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/TIP.2014.2372619, IEEE Transactions on Image Processing

Content-Based Image Retrieval Using Features Extracted From Halftoning-Based Block Truncation Coding

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

***Abstract*—This paper presents a technique for Content-Based Image Retrieval (CBIR) by exploiting the advantage of low complexity Ordered-Dither Block Truncation Coding (ODBTC) for the generation of image content descriptor. In encoding step, ODBTC compresses an image block into corresponding quantizers and bitmap image. Two image features are proposed to index an image, namely Color Co-occurrence Feature (CCF) and Bit Pattern Features (BPF), which are generated directly from ODBTC encoded data streams without performing the decoding process. The CCF and BPF of an image are simply derived from the two ODBTC quantizers and bitmap, respectively, by involving the visual codebook. Experimental results show that the proposed method is superior to the Block Truncation Coding (BTC) image retrieval systems and the other former methods, and thus prove that the ODBTC scheme is not only suited for image compression since of its simplicity, but also offers a simple and effective descriptor to index images in CBIR system.**

***Index Terms*—Bit Pattern Feature, Color Co-occurrence Feature, Content-Based Image Retrieval, Ordered Dither Block Truncation Coding.**

I. Introduction

A SIGNIFICANT amount of research efforts have been devoted in addressing the Content Based Image Retrieval (CBIR) problem [11-22, 26-71]. An image retrieval system returns a set of images from a collection of images in the database to meet users' demand with similarity evaluations such as image content similarity, edge pattern similarity, color similarity, etc. An image retrieval system offers an efficient way to access, browse, and retrieve a set of similar images in the real-time applications. Several approaches have been developed to capture the information of image contents by directly computing the image features from an image as reported in [15-20]. In [21], the image feature is simply constructed in DCT domain. An improvement of image retrieval in DCT domain is presented in [22], in which the JPEG standard compression (excluding the entropy coding) is involved to generate the image feature. Some attempts have been addressed to describe the visual image content [26-28, 30-34]. Most of them are dealing with the MPEG-7 Visual Content Descriptor, including the Color Descriptors (CD), Texture Descriptor (TD), and Shape Descriptor (SD) to establish the international standard for the CBIR task. This standard provides a great advantage in the CBIR research field, in which some important aspects such as sharing the image feature for benchmark database, comparative study between several CBIR tasks, etc., become relatively easy to be conducted using these standard features. The standard also offers a great benefit in the distributed system, in which the image content descriptor can be remotely modified by the user. In this scenario, the original image is not necessary transferred over different locations, but only the image descriptor is required for modification and recalculation. A new type of CBIR approach is presented in [65], in which the spatial pyramid and orderless bag-of-features image representation were employed for recognizing the scene categories of images from a huge database. This method offers a promising result and outperforms the former existing methods in terms of the natural scene classification. The method in [66] presented the holistic representation of spatial envelop with a very low dimensionality for representing the scene image. This approach presented an outstanding result in the scene categorization. The method in [67] proposed a new approach for image classification with the receptive field design and the concept of over-completeness methodology to achieve a preferable result. As reported in [67], this method achieved the best classification performance with much lower feature dimensionality compared to that of the former schemes in image classification task.

The CBIR system which extracts an image feature descriptor from the compressed data stream has become an important issue. Since most of the images are recorded in the storage device in compressed format for reducing the storage space requirement. In this scenario, the feature extractor simply generates an image feature for the CBIR task from compressed data stream without performing the decoding (decompression) process. The Block Truncation Coding (BTC) is an image compression method which requires simple process on both encoding and decoding stages. The BTC compresses an image in a simple and efficient way [1]. BTC firstly divides an input image into several image blocks, and each image block is subsequently represented with two specific quantizers to maintain its mean value and standard deviation identical to the original image block. The BTC produces two quantizers, namely high and low quantizers, and a bitmap image at the end of the decoding process. The BTC decoding performs the reverse procedure by simply replacing the bitmap information with the high or low quantizer. BTC never requires auxiliary

Jing-Ming Guo and Heri Prasetyo are with the 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).

information during encoding and decoding stages such as the codebook information in the Vector Quantization (VQ) image compression or the quantization table in JPEG. The BTC maintains acceptable visual image, and the size of the datastream can be further reduced using the entropy coding.

The first CBIR system developed using the BTC can be found in [11]. The method exploits the nature of BTC to generate the image feature in which an image block is merely represented using two quantized values and the corresponding bitmap image. In the early work [11], two image features have been proposed, namely block color co-occurrence matrix and block pattern histogram, to index a set of images in database. The method in [11] utilizes the RGB color space, whereas the image indexing scheme in [13] employs the YCbCr color space for the generation of image feature. In [13], an image with RGB color space is firstly converted into the YCbCr color space, subsequently, the BTC encoding is performed only for Y color space. By employing VQ, two images features (contrast and visual pattern co-occurrence matrix and color pattern co-occurrence matrix) are generated from a YCbCr image. The method in [13] yields a better result in terms of the retrieval accuracy compared to that of the former methods as reported in [13]. Some improvements on the BTC-based image retrieval system can also be found in [12, 14], in which both methods utilize the RGB color space for the extraction of the image feature descriptor. In [12], the BTC encoding is performed on each color space (red, green, and blue) separately. The traditional color histogram and bit pattern codebook are subsequently extracted from each color channel. A different approach for CBIR system incorporating the color moments and K-means clustering can be found in the BTC-based indexing method [14]. As reported in [11-14], the BTC has demonstrated its efficiency and ability in the compression domain. The BTC scheme can also be conveniently and effectively employed to index images in database for CBIR applications.

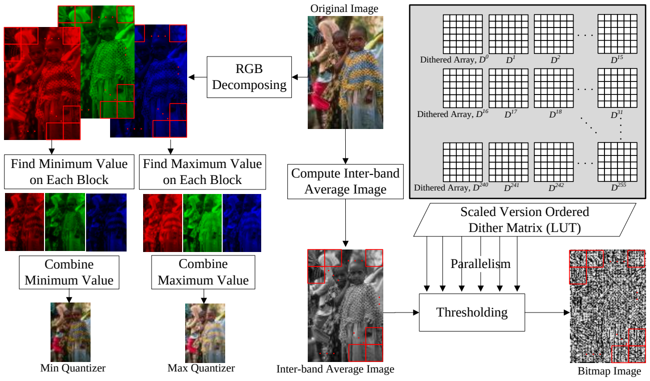


Fig. 1. Block diagram of the proposed ODBTC encoding for a color image.

BTC had played an important role in image coding. The successfulness of the BTC has been inspired many advanced coding techniques for its stability and simplicity. Several improvements and enhancements of the BTC scheme have been reported in literature [2-10] to further reduce the computational complexity, improve image quality, and achieve a higher compression ratio. The computational simplicity of the BTC and Halftoning-based Block Truncation Coding (HBTC) techniques have made it as an attractive tools in applications requiring fast real-time implementation. HBTC is an extended compression technique derived from BTC scheme, in which the BTC bitmap image is replaced with the halftone image. The main difference between the BTC and HBTC is on the image block quantizers determination. In contrast to the BTC scheme which tries to maintain its mean value and standard deviation in an image block, the HBTC quantizers are simply obtained from the minimum and maximum values found in an image block. The dithering-based BTC is an example of HBTC, in which the bit pattern configuration of the bitmap is merely generated from the dithering approach (void-and-cluster halftoning). The dithering-based BTC, namely Ordered Dither Block Truncation Coding (ODBTC) [7, 8], involves the low-pass nature of the Human Visual System (HVS) for achieving an acceptable perceptual image quality. It is based on the fact that the continuous and halftone images are perceived similarly by human vision when they are viewed from a certain distance. In encoding stage, the ODBTC scheme utilizes the dither array Look-Up-Table (LUT) to speed up the processing speed. The dither array in ODBTC method substitutes the fixed average value as the threshold value for the generation of bitmap image. The extreme values in ODBTC are simply obtained from the minimum and maximum value found in the image blocks. Given the high efficiency and low computational complexity of the ODBTC, some interesting applications have been developed based on it such as watermarking schemes [9, 10]. Thus, it offers a good solution for application requiring privacy and ownership protection.

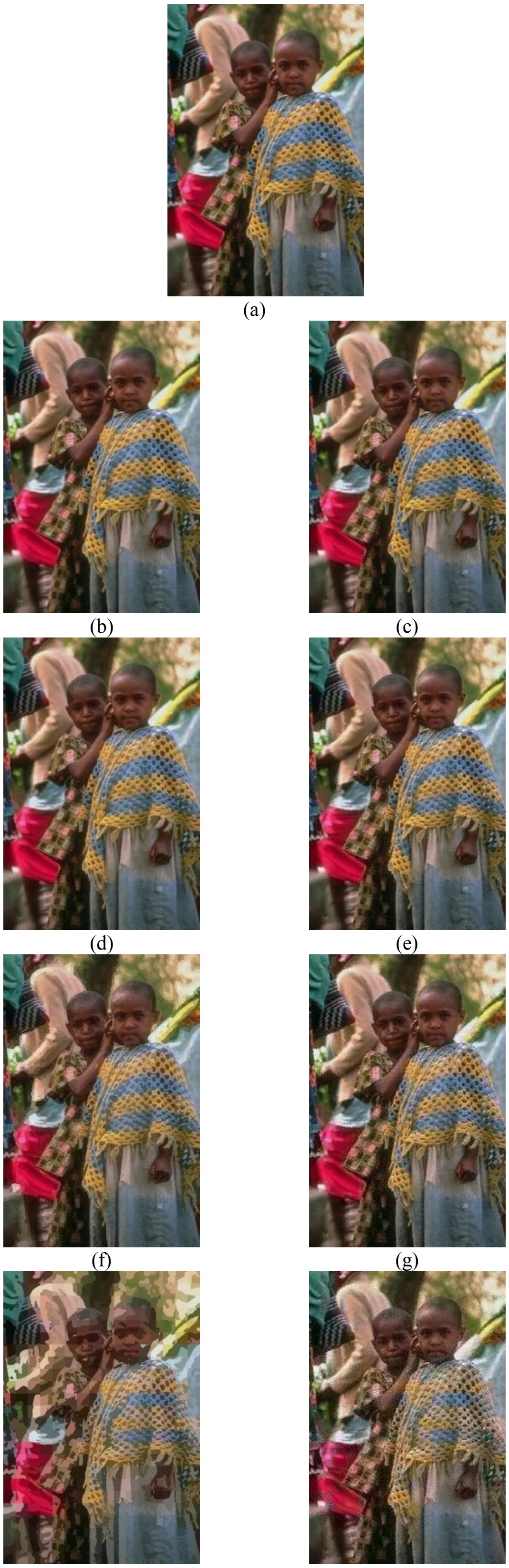
In this study, a new approach is proposed to index images in database using features generated from the ODBTC compressed data stream. This indexing technique can be extended for CBIR. ODBTC compresses an image into a set of color quantizers and a bitmap image. The proposed image retrieval system generates two image features, namely Color Co-occurrence Feature (CCF) and Bit Pattern Feature (BPF), from the above color quantizers and bitmap image, respectively. As documented in the experimental results, the proposed CBIR can provide promising results in terms of the retrieval accuracy compared to the state-of-the-arts.

