd) [56]	20	67.2
Index-2[54]	68	66.10	FS [57]	68	58.2
Index-3[54]	256	71.00	NFBMFV [58]	68	50.3
Index-4[54]	256	67.80	HE [59]	13	53.2
Proposed Method	48 ($N_{min} = N_{max} = N_b = 16$)	67.46	JHSE [59]	4×13	67.4
Proposed Method	96 ($N_{min} = N_{max} = N_b = 32$)	71.83	JHMSE [59]	$16 \times 4 \times 13 = 832$	85.3
Proposed Method	192 ($N_{min} = N_{max} = N_b = 64$)	80.96	HM+JHSE [59]	$16 + 4 \times 13 = 68$	82.7
Proposed Method	256 ($N_{min} = N_b = 128$)	**88.00**	HM+HS+HE [59]	$16 + 4 + 13 = 33$	81.2
Proposed Method	256 ($N_{max} = N_b = 128$)	87.30	HM+HS [59]	$16 + 4 = 20$	69.7
			Proposed Method	48 ($N_{min} = N_{max} = N_b = 16$)	83.57
			Proposed Method	96 ($N_{min} = N_{max} = N_b = 32$)	83.69
			Proposed Method	192 ($N_{min} = N_{max} = N_b = 64$)	**84.46**
			Proposed Method	256 ($N_{min} = N_b = 128$)	78.83
			Proposed Method	256 ($N_{max} = N_b = 128$)	79.41

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

information and distribution in an image. Table III shows the comparison results among the proposed method and former LBP-based image retrieval systems under the textural image datasets. As it can be seen, the proposed method outperforms the former LBP-based image retrieval systems under textural databases DB1 and DB2. However, the proposed method is slightly inferior to ODII [60] in textural color image database DB2. Based on this result, the proposed method performs well to index the natural images and grayscale images.

The other experiment was conducted to have an in-depth investigation of the proposed method in processing textural image. In this experiment, the proposed method is compared with the fractal-based image retrieval systems [51-59]. As reported in [51-59], the fractal-based image retrieval methods achieve promising image retrieval accuracy, making it is suitable to index the textural images. In this experiment, the comparison was conducted with DB3 and DB4 databases by investigating the ARR values with all images in the databases are successively turned as query images. Table IV shows the experimental results of the proposed method compared with the fractal-based image retrieval systems. The proposed image retrieval scheme provides a great improvement compared with the former fractal-based schemes, and thus the proposed scheme can be a potential candidate in the content based image retrieval system.

D. Computational Complexity

It is very interesting to compare the computational complexity of the proposed method with several competing image indexing schemes [22, 35, 50, 60, 68]. The computation complexity is measured with the big-Oh notation. The time complexity is examined for the feature extraction and distance computation process. To make a fair comparison, the complexity of the feature extraction is evaluated in terms of image size, image block size, and codebook size (or neighboring operation for LBP-based schemes). On the other hand, the complexity of distance computation is investigated in terms of the feature dimensionality. Let the image size, image block size, codebook size, and feature dimensionality (histogram length) are denoted as $M \times N$, $m \times n$, d, and h. Table V shows the comparison of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

TABLE V. COMPUTATIONAL COMPLEXITY COMPARISON. THE TIME COMPLEXITY OF FEATURE EXTRACTION IS COMPUTED IN TERMS OF IMAGE SIZE, BLOCK SIZE, AND CODEBOOK SIZE. THE TIME COMPLEXITY OF DISTANCE COMPUTATION IS MEASURED IN TERMS OF FEATURE DIMENSIONALITY.

