
Performance Evaluation of Error Diffusion Block Truncation Coding Feature for Color Image Retrieval

Jing-Ming Guo and Heri Prasetyo

This chapter presents a performance comparison of the Error Diffusion Block Truncation Coding (EDBTC) feature for color image retrieval and classification. In these approaches, the image retrieval and classification employ the feature descriptor derived from the EDBTC compressed data stream. Firstly, a color image is decomposed using EDBTC scheme to produce two new image representations, namely color quantizer and bitmap image. Two image feature descriptors, called Color Histogram Feature (CHF) and Bit Pattern Histogram Feature (BHF), can be subsequently generated from the EDBTC color quantizer and its corresponding bitmap image, respectively, without performing the decoding process. The similarity degree between two images is simply measured with the similarity distance score of their feature descriptor. In this chapter, the effectiveness of EDBTC feature descriptor is quantitatively examined and compared in the RGB color space as well as in YCbCr color channel. As reported in experimental result, the proposed method outperforms the former existing schemes in

Jing-Ming Guo and Heri Prasetyo
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

the image retrieval and classification tasks. It has shown that the EDBTC performs well in image compression domain, in addition, it also offers an effective and efficient way for performing image retrieval and classification.

4.1 Introduction

Content-Based Image Retrieval (CBIR) offers a convenient way to browse and search the desired image in the huge image database. The CBIR employs the image features of visual content to represent and index the image in database. These features can be color, texture, shape, etc. The feature choice depends on the user's preference or is decided by the expert-system. Finding a single best representative feature of an image is very difficult because of the fact that the photographer may take several images under different conditions such as different lighting sources, various view angles, different illumination changes, etc. Developing an effective and efficient image feature descriptor becomes a challenging task for CBIR system to achieve a high image retrieval performance. Many attempts and researches have been devoted to improve the retrieval accuracy in the CBIR system. One of these efforts is employing an image feature descriptor derived from the compressed data stream for CBIR task. As opposite to the classical approaches which extract an image descriptor from the original image, this image retrieval scheme directly generates image feature from the compressed data stream without firstly performing the decoding process. This type of image retrieval aims on reducing the computation time in feature extraction/generation since most of the multimedia contents and images are already converted into the compressed format before they are recorded in any storage devices.

A new CBIR system has been proposed [16] in which the image feature descriptor is simply derived from the compressed data stream. This new approach indexes the color images using the feature descriptor extracted from the Error-Diffusion Block Truncation Coding (EDBTC). The EDBTC is an improved version of Block Truncation Coding (BTC) [8, 53] which is an efficient image compression technique. The EDBTC has been demonstrated to yield a promising result on several image processing applications such as image compression [11], image watermarking [12, 15, 41], inverse halftoning [30], data hiding [14], image security [13], halftone image classification [29], CBIR system [16], etc. The EDBTC produces two color quantizers and a single bitmap image on the encoding stage. In the CBIR system, an image feature descriptor is directly derived from the EDBTC color quantizers and bitmap image in the compressed domain by involving the Vector Quantization (VQ). The two features are generated in the CBIR system, namely Color Histogram Feature (CHF) and Bit Pattern Histogram Feature (BHF), to measure the similarity criterion between a query image and a set target images stored in database. The CHF and BHF are computed from the VQ-indexed color quantizer and VQ-indexed bitmap image, respectively. The similarity distance computed from CHF and BHF can be further utilized for performing the similarity matching between two images. As reported in [16], the image retrieval with EDBTC feature offers lower feature dimensionality compared to the former BTC-based image retrieval scheme, and at the same time, outperforms the former BTC-based CBIR system. The image retrieval system with EDBTC feature also performs better compared

to the former competing schemes on natural and textural images.