The rest of this paper is organized as follows. The ODBTC is briefly introduced and generalized for color images in Section II to facilitate the understanding of the prospective readers. Section III elaborates the proposed CBIR system for the color and grayscale images. Extensive experimental results are reported in Section IV. Finally, the conclusion is drawn in Section V.

II. Ordered-Dither Block Truncation Coding and Its Color Extension

This section introduces the motivation of adopting the Ordered Dithered Block Truncation Coding (ODBTC) [7, 8], and its effectiveness in generating representative image features. In this paper, the ODBTC algorithm is generalized for color images in coping with the CBIR application. The main advantage of the ODBTC image compression is on its low complexity in generating bitmap image by incorporating the Look-Up Table (LUT), and free of mathematical multiplication and division operations on the determination of the two extreme quantizers. The traditional BTC derives the low and high mean

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/TIP.2014.2372619, IEEE Transactions on Image Processing



(a)

(b) (c)

(d) (e)

(f) (g)

(h) (i)

Fig. 2. Image quality comparison between the BTC and ODBTC. (a) original image, (b), (d), (f), (h) are images processed by the BTC with block of sizes 4×4, 8×8, 16×16, and 32×32, respectively. (c), (e), (g), (i) are ODBTC results with block of sizes 4×4, 8×8, 16×16, and 32×32, respectively.

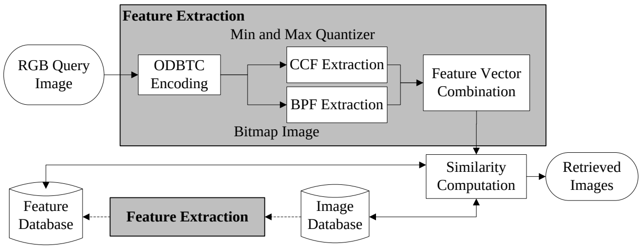


Fig. 3. Block diagram of the proposed image retrieval method.

values by preserving the first-order moment and second-order moment over each image block, which requires additional computational time. Conversely, ODBTC identifies the minimum and maximum values each image block as opposed to the former low and high mean values calculation, which can further reduce the processing time in the encoding stage. In addition, the ODBTC yields better reconstructed image quality by enjoying the extreme-value dithering effect compared to that of the typical BTC method as reported in [7, 8].

Given an original RGB color image of size $M \times N$. This image is firstly divided into multiple non-overlapping image blocks of size $m \times n$, and each image block can be processed independently. Figure 1 shows the conceptual block diagram of the ODBTC encoding for a color image. Let $B = \left\{b(i,j); i = 1,2,\dots,\frac{M}{m}; j = 1,2,\dots,\frac{N}{n}\right\}$ be a set of image blocks of size $m \times n$, containing the RGB color pixel information. The original image block $b(i,j)$ is firstly converted into the inter-band average image $\bar{b}_{k,l}(i,j)$ by

$$\bar{b}_{k,l}(i,j) = \frac{1}{3}\left[b_{k,l}^{red}(i,j) + b_{k,l}^{green}(i,j) + b_{k,l}^{blue}(i,j)\right]; k = 1,2,\dots,m; l = 1,2,\dots,n, \qquad (1)$$

where (k,l) denotes the pixel coordinate on image block (i,j). The inter-band average computation is applied to all image blocks.

The classical BTC approach performs the thresholding operation with a single threshold value obtained from the mean value of the pixels in an image block. A pixel of a smaller value compared to the threshold is turned to 0 (black pixel); otherwise it turns to 1 (white pixel) to construct the bitmap image representation.

As opposed the single threshold utilized in classical BTC, the ODBTC employs the void-and-cluster dither array of the same size as an image block to generate the bitmap image. Let $D(k,l)$ denotes the dither array coefficient at position (k,l), where $k = 1,2,\dots,m$ and $l = 1,2,\dots,n$. Let $D = \{D^0, D^1, \dots, D^{255}\}$ be a set of scaled version of dither array which can be easily computed as

$$D^d(k,l) = d\,\frac{D(k,l) - D_{min}}{D_{max} - D_{min}}, \qquad (2)$$

where D_{min} and D_{max} denote the minimum and maximum coefficient values in the dither array, respectively. The set $D = \{D^0, D^1, \dots, D^{255}\}$ can be off-line pre-calculated and stored as a Look-Up-Table (LUT) for later usage. Using this strategy, the computational time can be significantly reduced, making it suitable for the practical applications. The variable d denotes the dither array index in LUT, defined as $d = \bar{b}_{max}(i,j) - \bar{b}_{min}(i,j)$. Since $0 \le d \le 255$, it implies that all dither array coefficients $D^d(k,l)$ distribute in the range $[0,255]$. The variables $\bar{b}_{min}(i,j)$ and $\bar{b}_{max}(i,j)$ represent the minimum and maximum values, respectively, of the inter-band average image

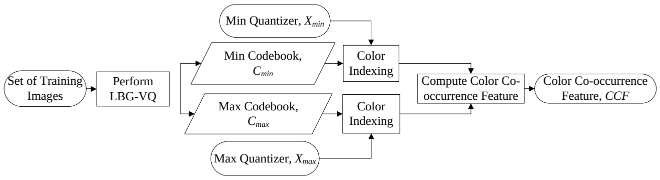


Fig. 4. Block diagram for computing the color co-occurrence feature.

on image block (i,j). These two values can be computed as

$$\bar{b}_{min}(i,j) = \min_{\forall k,l} \bar{b}_{k,l}(i,j), \tag{3}$$

$$\bar{b}_{max}(i,j) = \max_{\forall k,l} \bar{b}_{k,l}(i,j). \tag{4}$$

Subsequently, the ODBTC performs the thresholding on the inter-band average image with the scaled version of dither array for each image block to obtain the representative bitmap image $bm(i,j)$. The thresholding process for the pixels $\bar{b}_{k,l}(i,j)$ in each image block is given by

$$bm_{k,l}(i,j) = \begin{cases} 1; if\ \bar{b}_{k,l}(i,j) \geq \bar{b}_{min}(i,j) + D^d(k,l) \\ 0; if\ \bar{b}_{k,l}(i,j) < \bar{b}_{min}(i,j) + D^d(k,l) \end{cases}, \tag{5}$$

Except for sending the image bitmap to the decoder, ODBTC also transmits the two extreme color quantizers (minimum and maximum quantizers) to the decoder. The RGB color space is employed in this paper, thus the minimum and maximum quantizers are also in the RGB color representation. The set of minimum and maximum quantizers from all image blocks is given as

$$X_{min} = \{x_{min}(i,j); i = 1,2,\ldots,\frac{M}{m}; j = 1,2,\ldots,\frac{N}{n}\}, \tag{6}$$

$$X_{max} = \{x_{max}(i,j); i = 1,2,\ldots,\frac{M}{m}; j = 1,2,\ldots,\frac{N}{n}\}, \tag{7}$$

where $x_{min}(i,j)$ and $x_{max}(i,j)$ denote the minimum and maximum values, respectively, over red, green, and blue channels on the corresponding image block (i,j). The two values can be formally formulated as

$$x_{min}(i,j) = \left[\min_{\forall k,l} b_{k,l}^{red}(i,j), \min_{\forall k,l} b_{k,l}^{green}(i,j), \min_{\forall k,l} b_{k,l}^{blue}(i,j)\right], \tag{8}$$

$$x_{max}(i,j) = \left[\max_{\forall k,l} b_{k,l}^{red}(i,j), \max_{\forall k,l} b_{k,l}^{green}(i,j), \max_{\forall k,l} b_{k,l}^{blue}(i,j)\right], \tag{9}$$

for all $i = 1,2,\ldots,\frac{M}{m}$ and $j = 1,2,\ldots,\frac{N}{n}$.

At the end of the ODBTC encoding, the bitmap image, bm, the minimum quantizer, X_{min}, and maximum quantizer, X_{max}, are obtained and considered as encoded data stream, which are then transmitted to the decoder module over the transmission channel. The receiver decodes this encoded data stream to reconstruct the image. The decoder simply replaces the elements of value 0 in the bitmap by the minimum quantizer, and elements of value 1 in the bitmap by the maximum quantizer. Figure 2 shows the BTC and ODBTC reconstructed images over various image block sizes. It is clear that the ODBTC yields better reconstructed image quality compared to the traditional BTC scheme. The blocking effect and false contour are reduced in the ODBTC reconstructed image because of the halftoning-illusion from the dithering strategy. Except for the image compression, ODBTC compressed data stream, i.e., the bitmap image and two extreme color quantizers, can be further utilized as an image descriptor. A simple method for CBIR task is developed in this paper using the image feature derived from the ODBTC encoded data stream.

III. ODBTC Indexing

In this section, the proposed method is elaborated by introducing how to derive an image feature descriptor from the ODBTC data stream. Figure 3 illustrates the schematic diagram of the proposed CBIR method. The ODBTC employed in the proposed method decomposes an image into a bitmap image and two color quantizers which are subsequently exploited for deriving the image feature descriptor. Two image features are introduced in the proposed method to characterize the image contents, i.e., Color Co-occurrence Feature (CCF) and Bit Pattern Feature (BPF). The CCF is derived from the two color quantizers, and the BPF is from the bitmap image.

A. *Color Co-occurrence Feature (CCF)*

The color distribution of the pixels in an image contains huge amount of information about the image contents. The attribute of an image can be acquired from the image color distribution by means of color co-occurrence matrix. This matrix calculates the occurrence probability of a pixel along with its adjacent neighbors to construct the specific color information. This matrix also represents the spatial information of an image.