Method	Feature Extraction	Distance Computation	Remarks
SEC+GVC+GDC [69]	$O(\frac{M}{2} \times \frac{N}{2} \times d)$	$O(h_1 + h_2 + h_3) \approx O(h)$	$d = 8,\ h_1 = 8,\ h_2 = 32,\ h_3 = 90$
MCMCM+DBPSP [68]	$O(\frac{M}{2} \times \frac{N}{2} \times d)$	$O(9h_1^2 + h_2) \approx O(h^2)$	$d = 7,\ h_1 = 7,\ h_2 = 6$
LBP [35]	$O(M \times N)$	$O(h)$	$h = 59$
LTrP [50]	$O(M \times N \times d)$	$O(13h) \approx O(h)$	$d = 32,\ h = 59$
Poursistani [22]	$O(\frac{M}{8} \times \frac{N}{8} \times d)$	$O(h)$	$d = 256,\ h = 256$
ODII [60]	$O(\frac{M}{m} \times \frac{N}{n} \times d)$	$O(2h_1^2 + h_2) \approx O(h^2)$	$d = 256,\ h_1 = 64,\ h_2 = 256$
Proposed Method	$O(\frac{M}{m} \times \frac{N}{n} \times d)$	$O(2h_1 + h_2) \approx O(h)$	$d, h_1, h_2 \in \{4,8,\dots,256\}$ $m, n \in \{4,8,\dots,32\}$
$M \times N$: image size, $m \times n$: image block size, d: codebook size, h: feature dimensionality			

computational complexity among the proposed method with the competitive schemes. As it can be seen, the proposed method has a competitive computation complexity for feature extraction and distance computation stage.

CONCLUSIONS

A new method is proposed in this study 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 the former existing methods in the literature related to the content based image retrieval. 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, lab, etc. 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.

REFERENCE

[1] E. J. Delp and O. R. Mitchell, "Image coding using block truncation coding," *IEEE Trans. Commun.*, vol. 27, pp. 1335–1342, Sept. 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, 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.

[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, pp. 126-135, 2011.

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

[10] Y. F. Liu, J. M. Guo, and J. D. Lee, "Halftone image classification using LMS algorithm and naïve Bayes," *IEEE Trans. Image Process.*, vol. 20, no. 10, pp. 2837-2847, Oct. 2011.

[11] G. Qiu, "Color Image Indexing Using BTC," *IEEE Trans. Image Processing*, Vol. 12, No. 1, Jan. 2003.

[12] M. R. Gahroudi, and M. R. Sarshar, "Image retrieval based on texture and color method in BTC-VQ compressed domain," *Int. Symp. on Signal Processing and Its Application*, 20th, Feb. 2007.

[13] F.X. Yu, H. Luo, and Z.M. Lu, "Colour image retrieval using pattern co-occurrence matrices based on BTC and VQ," *Electronics Letters*, 20th, vol. 47, no. 2, Jan. 2011.

[14] S. Silakari, M. Motwani, and M. Maheshwari, "Color image clustering using block truncation algorithm," *Int. Jour. of Comp. Science Issues*, vol. 4, no. 2, 2009.

[15] C. H. Lin, R. T. Chen, and Y. K. Chan, "A smart content-based image retrieval system based on color and texture feature," *Image and Vision Computing*, vol. 27, no. 6, pp. 658–665, May 2009.

[16] N. Jhanwar, S. Chaudhurib, G. Seetharamanc, and B. Zavidovique, "Content based image retrieval using motif co-occurrence matrix," *Image and Vision Computing*, vol. 22, pp. 1211–1220, Dec. 2004.

[17] P. W. Huang, and S. K. Dai, "Image retrieval by texture similarity," *Pattern Recognition*, vol. 36, no. 3, pp. 665–679, Mar. 2003.

[18] T. C. Lu, and C. C. Chang, "Color image retrieval technique based on color features and image bitmap," *Inf. Process. Manage.*, vol. 43, no. 2, pp. 461-472, Mar. 2007.

[19] T. W. Chiang and T. W. Tsai, "Content-based image retrieval via the multiresolution wavelet features of interest," *J. Inf. Technol. Appl*, Vol. 1, no. 3, pp. 205-214, Dec. 2006.

[20] C. C. Lai, and Ying-Chuan Chen, "A user-Oriented Image Retrieval System Based on Interactive Genetic Algorithm," *IEEE Trans. Inst. Meas.*, vol. 60, no. 10, October 2011.

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

[22] 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*, 2011, http://dx.doi.org/10.1016/j.mcm.2011.11.064.

[23] M. E. ElAlami, "A novel image retrieval model based on the most relevant features," *Knowledge-Based Syst.*, vol. 24, no. 1, 2011.

[24] P. Franti, and T. Kaukoranta, "Binary vector quantizer design using soft centroids," *Signal Proc.: Image Comm.*, vol. 14, no. 9, pp. 677-681, 1999.