Triggerred by efficiency and successfulness of the image retrieval system using EDBTC feature, we propose to develop a new way to further improve the EDBTC image retrieval performance for this book chapter. Some extensions for the EDBTC image retrieval system proposed for this book chapter can be highlighted as follows: 1) extending the EDBTC image feature descriptor for the other color space, 2) proposing a new feature descriptor by combining the two color quantizer to further reduce the feature dimensionality, 3) investigating the effect of different similarity distance for overall image retrieval performance, 4) examining suitable similarity weighting constants to achieve the highest image retrieval performance, and 5) performing image retrieval and classification over various natural and textural image databases in the grayscale and color space. The rest of this chapter is organized as follows. A brief introduction of EDBTC is provided in Section 4.2. Section 4.3 presents the proposed EDBTC image retrieval including the image feature generation and similarity distance computation. Extensive experimental results are reported in Section 4.4. Finally, the conclusions are drawn at the end of this chapter.

4.2 Error Diffusion Block Truncation Coding for Color Image

This section introduces an EDBTC image compression for the color image. Herein, the compression is presented for RGB color image. However, this method can be extended into the other color spaces such as YCbCr, or the other color channels. In a simple way, the EDBTC compresses an image patch in RGB color space into a new representation, i.e. two color quantizer of the same size as a single color pixel and its corresponding bitmap image of the same size as original image patch. The two EDBTC color quantizers are simply set with the *min* and *max* pixel values found in an image patch. On the other hand, the EDBTC employs the error kernel to generate bitmap image. The EDBTC method produces better image quality compared to that of the classical BTC approach as it has been reported and deeply investigated in [11, 12].

Suppose a color image of size $M \times N$ is partitioned into multiple non-overlapping image patches of size $m \times n$. Let $f(x, y) = \{f_R(x, y), f_G(x, y), f_B(x, y)\}$ be an image patch, for $x = 1, 2, \dots, m$ and $y = 1, 2, \dots, n$. The inter-band average value of image patch $f(x, y)$ can be simply computed as:

$$\bar{f}(x, y) = \frac{1}{3}(f_R(x, y) + f_G(x, y) + f_B(x, y)) \quad (4.1)$$

where $f_R(x, y)$, $f_G(x, y)$, and $f_B(x, y)$ denote the image pixels in the red, green, and blue color channels, respectively. Figure 4.1 illustrates the schematic diagram of the EDBTC compression for color image.

The EDBTC produces a single bitmap image $bm(x, y)$ of the same size as image patch by incorporating error kernel. In this chapter, we employ Floyd-Steinberg error kernel for generating bitmap image. For performing the EDBTC thresholding, we firstly compute the minimum, maximum, and mean value of the inter-band average pixels as follows:

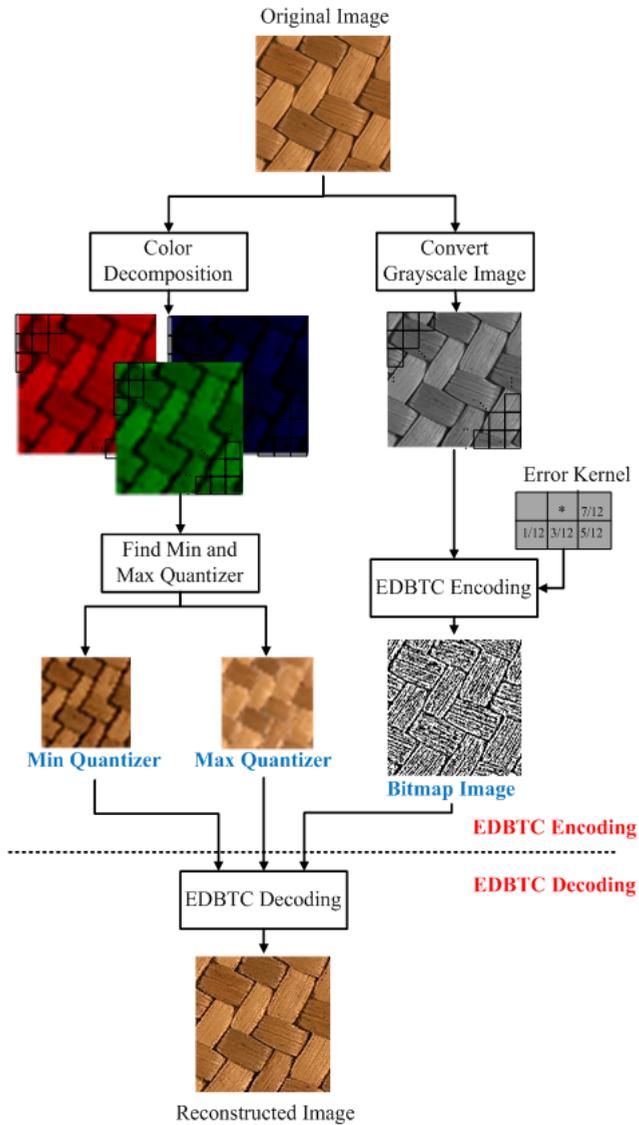


Figure 4.1: Schematic diagram of EDBTC processing for color image.

$$x_{min} = \min_{\forall x,y} \bar{f}(x,y) \quad (4.2)$$

$$x_{max} = \max_{\forall x,y} \bar{f}(x,y) \quad (4.3)$$

$$\bar{x} = \sum_{x=1}^m \sum_{y=1}^n \bar{f}(x,y) \quad (4.4)$$

The bitmap image $bm(x,y)$ is subsequently produced using the following thresholding method

$$bm(x,y) = \begin{cases} 1, & \text{if } \bar{f}(x,y) \geq \bar{x} \\ 0, & \text{if } \bar{f}(x,y) < \bar{x} \end{cases} \quad (4.5)$$

The intermediate value $o(x,y)$ is also generated at the same time with the bitmap image generation as follows

$$o(x,y) = \begin{cases} x_{max}, & \text{if } bm(x,y) = 1 \\ x_{min}, & \text{if } bm(x,y) = 0 \end{cases} \quad (4.6)$$

The EDBTC residual quantization error can be simply calculated as follow

$$e(x,y) = \bar{f}(x,y) - o(x,y) \quad (4.7)$$

The EDBTC processes an image pixel in a consecutive way in which one pixel is only processed once, and the residual quantization error is then diffused and accumulated into its unprocessed neighboring pixels. The unprocessed pixel value $\bar{f}(x,y)$ is updated using

$$\bar{f}(x,y) = \bar{f}(x,y) + e(x,y) \star \epsilon \quad (4.8)$$

where ϵ and \star denote the error kernel and convolution operation, respectively.

Two EDBTC color quantizers are simply set with the minimum and maximum pixel values found in an image patch as follows:

$$q_{min}(i,j) = \left\{ \min_{\forall x,y} f_R(x,y), \min_{\forall x,y} f_G(x,y), \min_{\forall x,y} f_B(x,y) \right\} \quad (4.9)$$

$$q_{max}(i,j) = \left\{ \max_{\forall x,y} f_R(x,y), \max_{\forall x,y} f_G(x,y), \max_{\forall x,y} f_B(x,y) \right\} \quad (4.10)$$

where q_{min} and q_{max} are the EDBTC *min* and *max* quantizer, respectively. The EDBTC encoder module sends the two color quantizer and bitmap image into decoder side via transmission channel. For decoding this EDBTC data stream, the decoder simply replaces the bitmap image of having value 1 with the max quantizer, vice versa. Figure 4.2 shows the EDBTC image reconstructed under the RGB and YCbCr color space. In the YCbCr color space, the bitmap image is simply generated from the Y color value, while the two color quantizers are obtained from all color channels. The EDBTC offers an efficient and effective way in image compression with lower computational complexity.