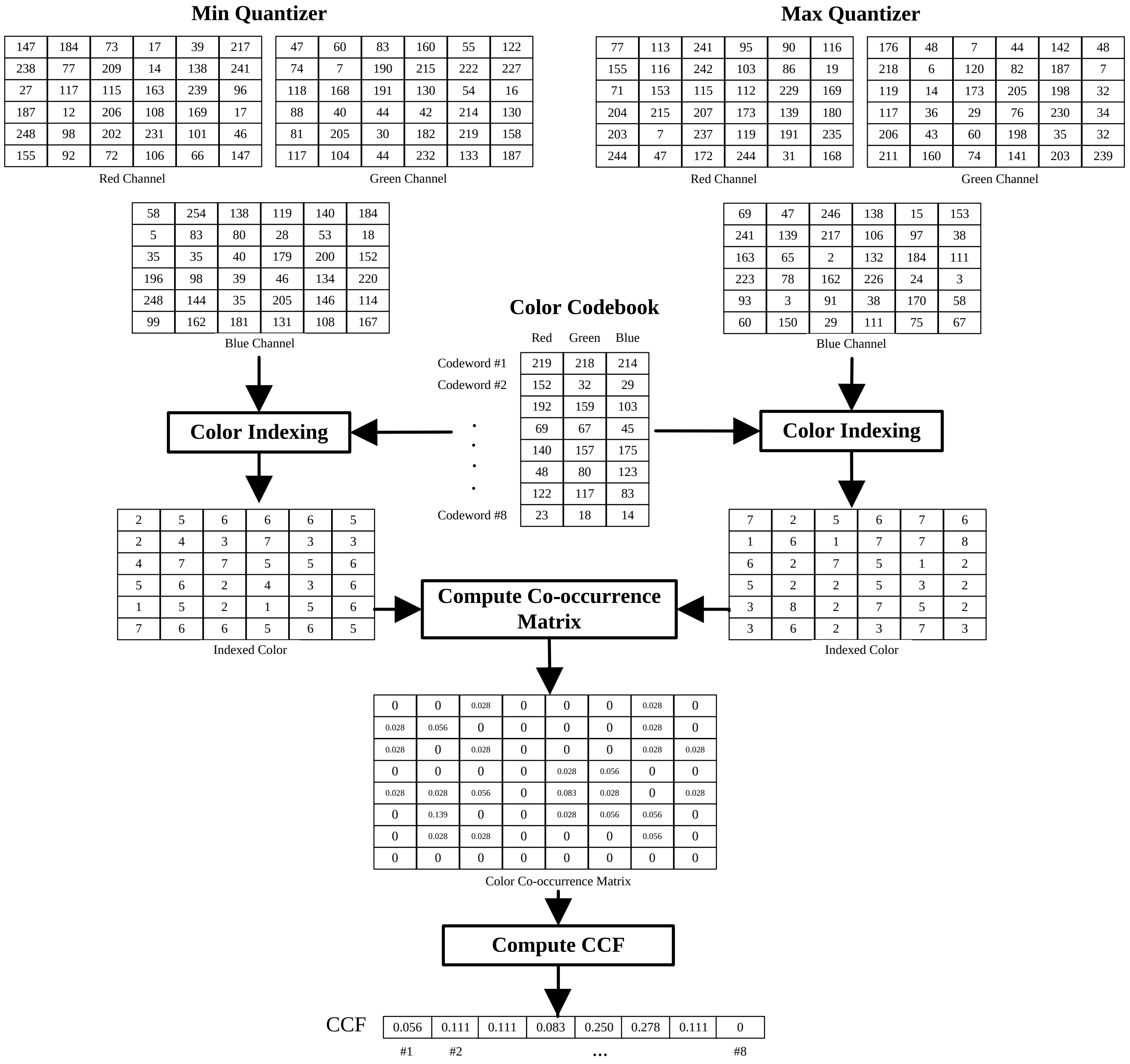
Color Co-occurrence Feature (CCF) can be derived from the color co-occurrence matrix. Figure 4 shows the schematic diagram of the CCF computation. In the proposed scheme, CCF is computed from the two ODBTC color quantizers. The minimum and maximum color quantizers are firstly indexed using a specific color codebook. The color co-occurrence matrix is subsequently constructed from these indexed values. Subsequently, the CCF is derived from the color co-occurrence matrix at the end of computation. In general, the color indexing process on RGB space can be defined as mapping a RGB pixel of three tuples into a finite subset (single tuple) of codebook index (the most representative codeword). LBG Vector Quantization (LBGVQ) generates a representative codebook from a number of training vectors. Let $C_{min} = \{c_1^{min}, c_2^{min}, \ldots, c_{N_c}^{min}\}$ and $C_{max} = \{c_1^{max}, c_2^{max}, \ldots, c_{N_c}^{max}\}$ be the codebooks generated from the minimum quantizer, X_{min} , and maximum quantizer, X_{max} , respectively. Herein, c_i^{min}, c_i^{max}, and N_c denote the codewords from minimum quantizer, codeword from maximum quantizer, and color codebook size, respectively. The color indexing process of the ODBTC minimum quantizer can be considered as the closest matching between the minimum quantizer value of each image block $x_{min}(i,j)$ and the codebook C_{min} which meets the following condition.

$$\tilde{\imath}_{min}(i,j) = \operatorname*{argmin}_{q=1,2,\ldots,N_c} \left\| x_{min}(i,j), c_q^{min} \right\|_2^2, \tag{10}$$

for all $i = 1,2,\ldots,\frac{M}{m}$ and $j = 1,2,\ldots,\frac{N}{n}$.

Similarly, the indexing for the maximum quantizer of each image block $x_{max}(i,j)$ with codebook C_{max} is formally defined as

$$\tilde{\imath}_{max}(i,j) = \operatorname*{argmin}_{q=1,2,\ldots,N_c} \left\| x_{max}(i,j), c_q^{max} \right\|_2^2 , \tag{11}$$



Min Quantizer – Red Channel

147	184	73	17	39	217
238	77	209	14	138	241
27	117	115	163	239	96
187	12	206	108	169	17
248	98	202	231	101	46
155	92	72	106	66	147

Min Quantizer – Green Channel

47	60	83	160	55	122
74	7	190	215	222	227
118	168	191	130	54	16
88	40	44	42	214	130
81	205	30	182	219	158
117	104	44	232	133	187

Min Quantizer – Blue Channel

58	254	138	119	140	184
5	83	80	28	53	18
35	35	40	179	200	152
196	98	39	46	134	220
248	144	35	205	146	114
99	162	181	131	108	167

Max Quantizer – Red Channel

77	113	241	95	90	116
155	116	242	103	86	19
71	153	115	112	229	169
204	215	207	173	139	180
203	7	237	119	191	235
244	47	172	244	31	168

Max Quantizer – Green Channel

176	48	7	44	142	48
218	6	120	82	187	7
119	14	173	205	198	32
117	36	29	76	230	34
206	43	60	198	35	32
211	160	74	141	203	239

Max Quantizer – Blue Channel

69	47	246	138	15	153
241	139	217	106	97	38
163	65	2	132	184	111
223	78	162	226	24	3
93	3	91	38	170	58
60	150	29	111	75	67

Color Codebook

	Red	Green	Blue
Codeword #1	219	218	214
Codeword #2	152	32	29
.	192	159	103
.	69	67	45
.	140	157	175
.	48	80	123
	122	117	83
Codeword #8	23	18	14

Indexed Color (Min Quantizer)

2	5	6	6	6	5
2	4	3	7	3	3
4	7	7	5	5	6
5	6	2	4	3	6
1	5	2	1	5	6
7	6	6	5	6	5

Indexed Color (Max Quantizer)

7	2	5	6	7	6
1	6	1	7	7	8
6	2	7	5	1	2
5	2	2	5	3	2
3	8	2	7	5	2
3	6	2	3	7	3

Color Co-occurrence Matrix

0	0	0.028	0	0	0	0.028	0
0.028	0.056	0	0	0	0	0.028	0
0.028	0	0.028	0	0	0	0.028	0.028
0	0	0	0	0.028	0.056	0	0
0.028	0.028	0.056	0	0.083	0.028	0	0.028
0	0.139	0	0	0.028	0.056	0.056	0
0	0.028	0.028	0	0	0	0.056	0
0	0	0	0	0	0	0	0

CCF

#1	#2						#8
0.056	0.111	0.111	0.083	0.250	0.278	0.111	0

Fig. 5. Example of CCF computation.

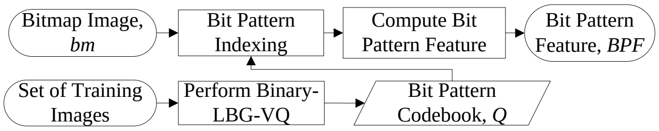


Fig. 6. Block diagram for computing the bit pattern feature.

for all $i = 1,2,\ldots,\frac{M}{m}$ and $j = 1,2,\ldots,\frac{N}{n}$.

After performing the color indexing for minimum and maximum quantizers, the color co-occurrence matrix (i.e., Color Co-occurrence Features (CCF)) for a given image can be directly computed as

$$CCF(t_1,t_2) = \Pr\left\{\tilde{i}_{min}(i,j) = t_1, \tilde{i}_{max}(i,j) = t_2 \middle| i = 1,2,\ldots,\frac{M}{m}; j = 1,2,\ldots,\frac{N}{n}\right\}, \quad (12)$$

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

The color co-occurrence matrix is a sparse matrix, in which the zeros dominate its entries. To further reduce the feature dimensionality of the CCF and to speed up the image retrieval process, the color co-occurrence matrix can be binned along its columns or rows to form a one-dimensional image feature descriptor. Thus, the feature dimensionality of the CCF is N_c, i.e., identical to the color codebook size. Figure 5 shows an example of the CCF computation. As it can be seen, the CCF calculation is simple, making it more preferable for CBIR task.