[25] Corel Photo Collection Color Image Database, online available on http://wang.ist.psu.edu/docs/realted/.

[26] T. M. Saadatmand, and H. A. Moghaddam, "A novel evolutionary approach for optimizing content based image retrieval," *IEEE Trans. System, Man, and Cybernetics*, vol. 37, no. 1, pp. 139-153, 2007.

[27] H. A. Moghaddam, and T. M. Saadatmand, "Gabor wavelet correlogram algorithm for image indexing and retrieval," *18th Intl. Conf. Pattern Recognition*, pp. 925-92, Iran, 2006.

[28] M. Subrahmanyam, R. P. Maheswari, and R. Balasubramanian, "Expert system design using wavelet and color vocabulary trees for image retrieval," *Expert Systems with Applications*, vol. 39, no. 5, pp. 5104-5114, 2012.

[29] E. Yildizer, A. M. Balci, M. Hassan, and R. Alhajj, "Efficient content-based image retrieval using multiple support vector machines ensemble," *Expert Systems with Applications*, vol. 39, no. 3, 2012.

[30] W. Xingyuan, and W. Zongyu, "A novel method for image retrieval based on structure element's descriptor," *Journal of Visual Communication and Image Representation*, vol. 24, no. 1, pp. 63-74, 2013.

[31] W. Xingyuan, C. Feng, and Y. Jiao, "An effective method for color image retrieval based on texture," *Computer Standards and Interfaces*, vol. 34, no. 1, pp. 31-35, 2012.

[32] W. Xingyuan, and C. Feng, "A fast fractal coding in application of image retrieval," *Fractals*, vol. 17, no. 4, pp. 441-450, 2009.

[33] W. Xingyuan, L. Fanping, and W. Shuguo, "Fractal image compression based on spatial correlation and hybrid genetic algorithm," *Journal of Visual Communication and Image Representation*, vol. 20, no. 8, pp. 505-510, 2009.

[34] V. Arvis, C. Bebain, M. Berducat, et. al., "Classification of textures by multispectral co-occurrence matrix," Image Analysis Stereology, vol. 23, no. 1, pp. 63-72, 2004.

[35] T. Ojala, M. Pietikainen, and D, Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51-59, 1996.

[36] X. Tan, and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE. Trans. Image Proc.*, vol. 19, no. 6, pp. 1635-1650, 2010.

[37] Z. Guo, L. Zhang, and D. Zhang, "Rotation invariant texture classification using LBP variance with global matching," *Pattern Recognition*, vol. 43, pp. 706-716, 2010.

[38] J. Ning, L. Zhang, D. Zhang, W. Chengke, "Robust object tracking using joint color-texture histogram," Int. J. Pattern Recognition and Artificial Intelligence, vol. 23, no. 7, 2009.

[39] B. S. Manjunath, and W. Y. Ma, "Texture feature for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Machine Intel*. Vol. 18, no. 8, pp. 837-842, 1996.

[40] M. Kokare, P. K. Biswas, and B. N. Chatterji, "Texture image retrieval using new rotated complex wavelet filters," *IEEE Trans. Systems, Man, Cyber.*, vol. 33, no. 6, pp. 1168-1178.

[41] M. Subrahmanyam, R. P. Maheswari, and R. Balasubramanian, "Local maximum edge binary patterns: A new descriptor for image retrieval and object tracking," *Signal Processing*, vol. 92, pp. 1467-1479, 2012.

[42] G. H. Liu, L. Zhang, Y. K. Hou, Z. Y. Li, and J. Y. Yang, "Image retrieval based on multitexton histogram," *Pattern Recognition*, vol. 43., no. 7, pp. 2380-2389, 2010.

[43] G. H. Liu, Z. Y. Li, L. Zhang, and Y. Xu, "Image retrieval based on micro-structure descriptor," *Pattern Recognition*, vol. 44, no. 9, pp. 2123-2133, 2011.

[44] C. H. Lin, D. C. Huang, Y. K. Chan, K. H. Chen, and Y. J. Chang, "Fast color-spatial feature based image retrieval methods," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11412-11420, 2011.

[45] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and texture descriptors," *IEEE Trans. Circuit and Systems for Video Tech.*, vol. 11, no. 6, pp. 703-715, 2001.