B. Bit Pattern Feature (BPF)

Another feature, namely Bit Pattern Feature (BPF), characterizes the edges, shape, and image contents. Figure 6 shows the schematic diagram for deriving the BPF. The binary vector quantization produces a representative bit pattern codebook from a set of training bitmap images obtained from the ODBTC encoding process. Let $Q = \{Q_1, Q_2, \ldots, Q_{N_b}\}$ be the bit pattern codebook consisting N_b binary codewords. These bit

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/TIP.2014.2372619, IEEE Transactions on Image Processing

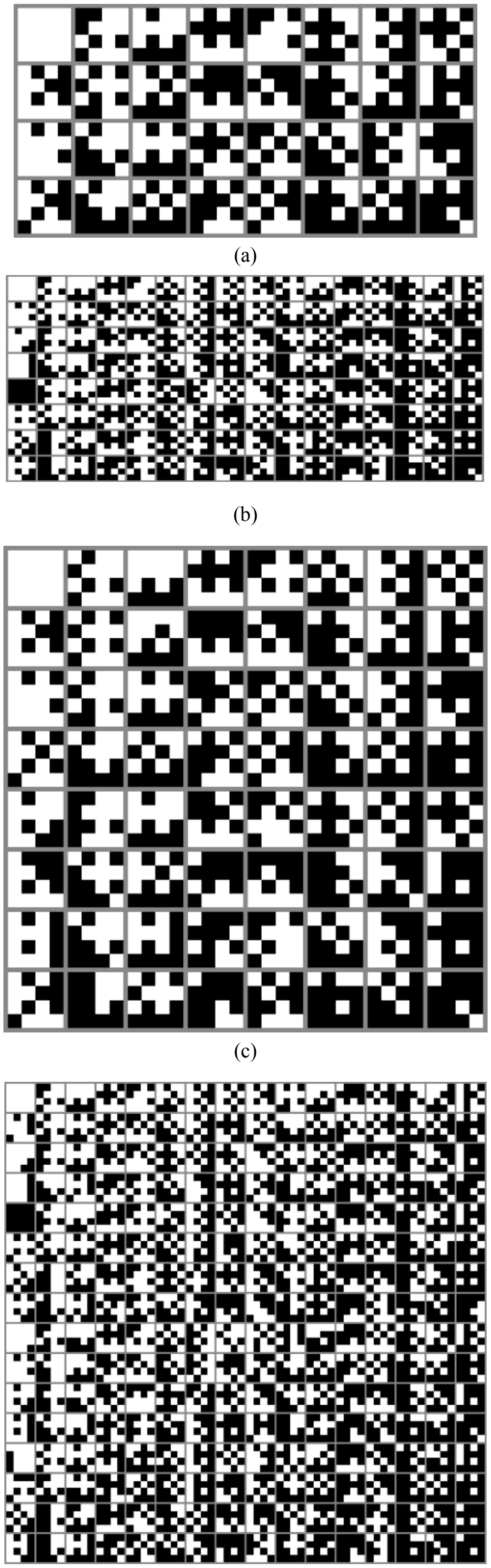


Fig 7. An example of bit pattern codebook trained using binary vector quantization. (a) $N_b = 32$, (b) $N_b = 128$, (c) $N_b = 64$, and (d) $N_b = 256$.

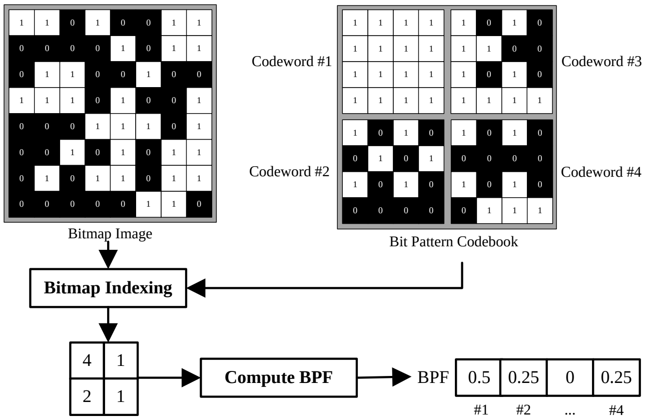


Fig 8. Example of BPF computation.

pattern codebooks are generated using binary vector quantization with soft centroids [23], and many bitmap images are involved in the training stage. At the codebook generation stage, all codevector components may have intermediate real values between zero (black pixel) and one (white pixel) as opposed to binary values. At the end of training stage, the hard thresholding performs the binarization of all codevectors to yield the final result. Figure 7 shows an example of bit pattern codebook over various codebook sizes.

The bitmap of each block $bm(i,j)$ is simply indexed based on the similarity measurement between this bitmap and the codeword Q_q which meets the following criterion.

$$\tilde{b}(i,j) = \underset{q=1,2,\dots,N_b}{\operatorname{argmin}} \; \delta_H\{bm(i,j), Q_q\}, \tag{13}$$

for all $i = 1,2,\dots,\frac{M}{m}$ and $j = 1,2,\dots,\frac{N}{n}$. The symbol $\delta_H\{\cdot,\cdot\}$ denotes the Hamming distance between the two binary patterns (vectors), i.e., bitmap image $bm(i,j)$ and bit pattern codeword Q_q.

Subsequently, the BPF is simply derived as the occurrence probability of the bitmap image mapped into the a specific bit pattern codeword Q_q. Thus, BPF is formally defined as

$$BPF(t) = \Pr\left\{\tilde{b}(i,j) = t \,\middle|\, i = 1,2,\dots,\frac{M}{m}; j = 1,2,\dots,\frac{N}{n}\right\}, \tag{14}$$

for all $t = 1,2,\dots,N_b$.

The feature dimensionality of the BPF is N_b, i.e., identical to the bit pattern codebook size. The overall dimensionality of the proposed feature descriptor is $N_c + N_b$. Figure 8 illustrates the BPF computation given an ODBTC bitmap image and a bit pattern codebook of size N_b. Similar to that of the CCF, the BPF only needs a simple computation, making it suitable for real applications where fast response is required.

C. The similarity measure of the features

The similarity between two images (i.e., a query image and the set of images in the database as target image) can be measured using the relative distance measure [11]. The similarity distance plays an important role for retrieving a set of similar images. The query image is firstly encoded with the ODBTC, yielding the corresponding CCF and BPF. The two feature are later compared with the features of target images in the database. A set of similar images to the query image is returned and ordered based on their similarity distance score, i.e,. the lowest score indicates the most similar image to the query image. Formally, the similarity measurement between

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/TIP.2014.2372619, IEEE Transactions on Image Processing

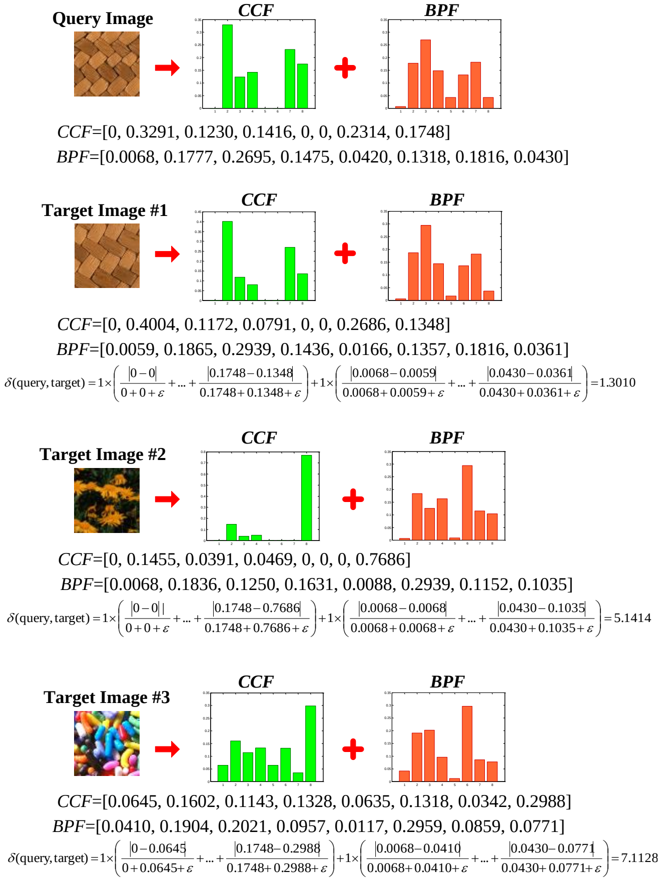


Fig 9. Example of similarity distance computation with $\{\alpha_1 = 1, \alpha_2 = 1\}$ and $\varepsilon = 10^{-16}$.

two images is defined as follows.

$$\delta(query, target) = \alpha_1 \sum_{t=1}^{N_c} \frac{|CCF^{query}(t)-CCF^{target}(t)|}{CCF^{query}(t)+CCF^{target}(t)+\varepsilon} + \alpha_2 \sum_{t=1}^{N_b} \frac{|BPF^{query}(t)-BPF^{target}(t)|}{BPF^{query}(t)+BPF^{target}(t)+\varepsilon}, \qquad (15)$$

where α_1 and α_2 denote the similarity weighting constants, representing the percentage contributions of the CCF and BPF in the proposed image retrieval system. A small number ε is placed at the denominator to avoid the mathematic division error. Notably, the CCF and BPF are from different modalities such that combining these features and determining the similarity weighting constants can be carried out through the experiments. Figure 9 illustrates an example of similarity distance computation. As shown in this figure, the target image #1 has the lowest similarity score, indicating it has the most similar appearance to the query image. In addition, the target image #1 is a scaled version of the query image.

IV. Experimental Results

Extensive experiments were conducted to examine the performance of the proposed method. The image descriptors are obtained from the ODBTC encoded data stream which is already stored in the database. First, CCF and BPF are computed over all images in the database. Subsequently, the system returns a set of similar images from the database based on their similarity distance. Finally, the image retrieval performance is tested when several images are turned as queries.

Table I
Summary of Image Databases Used in Experiment

Database Name	Image Size	Number of Class	Number of Image Each Class	Total Images
Corel [24]	384 × 256	10	100	1000
Brodatz-1856 [69]	128 × 128	116	16	1856
Vistex-640 [70]	128 × 128	40	16	640
ALOT [71]	192 × 128	250	16	4000
Holidays [72]	Vary	500	Vary	1491
UKBench [73]	640 × 480	2550	4	10200
Corel DB [24]	384 × 256	11	90	990
Vistex DB [70]	128 × 128	75	16	1200
Vistex-864 [70]	128 × 128	54	16	864
USPTex [74]	128 × 128	191	12	2292
Outex TC00013 [75]	128 × 128	68	20	1360
KTH-TIPS [76]	200 × 200	10	81	810
KTH-TIPS 2A [76]	200 × 200	11	Vary	4395

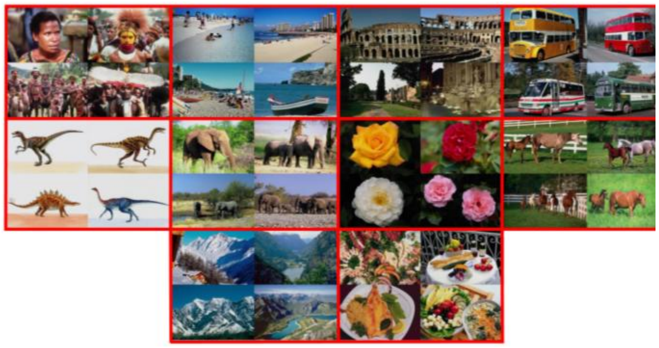


Fig 10. Sample images of each class/category in the database.

Many databases are involved in the experiments.

A. Performance Evaluation

In this subsection, a performance evaluation is introduced to measure the effectiveness of the proposed method and the former existing schemes. The proposed method performs the nearest neighbor searching to retrieve a set of similar images based on the similarity distance score. Four quantitative evaluations are used to examine the performance of proposed method, i.e., precision, recall, Average Retrieval Rate (ARR), and Average Normalized Modified Retrieval Rank (ANMRR). In the image classification task, the proposed method performance is measured with the proportion correct classification (accuracy) from the nearest neighbor classifier. The classifier assigns a class label of testing set using the similarity distance computation as used in the image retrieval task.

The similarity distance is computed and sorted in the ascending order between the query image q and target images in the database, and then the first L images are returned as a set of retrieved images. In image retrieval experiment, all images

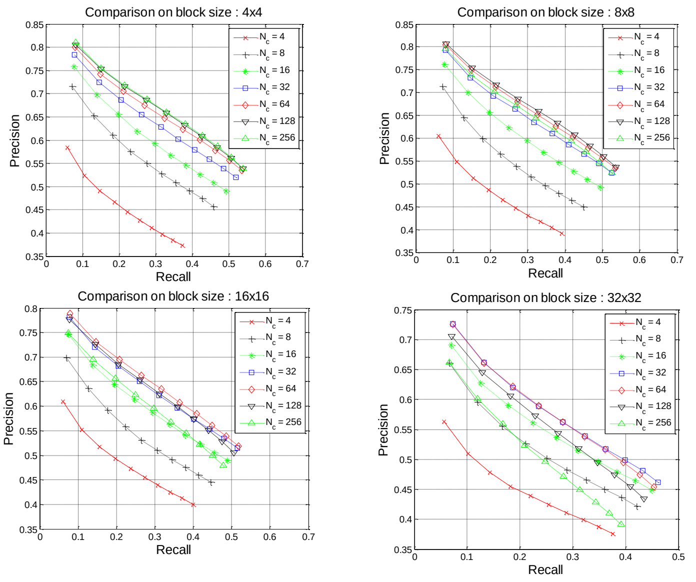


Fig 11. Comparison on different color codebooks, $\alpha_1 = 1$, and $\alpha_2 = 0$.

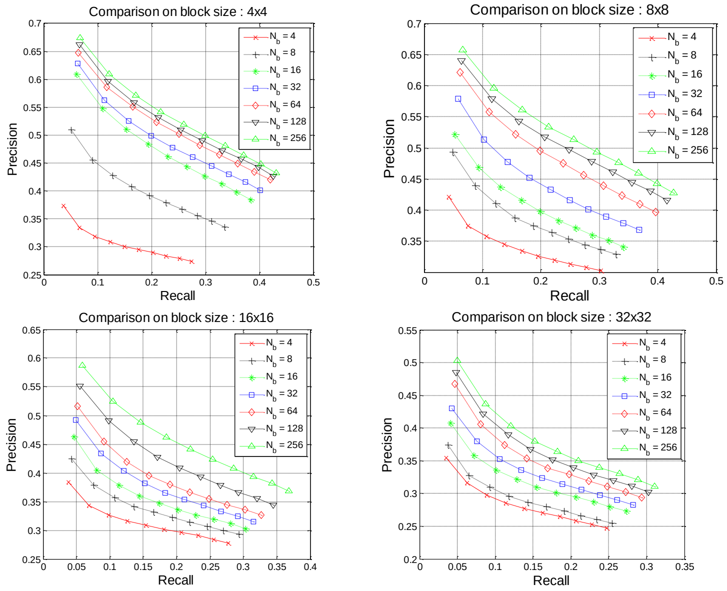


Fig 12. Comparison on different bit pattern codebook, $\alpha_1 = 0$, and $\alpha_2 = 1$.

are turned as query image such that the performance evaluation is conducted by averaging the values over all query images. Formally, the average precision $P(q)$ and average recall $R(q)$ measurements for describing the image retrieval performance are defined in [44] as below:

$$P(q) = \frac{1}{N_t L}\sum_{q=1}^{N_t} n_q(L), \qquad (16)$$

$$R(q) = \frac{1}{N_t N_R}\sum_{q=1}^{N_t} n_q(L), \qquad (17)$$

where L, N_t, and N_R denote the number of retrieved images, the number of images in database, and the number of relevant images on each class, respectively. The symbols q and $n_q(L)$ denote the query image and the number of correctly retrieved images among L retrieved images set, respectively.

The ARR defines the average precision rate $P(N_R)$ on retrieving N_R retrieved image. The ARR is formally defined as

$$ARR = \frac{1}{N_t N_R}\sum_{q=1}^{N_t} n_q(N_R), \qquad (18)$$

A higher value on precision, recall, and ARR indicate the higher retrieval rate or better retrieval performance.

ANMRR has been widely used in the MPEG-7 color core experiments [25, 26] for measuring the image retrieval performance. Let $B(q)$ be a set of relevant images returned by the system given the query image q. Let $Rank(k,q)$ be a similarity rank between image k and query image q, where $k \in B(q)$. With $K(q) \geq |B(q)|$, a relevant rank $RR\,(k,q)$ is later defined as:

$$RR(k,q) = \begin{cases} Rank(k,q), & \text{if } Rank(k,q) \leq K(q); \\ 1.25K(q), & \text{otherwise,} \end{cases} \qquad (19)$$

2here

$$K(q) = min[4 \cdot |B(q)|, 2 \cdot max\{B(q), \forall q\}], \qquad (20)$$

The average rank $AVR(q)$ of each query image q can be simply computed by averaging all $RR(k,q)$ as

$$AVR(q) = \langle RR(k,q) \rangle, \qquad (21)$$

Thus, a normalized modified retrieval rank $NMRR(q)$ for the query image q is defined as

$$NMRR(q) = \frac{AVR(q) - 0.5[1+B(q)]}{1.25K(q) - 0.5[1+B(q)]}, \qquad (22)$$

Finally, the ANMRR is obtained by averaging $NMRR(q)$ over all query images q as

$$ANMRR = \langle NMRR(q) \rangle, \qquad (23)$$

A lower value on ANMRR indicates that a more accurate result is achieved by the image retrieval system.

B. *Experimental Setup*

Thirteen image databases, as organized in Table I, are employed in this paper to examine the performance of the proposed method in the CBIR system as well as in the image classification task. These image databases contain various textural and natural images of different appearance in the grayscale and color space with different image sizes. All images in the databases are divided into several image classes semantic categories), in which all images under the same categories are regarded as similar images. For example, Corel image database consists of 1000 natural images grouped into 10 classes, in which each class contains 100 images. All images are of size 384 × 256 clustered into several semantic categories such as people, beach, building, buses, dinosaurs, elephants, flowers, horses, mountains, and foods. Figure 10 shows some sample images of each class in Corel image database.

The performance of proposed feature descriptor is tested for CBIR system using Corel, Brodatz-1856, Vistex-640, ALOT, Holidays, UKBench, Corel-DB, and Vistex-DB image databases. Whereas, the other image databases are tested for the image classification task. The image retrieval performance of the Corel is measured in terms of average precision and recall rate. The Bordatz-1856, Vistex-640, and ALOT image databases are examined in terms of the ARR value. The Holidays and UKBench image database are examined using the average precision rate and Normalized Score 4 (NS-4), respectively, as previously suggested in [45-51]. The performance of Corel DB and Vistex DB are investigated in terms of ANMRR. The other image databases employs the classification accuracy to assess the classification performance of the proposed feature descriptor.

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/TIP.2014.2372619, IEEE Transactions on Image Processing

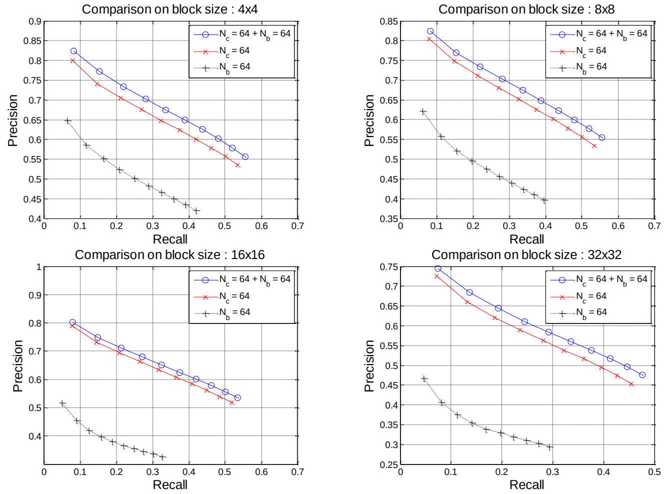


Fig 13. Comparison on features combination with various block sizes, $\alpha_1 = 1$, and $\alpha_2 = 1$.

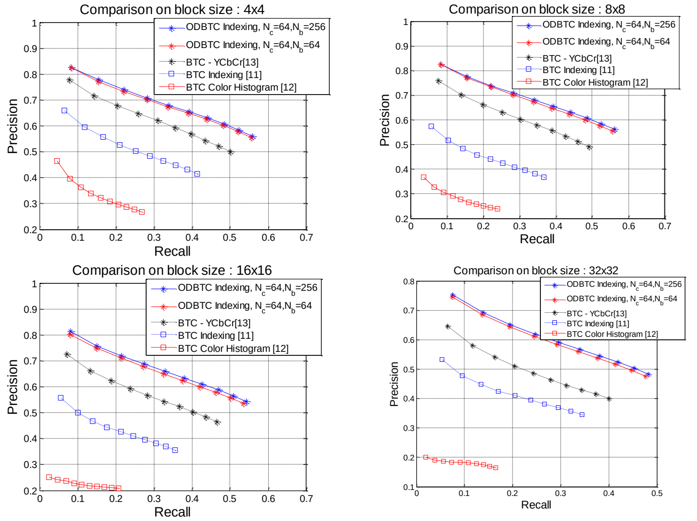
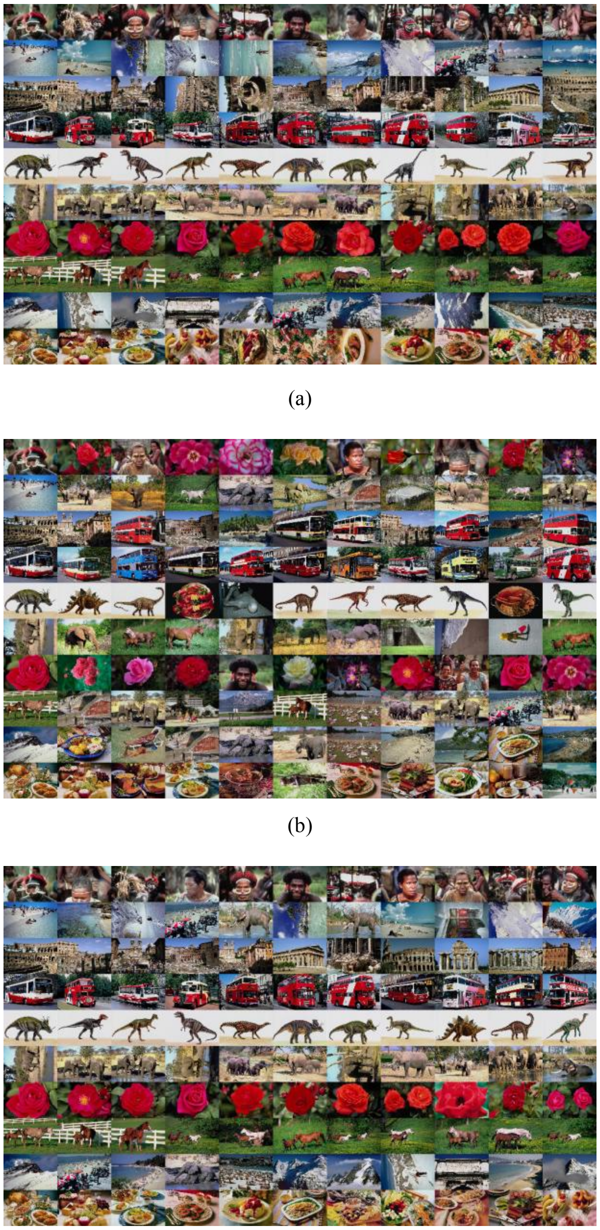


Fig 14. Comparison between the BTC-based and ODBTC-based image indexing.