[46] F. Mahmoudi, and J. Shanbehzadeh, "Image retrieval based on shape similarity by edge

orientation auto correlogram," *Pattern Recognition*, vol. 36, no. 8, pp. 1725-1736, 2003.

[47] P. Brodatz, "Textures photographic album for artist and designers," New York, 1996.

[48] Texture Image Database of University of Southern California, Signal and Image Processing Institute, online available on http://sipi.usc.edu/database/.

[49] Vision Texture, MIT Vision and Modeling Group, online available on http://vismod.www.media.mit.edu.

[50] S. Murala, R. P. Maheshwari, and R. Balasubramanian, "Local tetra patterns: a new feature descriptor for content-based image retrieval," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2874-2886, May 2012.

[51] A. Lasfar, S. Mouline, D. Aboutajdine, and H. Cherifi, "Content-based retrieval in fractal coded image database," in *Proc. 15th Int. Conf. Pattern Recognition*, vol. 1, pp. 1031-1034, 2000.

[52] A. Zhang, B. Cheng, and R. Acharya, "An approach to query-by-texture in image database system," in *Proc. SPIE Conf. Digital Image Storage and Archiving Systems*, Oct. 1995.

[53] B. Schouten, and P. Zeeuw, "Image databases, scale, and fractal transforms," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, pp. 534-537, 2000.

[54] M. Pi, M. K. Mandal, and A. Basu, "Image retrieval based on histogram of fractal parameters," *IEEE Trans. Multimedia*, vol. 7, no. 4, pp. 597-605, 2005.

[55] M. Pi, M. Mandal, and A. Basu, "Image retrieval based on histogram of new fractal parameters," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process.*, vol. 3, pp. 585-588, 2003.

[56] R. Distasi, M. Nappi, and M. Tucci, "FIRE: fractal indexing with robust extensions for image databases," *IEEE Trans. Image Process.*, vol. 12, no. 3, pp. 373-384, 2003.

[57] S. Peleg, J. Naor, R. Hartley, and D. Avnir, "Multiple resolution texture analysis and classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, no. 4, pp. 518-523, 1984.

[58] C. C. Chen, J. S. Daponte, and M. D. Fox, "Fractal feature analysis and classification in medical imaging," *IEEE Trans. Med. Imaging*, vol. 8, no. 2, pp. 133-142, 1989.

[59] M. Pi, and H. Li, "Fractal indexing with the joint statistical properties and its application in texture image retrieval," *IET Image Process.*, vol. 2, no. 4, pp. 218-230, 2008.

[60] J. M. Guo, H. Prasetyo, and H. S. Su, "Image indexing using the color and bit pattern feature fusion," *J. Vis. Commun. Image Representation*, vol. 24, no. 8, pp. 1360-1379, 2013.

[61] R. W. Floyd, and L. Steinberg, "Adaptive algorithm for spatial grey scale," in SID Int. Sym. Dig. Tech. Papers, pp. 36-37, 1975.

[62] P. Stucki, "Image processing for document reproduction," in Advances in Digital Image Processing, New York, NY: Plenum, pp. 177-218, 1979.

[63] J. F. Jarvis, C. N. Judice, and W. H. Ninke, "A survey of techniques for the display of continuous-tone pictures on bilevel displays," Comp. Graph. Image Processing, vol. 5, pp. 13-40, 1976.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

[64] R. L. Stevenson, and G. R. Arce, "Binary display of hexagonally sampled continuous-tone images," J. Opt. Soc. Amer. A, vol. 2, no. 7, pp. 1009-1013, 1985.

[65] D. L. Lau, and G. R. Arce, "Modern digital halforning (2nd)," CRC Press, pp. 83-89, 2008.

[66] T. Ojala, M. Pietikainen, and D. Harwood, "Multiresolution gray-scale and rotation invariance texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971-987, 2002.

[67] B. Zhang, Y. Gao, S. Zhao, and J. Liu, "Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 533-544, 2010.

[68] M. Subrahmanyam, Q. M. J. Wu, R. P. Maheshwari, and R. Balasubramanian, "Modified color motif co-occurrence matrix for image indexing," *Comp. Electrical Eng.*, vol. 39, no. 3, pp. 762-774, 2013.