For the Corel image database, five images are randomly selected from each class to form a training set for generating the color and bit pattern codebook. The color and bit pattern codebook are derived using the well-known LBG-VQ algorithm and the binary VQ with soft centroids, respectively. All training images are firstly encoded using ODBTC method over various image block sizes, 4×4, 8×8, 16×16, and 32×32. Subsequently, several color codebook of sizes, $N_c = \{4, 8, \ldots, 256\}$, and bit pattern codebook of sizes $N_b = \{4, 8, \ldots, 256\}$ are generated in this training stage. We employ the same training procedure for the other color image databases. In the case of grayscale image database, scalar VQ is employed to generate the color codebook. The similarity weighting constant is set as $\alpha_1, \alpha_2 = \{0,1\}$. Corel image database is employed in the subsequent three consecutive subsections to investigate the usability and effectiveness of proposed feature descriptor in CBIR system.

C. *Effects of codebook size*

In this subsection, the effect of different codebook sizes on the image retrieval performance is investigated. The ODBTC encodes all images in database over various image block sizes, 4×4, 8×8, 16×16, and 32×32, to obtain the image feature descriptor. Then, the color indexing is performed by incorporating the trained color codebook to produce the CCF. Meanwhile, the bit pattern indexing is also performed by means of the trained bit pattern codebook to obtain the BPF. Herein, various configurations of the similarity weighting constants are set, i.e., $\{\alpha_1 = 1, \alpha_2 = 0\}$, $\{\alpha_1 = 0, \alpha_2 = 1\}$ and $\{\alpha_1 = 1, \alpha_2 = 1\}$.



(a)

(b)

(c)

Fig 15. Example of top 10 retrieved images for each class with the proposed method. The images on the first column are query images randomly drawn from each class, and subsequent images from left to right are retrieved images ordered in ascending manner according to their similarity to the query images. (a) Proposed method with $\{\alpha_1 = 1, \alpha_2 = 0\}$, (b) $\{\alpha_1 = 0, \alpha_2 = 1\}$, and (c) $\{\alpha_1 = 1, \alpha_2 = 1\}$.

TABLE II
COMPARISONS AMONG THE PROPOSED SCHEME AND THE FORMER SCHEMES IN TERMS OF AVERAGE PRECISION RATE (MEAN VALUE±STANDARD DEVIATION) FOR COREL IMAGE DATABASE

Category Methods	People	Beach	Building	Bus	Dinosaur	Elephant	Flower	Horse	Mountain	Food	Average
Gahroudi [12]	0.411	0.344	0.335	0.249	0.889	0.297	0.789	0.204	0.209	0.235	0.396 ± 0.2434
Jhanwar [16]	0.453	0.398	0.374	0.741	0.915	0.304	0.852	0.568	0.293	0.370	0.526 ± 0.2305
Huang [17]	0.424	0.446	0.411	0.852	0.587	0.426	0.898	0.589	0.268	0.427	0.532 ± 0.2023
Chiang [19]	0.060	**0.930**	0.260	0.070	**1.000**	0.680	0.880	0.260	0.260	**0.930**	0.533 ± 0.3854
Silakari [14]	0.336	0.424	**0.792**	0.448	0.971	0.448	0.941	0.588	0.432	0.223	0.560 ± 0.2566
Qiu [11]	0.495	0.423	0.405	**0.909**	0.955	0.498	0.821	0.689	0.313	0.443	0.595 ± 0.2299
Lu [21]	0.630	0.318	0.480	0.674	0.988	0.494	0.884	0.588	0.338	0.612	0.600 ± 0.2139
Lu [18]	0.500	0.900	0.590	0.520	**1.000**	**0.800**	0.800	0.680	0.410	0.450	0.665 ± 0.2026
Yu [13]	**0.849**	0.356	0.616	0.818	**1.000**	0.591	0.931	0.928	0.404	0.682	0.717 ± 0.2245
Lin [15]	0.683	0.540	0.562	0.888	0.993	0.658	0.891	0.803	0.522	0.733	0.727 ± 0.1634
Poursistani [22]	0.702	0.444	0.708	0.763	**1.000**	0.638	0.924	**0.947**	**0.562**	0.745	0.743 ± 0.1754
ODBTC indexing	0.847	0.466	0.682	0.885	0.992	0.733	**0.964**	0.939	0.474	0.806	**0.779** ± 0.1898

All images in database are turned as query images sequentially, and then the average precision and recall rate is computed over all query images along with the number of retrieved images $L = \{10, 20, \dots, 100\}$. Figure 11 shows the average precision and recall rate of the proposed method over various color codebook sizes and several image block sizes. To examine the effectiveness of the CCF, we simply set the similarity weighting constants with $\{\alpha_1 = 1, \alpha_2 = 0\}$, implying that only CCF is used in the image retrieval. As it can be seen in Fig. 11, the proposed method is able to perform the color image retrieval. The color codebook sizes $N_c = \{32, 64, 128, 256\}$ offer satisfactory choices for performing the ODBTC image retrieval over various image block sizes.

The effectiveness of BPF is further investigated for the color image retrieval. The similarity weighting constants are conversely set at $\{\alpha_1 = 0, \alpha_2 = 1\}$. Figure 12 shows the average precision and recall rate over various bit pattern codebook sizes and several image block sizes with $L = \{10, 20, \dots, 100\}$. As shown in this figure, a greater size of bit pattern codebook leads to a higher precision and recall accuracy, yet it also leads to a higher computational burden.

Figure 13 shows the performance of the color image retrieval with the combination of CCF and BPF. In this experiment, we simply use the color and bit pattern codebook sizes as $N_c = 64$ and $N_b = \{64, 256\}$, respectively. The similarity weighting constants are set at $\{\alpha_1 = 1, \alpha_2 = 1\}$. Apparently, better precision and recall rates can be obtained from the combination of CCF and BPF compared to that of the results solely from CCF (with similarity weighting constants $\{\alpha_1 = 1, \alpha_2 = 0\}$) or BPF ($\{\alpha_1 = 0, \alpha_2 = 1\}$).

D. *Effects of the block size*

Several experiments were conducted to further investigate the image retrieval performance of proposed method over various image block sizes, i.e., 4×4, 8×8, 16×16, and 32×32. The proposed method is compared to the former BTC indexing method [11-13] under the same conditions to endorse a fair comparison. All images in the database are firstly encoded using the ODBTC and BTC schemes with different image block sizes 4×4, 8×8, 16×16, and 32×32. Subsequently, the image feature descriptors are obtained for both schemes. The color and bit pattern codebook sizes are set at $N_c = 64$ and $N_b = 256$, respectively, for the BTC [11] and proposed ODBTC scheme. The color codebook size for BTC-YCbCr indexing [13] is set at $N_c = 64$. The color histogram for BTC indexing [12] is set at $ch = 256$. The similarity weighting constants are set at $\{\alpha_1 = 1, \alpha_2 = 1\}$.

All images in database are turned as query images, then a set of retrieved images is returned and ordered in ascending order based on their similarity distance. The image retrieval accuracy is assessed with the average precision and recall values over all query images under different number of retrieved images $L = \{10, 20, \dots, 100\}$. Figure 14 shows the average precision and recall rates of the BTC and ODBTC image retrieval methods over various image block sizes. As it can be seen, the proposed method outperforms the BTC indexing method [11-13] over all image block sizes and all different numbers of retrieved images, $L = \{10, 20, \dots, 100\}$. The feature dimensionality of the proposed method, [11], [12], and [13] are $N_c + N_b$, $N_c^2 + N_b$, $9ch$, and $2N_c^2$, respectively. By setting $N_c = 64$ and $N_b = 64$, the proposed method is still better compared to [11-13] as shown in Fig. 14. Notably, the proposed method yields a better

TABLE III
COMPARISONS AMONG THE PROPOSED SCHEME AND THE FORMER SCHEMES IN TERMS OF ARR FOR BRODATZ-1856 AND VISTEX-640 IMAGE DATABASES

Method	Feature Dimension	Brodatz-1856	Vistex-640
GT with GGD&KLD [33]	4x6x2=36	17.19	76.57
DT-CWT [34]	(3x6+2)x2=40	74.73	80.78
DT-RCWT [34]	(3x6+2)x2=40	71.17	75.78
DT-CWT+DT-RCWT [34]	2x(3x6+2)x2=80	77.75	82.34
LBP [35]	59	79.97	82.23
LTP [36]	2x59=118	82.51	87.52
GLBP [37]	6x59=354	75.21	84.74
LMEBP [38]	8x512=4096	83.28	87.77
GLMEBP [38]	3x4x512=6144	82.01	87.93
LDP [39]	4x59=236	79.91	87.27
LTrP [40]	13x59=767	85.3	90.02
GLTrP [40]	3x13x59=2301	81.97	90.16
GLDP [39]	6x4x59=1416	79.11	87.51
GLTP [36]	6x2x59=708	78.75	80.98
MCMCM+DBPSP [41]	9x7x7+6=477	-	86.17
Proposed Method	16+16=32	63.24	80.04
Proposed Method	32+32=64	68.3	87.55
Proposed Method	64+64=128	75.62	89.42
Proposed Method	128+128=256	**85.8**	**90.67**

TABLE IV
COMPARISONS AMONG THE PROPOSED SCHEME AND THE FORMER SCHEMES IN TERMS OF ARR FOR ALOT IMAGE DATABASE

Methods	Feature Dimensionality	ALOT
Wbl-DT-CWT-1 scale [42]	3x2=6	23.28
Wbl-DT-CWT-2 scale [42]	6x2=12	33.56
Wbl-DT-CWT-3 scale [42]	9x2=18	40.01
GG-DT-CWT-1 scale [42]	3x2=6	23.86
GG-DT-CWT-2 scale [42]	6x2=12	33.38
GG-DT-CWT-3 scale [42]	9x2=18	39.33
GC-MWbl-DT-CWT-1 scale [42, 43]	1+1+4x4=18	27.69
GC-MWbl-DT-CWT-2 scale [42, 43]	1+1+4x4=18	37.68
GC-MWbl-DT-CWT-3 scale [42, 43]	1+1+4x4=18	43.25
GC-MGG-DT-CWT-1 scale [44]	1+1+4x4=18	30.58
GC-MGG-DT-CWT-2 scale [44]	1+1+4x4=18	38.2
GC-MGG-DT-CWT-3 scale [44]	1+1+4x4=18	43.06
Proposed Method	8+4=12	39.93
Proposed Method	8+8=16	**43.62**

TABLE V
COMPARISONS AMONG THE PROPOSED SCHEME AND THE FORMER SCHEMES IN TERMS OF AVERAGE PRECISION RATE (APR) AND NORMALIZED SCORE – 4 (NS-4) FOR HOLIDAYS AND UKBENCH IMAGE DATABASES

Method	APR for Holidays	NS-4 for UKBench
TF-IDF	0.604	3.393
TF-avgIDF	0.624	3.418
TF-maxIDF	0.601	3.408
Okapi-BM25 [45]	0.64	3.417
Jegou et.al. [46]	0.61	3.447
TF-pIDF [47]	0.642	3.454
Jegou et.al [46] + pIDF [47]	0.641	**3.476**
BoVW [48]	0.451	-
VTCW [49]	0.702	-
WGC [50]	0.595	-
Contextual Binary Code [51]	0.645	-
Proposed Method, 16+16=32	0.643	3.070
Proposed Method, 32+32=64	0.690	3.148
Proposed Method, 64+64=128	0.713	3.324
Proposed Method, 128+128=256	**0.733**	3.421

image retrieval accuracy compared to [11-13] under the same or lower feature dimensionality.

E. Example of the proposed image retrieval system

In this subsection, an example is provided to illustrate the practicability and usability of the proposed image retrieval system. Herein, an image is randomly picked from each class, and considered as a query image. Subsequently, a set of similar images from database are returned and ordered in ascending manner based on their similarity distance score. The effectiveness of proposed retrieval system is determined whether it is able or not to retrieve a set of images with the similar content appearances. Figure 15 presents retrieval examples of the proposed scheme with the similarity weight constants set at $\{\alpha_1 = 1, \alpha_2 = 0\}$, $\{\alpha_1 = 0, \alpha_2 = 1\}$, and $\{\alpha_1 = 1, \alpha_2 = 1\}$. The first column of Fig. 15 denotes the query image of each class, and the subsequent images from left to right are a set of returned images corresponding to the query image. As it can be seen, the combination of the CCF and BPF yields the better results in terms of the visual image appearances compared to using either CCF or BPF alone.

F. Comparison with former schemes

In this subsection, numerous former schemes are involved for performance comparisons. To ensure a fair comparison against the other methods, the ODBTC image block size is fixed at 4×4 for the image feature generation as formerly used in [11-15]. The color and bit pattern codebook sizes are set at $N_c = 64$ and $N_b = 256$. In [11], the color indexing for low and high mean values is performed by incorporating the local and global color tables of sizes 256 and 64, respectively. The proposed method and the method in [11] consider the RGB color space, while the method in [13] exploits the YCbCr color information for performing image retrieval. The color codebook size in [13] is also 64 and generated using the LGB-VQ with five training images on each class. The similarity weighting constants are set at $\{\alpha_1 = 1, \alpha_2 = 1\}$ for the proposed method.

In this experiment, all images in database are turned as the query images. The average precision rate is computed among all query images with the number of retrieved images set at $L = 20$. Table II reports the comparisons among the proposed scheme and the former methods in terms of the average precision rate with $L = 20$. As it can be seen in this Table, the proposed method does not consistently achieve the best average precision rate on each image class, however, the proposed method yields stable average precision rates as indicated with a small value of standard deviation. In addition, the proposed method achieves the best retrieval accuracy compared to the former existing schemes even though the image descriptors are simply derived from low level visual features, i.e., color distribution and ordered dither bit pattern of block-based ODBTC encoded data stream. In addition, the ten images are turned as query images in comparison with [20]. In this case, the average precision rate of [20] is 0.806, while the proposed scheme achieves 0.9187 under the same experimental setup. Thus, the proposed scheme achieves a better image retrieval accuracy compared to that of the former scheme [20].

An additional experiment were carried out to further

TABLE VI
COMPARISONS AMONG THE PROPOSED METHOD AND THE FORMER SCHEME IN TERMS OF ANMRR USING COREL-DB AND VISTEX-DB IMAGE DATABASES

Method	Color Space	Dimension	Corel DB	Vistex DB
Color histogram [27]	RGB	128	0.4622	0.2637
SCD [29]	HSV	128	0.3574	0.2047
CSD [29]	HMMD	128	0.4040	0.1949
Color autocorrelogram [28]	HSV	128	0.3480	0.1705
Wavelet moments	RGB	96	0.3949	0.6831
EHD [29]	RGB	240	0.2213	0.1193
BDIP-BVLC [30]	RGB, V	96	0.3751	0.1853
Color histogram+Wavelet moments	RGB, V	84(64+20)	0.3751	0.1853
SCD+BDIP-BVLC	HSV, V	96(64+32)	0.2310	0.1291
CSD+EHD	HMMD, V	144(64+80)	0.2861	0.3133
MRCTF [31]	HSV	92(60+32)	0.1837	0.0840
ODBTC Indexing	RGB	96(64+32)	**0.1792**	**0.0801**

TABLE VII
CLASSIFICATION PERFORMANCE AMONG THE PROPOSED SCHEME AND THE FORMER SCHEMES UNDER VISTEX 864, USPTEX, AND OUTEX TC-00013 IMAGE DATABASES

Methods	Feature Dimensionality	Vistex 864	USPTex	TC-00013
Average RGB	3	58.78	36.19	76.49
LBP+Haralick [52]	10	91.59	73.17	77.89
MSD [53]	72	82.07	51.29	49.83
Multilayer CCR [54]	640	94.74	82.08	68.50
HRF [55]	-	58.89	49.86	40.00
Gabor EEE [56, 27]	192	96.92	92.58	75.86
Shortest Graph [58]	96	87.62	66.71	**88.06**
Proposed Method	16+16=32	97.11	83.81	78.60
Proposed Method	32+32=64	98.38	88.66	76.25
Proposed Method	64+64=128	99.42	**91.23**	80.81
Proposed Method	128+128=256	**98.96**	90.79	78.31

TABLE VIII
CLASSIFICATION PERFORMANCE AMONG THE PROPOSED SCHEME AND THE FORMER SCHEMES UNDER KTH-TIPS IMAGE DATABASE

Methods	Feature Dimensionality	KTH-TIPS
CK-1 [59]	-	86.00
Sparse [60]	-	84.50
Proposed Method	16+16=32	97.28
Proposed Method	32+32=64	98.40
Proposed Method	64+64=128	99.14
Proposed Method	128+128=256	**99.63**

TABLE IX
CLASSIFICATION PERFORMANCE AMONG THE PROPOSED SCHEME AND THE FORMER SCHEMES UNDER KTH-TIPS 2A IMAGE DATABASE

Methods	Feature Dimensionality	KTH-TIPS 2A
SIFT [61]	-	52.70
LBP [62]	-	49.90
WLD [63]	-	56.40
DRLBP [64]	60	59.00
DRLTP [64]	176	62.60
Proposed Method	16+16=32	64.10
Proposed Method	32+32=64	68.87
Proposed Method	64+64=128	**73.09**
Proposed Method	128+128=256	69.88

investigate the proposed method superiority in the CBIR system under textural image databases Brodatz-1856, Vistex-640, and ALOT. The comparison is conducted among the proposed method and former LBP-based image retrieval schemes [35-40], statistical modeling of wavelet-based output [33, 34, 42-44], and modified color matrix [41]. In this experiment, the image retrieval accuracy is measured in terms of the ARR. Tables III and IV summarize the ARR comparison between the proposed method and former schemes under the textural image databases. In the case of Brodatz-1856, Vistex-640, and ALOT image databases, the proposed method offers a better image retrieval accuracy compared to that of the former schemes under the same or lower feature dimensionality.

Table IV reports the average precision rate and Normalized Score-4 (NS-4) under Holidays and UKBench color image databases. NS-4 is the recall value for the top-4 image candidates. In the case of the Holidays image database, the proposed method outperforms the former existing methods with the same or lower feature dimensionality. However, the proposed method is inferior to the former schemes under UKBench image database.

The other experiment is conducted between the proposed method and the former schemes in terms of the ANMRR. The accuracy of the proposed method is compared with various visual features in CBIR such as color histogram [27], color autocorrelogram [28], Scalable Color Descriptor (SCD) [29], Color Structure Descriptor (CSD) [29], Edge Histogram Descriptor (EHD) [29], Block Difference of Inverse Probabilities (BDIP) and Block Variation of Local Correlation Coefficients (BVLC) [30], and the Multiresolution Color and Texture Feature (MRCTF) [31]. The SCD and CSD are part of MPEG-7 color descriptor [29], and the EHD is one of the MPEG-7 texture descriptors [29]. Several schemes derive an image feature descriptor by combining several conventional features such as CSD+EHD, SCD+BDIP-BVLC, etc. The size of the color and bit pattern codebook for the proposed method are set at $N_c = 64$ and $N_b = 32$, respectively. The other experimental setting is set identical to that used in [31] to provide a fair comparison. Table VI shows the ANMRR comparison between the proposed method and the former methods. The proposed ODBTC indexing yields the better result compared to the former schemes [27-31] under Corel DB and VisTex DB image databases. The ODBTC indexing also provides a better retrieval accuracy compared to the MRCTF scheme [31] when the feature dimension is set nearly identical. It is worthy pointed out that the proposed ODBTC indexing outperforms the former CBIR system using MPEG-7 color and texture descriptor [29].

We also compare the proposed indexing with the most competitive scheme [31] in terms of the processing time. The simulation platform is with Intel Core 2 Quad CPU PC with 2.40GHz processor, 4.00 GB memory, Windows 7 Enterprise operating system, and Matlab R2010b. Experimental result shows that the proposed ODBTC indexing needs 27 ms for retrieving a similar image set for VisTex image database, while the method in [31] needs 38 ms.