[69] W. Xingyuan, and W. Zongyu, "The method for image retrieval based on multi-factors correlation utilizing block truncation coding," *Pattern Recog.*, vol. 47, no. 10, pp. 3293-3303, 2014.

[70] X. Wang, D. Zhang, and X. Guo, "Novel hybrid fractal image encoding algorithm using standard deviation and DCT coefficients," *Nonlinear Dynamics*, vol. 73, no. 1, pp. 347-355, 2013.

[71] X. Wang, D. Zhang, and X. Guo, "A novel image recovery method based on discrete cosine transform and matched blocks," *Nonlinear Dynamics*, vol. 73, no. 3, pp. 1945-1954, 2013.

[72] Y. Zhang, and X. Wang, "Fractal compression coding based on wavelet transform with diamond search," *Nonlinear Anal.: Real World Apps.*, vol. 13, no. 1, pp. 106-112, 2012.

[73] X. Y. Wang, and S. G. Wang, "An improved no-search fractal image coding method based on a modified gray-level transform," *Comp. Graphics*, vol. 32, no. 4, pp. 445-450, 2008.

[74] X. Y. Wang, Y. X. Wang, and J. J. Yun, "An improved no-search fractal image coding method based on a fitting plane," *Image and Vision Comp.*, vol. 28, no. 8, pp. 1303-1308, 2010.

[75] X. Y. Wang, and L. X. Zou, "Fractal image compression based on matching error threshold," *Fractals*, vol. 17, no. 1, pp. 109-115, 2009.

Jing-Ming Guo (M'04–SM'10) received the Ph.D. degree from the Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan, in 2004. He is currently a Professor with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan. His research interests include multimedia signal processing, biometrics, computer vision, and digital halftoning.

Dr. Guo is a senior member of the IEEE and a Fellow of the IET. He has been promoted as a Distinguished Professor in 2012 for his significant research contributions. He received the Outstanding youth Electrical Engineer Award from Chinese Institute of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology

Electrical Engineering in 2011, the Outstanding young Investigator Award from the Institute of System Engineering in 2011, the Best Paper Award from the IEEE International Conference on System Science and Engineering in 2011, the Excellence Teaching Award in 2009, the Research Excellence Award in 2008, the Acer Dragon Thesis Award in 2005, the Outstanding Paper Awards from IPPR, Computer Vision and Graphic Image Processing in 2005 and 2006, and the Outstanding Faculty Award in 2002 and 2003.

Dr. Guo will/has been the General Chair of IEEE International Conference on Consumer Electronics in Taiwan in 2015, and the Technical program Chair for IEEE International Symposium on Intelligent Signal Processing and Communication Systems in 2012, IEEE International Symposium on Consumer Electronics in 2013, and IEEE International Conference on Consumer Electronics in Taiwan in 2014. He has served as a Best Paper Selection Committee member of the IEEE Transactions on Multimedia. He has been invited as a lecturer for the IEEE Signal Processing Society summer school on Signal and Information Processing in 2012 and 2013. He has been elected as the Chair of the IEEE Taipei Section GOLD group in 2012. He has served as a Guest Co-Editor of two special issues for Journal of the Chinese Institute of Engineers and Journal of Applied Science and Engineering. He serves on the Editorial Board of the Journal of Engineering, The Scientific World Journal, International Journal of Advanced Engineering Applications, Detection, and Open Journal of Information Security and Applications. Currently, he is Associate Editor of the IEEE Transactions on Multimedia, IEEE Signal Processing Letters, the Information Sciences, and the Signal Processing.

Heri Prasetyo received the B.S. degree (with Honors) from Department of Informatics, Sepuluh Nopember Institute of Technology, Surabaya, Indonesia, in 2006, and the M.S. degree from Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, in 2009. He is currently pursuing the Ph.D. degree in Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan. His research interests include image watermarking, data hiding, image retrieval, and meta-heuristics optimization.

Jen-Ho Chen received the B.S. degree from Department of Electronic Engineering, Jin Wen University of Science and Technology, Taiwan, in 2008. He is currently pursuing the M.S. degree in Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan. His research interests include inverse halftoning, biometrics system, computer vision, and image watermarking.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2358011, IEEE Transactions on Circuits and Systems for Video Technology