An additional experiment is conducted to further investigate the effectiveness of the proposed image feature on the image classification task. Herein, the proposed method is compared to the former schemes using Vistex-864, USPTex, Outex

TC-00013, KTH-TIPS, and KTH-TIPS 2A image databases. The experimental setting is identical to the former existing methods to provide a fair comparison. The performance of image classification is assessed in terms of the correct classification percentage. Tables VII-IX summarize the image classification comparison among the proposed method and former schemes. The proposed method yields a better image classification result compared to the former schemes under Vistex-864, USPTex, KTH-TIPS, and KTH-TIPS 2A image databases with the same or lower feature dimensionality. However, the proposed method is inferior to the former schemes under Outex TC-00013 image database. The KTH-TIPS and KTH-TIPS 2A image databases are utilized to test whether a CBIR system is robust against illumination changing, rotation and scaling.

V. Conclusions

In this study, an image retrieval system is presented by exploiting the ODBTC encoded data stream to construct the image features, namely Color Co-occurrence and Bit Pattern features. As documented in the experimental results, the proposed scheme can provide the best average precision rate compared to various former schemes in the literature. As a result, the proposed scheme can be considered as a very competitive candidate in color image retrieval application.

For the further studies, the proposed image retrieval scheme can be applied to video retrieval. The video can be treated as sequence of image in which the proposed ODBTC indexing can be applied directly in this image sequence. The ODBTC indexing scheme can also be extended to another color space as opposed to the RGB triple space. Another feature can be added by extracting the ODBTC data stream, not only CCF and BPF, to enhance the retrieval performance. In the future possibilities, the system shall be able to bridge the gap between explicit knowledge semantic, image content, and also the subjective criteria in a framework for human-oriented testing and assessment.

References

[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] V. R. Udpikar and J. P. Raina, "BTC image coding using vector quantization," *IEEE Trans Commun.*, vol. COMM-35, pp. 352–256, Sep. 1987.

[3] Y. Wu and D. Coll, "BTC-VQ-DCT hybrid coding of digital images," *IEEE Trans. Commun.*, vol. 39, pp. 1283–1287, Sep. 1991.

[4] C. S. Huang and Y. Lin, "Hybrid block truncation coding," *IEEE Signal Process. Lett.*, vol. 4, no. 12, pp. 328–330, Dec. 1997.

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

[6] M. D. Lema, and O. R. Mitchell, "Absolute moment block truncation coding and its application to color images," *IEEE Trans. Commun.*, vol. COM-32, pp. 1148-1157, Oct. 1984.

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

[8] J. M. Guo, "High efficiency ordered dither block truncation with dither array LUT and its scalable coding application," *Digital Signal Proc.*, vol. 20, no. 1, pp. 97-110, Jan. 2010.

[9] J. M. Guo, M. F. Wu, and Y. C. Kang, "Watermarking in conjugate ordered dither block truncation coding images," *Signal Processing*, vol. 89, no. 10, pp. 1864-1882, Oct. 2009.

[10] J. M. Guo, and J. J. Tsai, "Reversible data hiding in highly efficient compression scheme," *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 2012-2024, Apr. 2009.

[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 Y. C. 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] P. Franti, and T. Kaukoranta, "Binary vector quantizer design using soft centroids," *Sign. Proc.: Image Comm.*, Vol. 14, page 677-681, 1999.

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

[25] P. Ndjiki-Nya, J. Restat, T. Meiers, J. R. Ohm, A. Seyferth, and R. Sniehotta, "Subjective evaluation of the MPEG-7 retrieval accuracy measure (ANMRR)," in ISO/WG11 MPEG Meeting, Geneva, Switzerland, May 2000, Doc. M6029.

[26] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and teture descriptors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 703-715, Jun. 2001.

[27] M. J. Swain, and D. H. Ballard, "Color indexing," *Int. J. Comput. Vis.*, vol. 7, pp. 11-32, 1991.

[28] J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu, and R. Zabih, "Image indexing using color correlograms," *in Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, Jun. 1997, pp. 762-768.

[29] ISO/IEC 15938-3/FDIS Information Technology - Multimedia Content Description Interface – Part 3 Visual Jul. 2001, ISO/IEC/JTC1/SC29/WG11 Doc. N4358.

[30] Y. D. Chun, S. Y. Seo, and N. C. Kim, "Image retrieval using BDIP and BVLC moments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13. Pp. 951-957, Sept 2003.

[31] Y. D. Chun, N. C. Kim, and I. H. Jang, "Content-Based Image Retrieval Using Multiresolution Color and Texture Features," *IEEE Trans. Multimedia*, vol. 10, no. 6, Oct. 2008.

[32] D. Nister, and H. Stewenius, "Scalable recognition with a vocabulary tree," *in Proc. CVPR*, pp. 2161-2168, 2006.

[33] 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.

[34] 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.

[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] 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.

[39] 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.

[40] 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.

[41] 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.

[42] R. Kwitt, and A. Uhl, "Lightweight probabilistic texture retrieval," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 241-253, Jan. 2010.

[43] R. Kwitt, and A. Uhl, "Image similarity measurement by Kullback-Leibler divergences between complex wavelet subband statistics for texture retrieval," in *Proc. 15th IEEE ICIP*, pp. 933-936, Oct. 2008.

[44] N. E. Lasmar, and Y. Berthoumieu, "Gaussian copula multivariate modeling for texture image retrieval using wavelet transforms," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2246-2261, May. 2014.

[45] S. E. Robertson, and S. Walker, "Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval," in *Proc. 17th Annu. Int. ACM SIGIR Conf. Res. Develop. Inform. Retr.*, 1994, pp. 232-241.

[46] H. Jegou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 1169-1176.

[47] L. Zheng, S. Wang, and Q. Tian, "Lp-Norm IDF for scalable image retrieval," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3604-3617, Aug. 2014.

[48] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 1470-177, 2012.

[49] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. Han, "Contextual weighting for vocabulary tree based image retrieval," in *Proc. IEEE Int. Conf. Comp. Vis. (ICCV)*, pp. 209-216, 2011.

[50] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 304-317, 2013.

[51] Z. Liu, H. Li, W. Zhou, R. Zhao, and Q. Tian, "Contextual hashing for large-scale image search," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1606-1614, Apr. 2014.

[52] A. Porebski, N. Vandenbroucke and L. Macaire, "Haralick feature extraction from LBP images for color texture classification," *Proc. 1st Workshops Image Process. Theory, Tools Appl. (IPTA)*, pp. 1-8, 2008.

[53] G. H. Liu, Z. Li, Z. Lei, and Y. Xu, "Image retrieval based on micro-structure descriptor," *Pattern Recognit. Lett.*, vol. 44, no. 9, pp. 2123-2133, 2011.

[54] F. Bianconi, A. Fernandez, E. Gonzalez, D. Caride and A. Calvino, "Rotation-invariant colour texture classification through multilayer CCR," *Pattern Recognit. Lett.*, vol. 30, no. 8, pp. 765-773, 2009.

[55] G. Paschos and M. Petrou, "Histogram ratio features for color texture classification," *Pattern Recognit. Lett.*, vol. 24, no. 1–3, pp. 309-314, 2003.

[56] M. A. Hoang, J.-M. Geusebroek and A. W. M. Smeulders, "Color texture measurement and segmentation," *Signal Process.*, vol. 85, no. 2, pp. 265-275, 2005.

[57] M. A. Hoang and J. M. Geusebroek, "Measurement of color texture," in *Proc. Workshop Texture Anal. Mach. Vis.*, pp. 73-76, 2002.

[58] J. J. Junior, P. C. Cortex, and A. R. Backes, "Color texture classification using shortest paths in graphs," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3751-3761, Sept. 2014.

[59] B. J. L. Campana, and E. J. Keogh, "A compression-based distance measure for texture," *Statist. Anal. Data Mining*, vol. 3, no. 6, 2010.

[60] T. Guha, and R. K. Ward, "Image similarity using sparse representation and compression distance," *IEEE Trans. Multimedia*, vol. 16, no. 4, pp. 980-987, Jun. 2014.

[61] D. Lowe, "Distinctive image feature from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91-110, Nov. 2004.

[62] T. Ojala. K. Valkealahti, E. Oja, and M. Pietikainen, "Texture discrimination with multidimensional distributions of signed gray-level differences," *Pattern Recognit.*, vol. 34, no. 3, pp. 727-739, 2001.

[63] J. Chen et. al. "WLD: A robust local image descriptor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1705-1720, Sep. 2010.

[64] A. Satpathy, X. Jiang, and H. L. Eng, "LBP-based edge-texture features for object recognition," *IEEE Trans. Image Process.*, vol. 23, no. 5, May 2014.

[65] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Int. Conf. Comp. Vis. Pattern Recognit.*, vol. 2, pp. 2169-2178, Jun. 2006.

[66] A. Oliva, and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *Int. J. Comp. Vision*, vol. 43, no. 3, pp. 145-175, 2001.

[67] J. Yangqing, H. Chang, and T. Darell, "Beyond spatial pyramids: receptive field learning for pooled image features," in *IEEE Conf. Comp. Vision Pattern Recognit.* (CVPR), pp. 3370-3377, 2012.

[68] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40, no. 2, 2008.

[69] SIPI-USC Brodatz texture image database, online available on http:// http://sipi.usc.edu/database/database.php?volume=textures.

[70] MIT-Vision Texture (VisTex) image database, online available on http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html.

[71] G. J. Burghouts and J. M. Geusebroek, "Material-specific adaptation of color invariant features", *Pattern Recognit. Lett.*, vol. 30, 306-313, 2009.

[72] Herve Jegou, Matthijs Douze and Cordelia Schmid "Hamming embedding and weak geometry consistency for large scale image search," in *Proc. 10th European Conf. Comp. Vision*, October, 2008.

[73] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *IEEE Conf. Comp. Vision Pattern Recognit. (CVPR)*, vol. 2, pp. 2161-2168, June 2006.

[74] A. R. Backes, D. Casanova, and O. M. Bruno, "Color texture analysis based on fractal descriptors," *Pattern Recognit.*, vol. 45, no. 5, pp. 1984-1992, 2012.

[75] Outex texture image database, online available on http://www.outex.oulu.fi/index.php?page=outex_home.

[76] KTH-TIPS texture image database, online available on http://www.nada.kth.se/cvap/databases/kth-tips/index.html.

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 Image/video processing, multimedia signal processing, computer vision, and biometrics.

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 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 be/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

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/TIP.2014.2372619, IEEE Transactions on Image Processing

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) form 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.