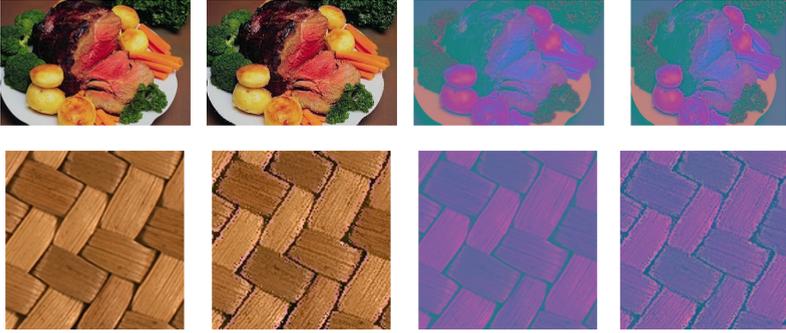


Figure 4.2: Image quality comparison of original image (first column) and EDBTC reconstruction (second column) in RGB color space. The third and fourth column are the original and EDBTC image reconstruction, respectively, in YCbCr color space.

4.3 EDBTC Image Indexing

This section presents the proposed image retrieval scheme using EDBTC feature. Herein, the proposed method is detail explained in RGB color space which can be easily extended for the other color spaces such as YCbCr, etc. Figure 4.3 illustrate the proposed image retrieval system using EDBTC image feature. Firstly, an image is decomposed using EDBTC to obtain the color quantizers and its corresponding bitmap image. The image feature is subsequently extracted from these EDBTC color quantizers and bitmap image to yield CHF and BHF, respectively.

4.3.1 Color and Bit Pattern Indexing

Let $C_{min} = \{c_1^{min}, c_2^{min}, \dots, c_{N_{min}}^{min}\}$ and $C_{max} = \{c_1^{max}, c_2^{max}, \dots, c_{N_{max}}^{max}\}$ be the color codebook from the EDBTC *min* and *max* quantizer, respectively. The N_{min} and N_{max} are the color codebook size for *min* and *max* quantizer, respectively. While the color quantizers are in RGB color space, then C_{min} and C_{max} are also in the same color space. Let $C = \{c_1, c_2, \dots, c_{N_c}\}$ be the color codebook obtained by concatenating the min and max quantizer as a single vector, i.e. $[q_{min}; q_{max}]$. The N_c denotes the color codebook size of C . These color codebooks (C_{min} , C_{max} , and C) can be trained by means of Vector Quantization (VQ) using the *min* and *max* quantizer as a training set.

Let $B = \{B_1, B_2, \dots, B_{N_b}\}$ be the bit pattern codebook of size N_b . This bit pattern codebook can be generated from a set of EDBTC bitmap images with soft centroid method. All bitmap images are treated as non-integer value during VQ training process. The hard thresholding is conducted at the end of VQ process to force all trained data into the binary form. Figure 4.4 shows an example of bit pattern codebook of size $N_b = \{16, 32, 64, 128\}$.

For obtaining the EDBTC image feature, the min and max quantizer are firstly

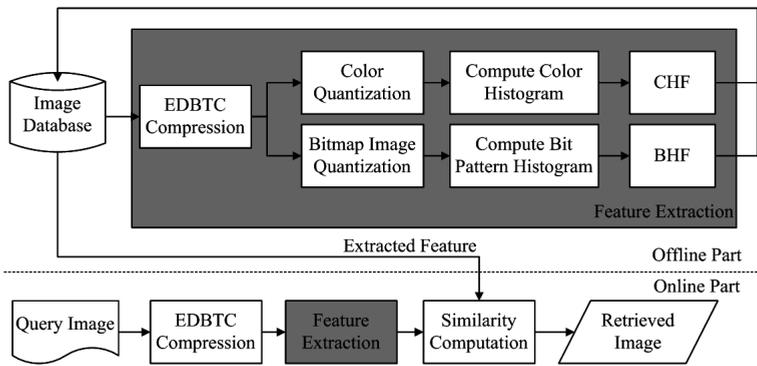


Figure 4.3: Schematic diagram of the proposed image retrieval framework.

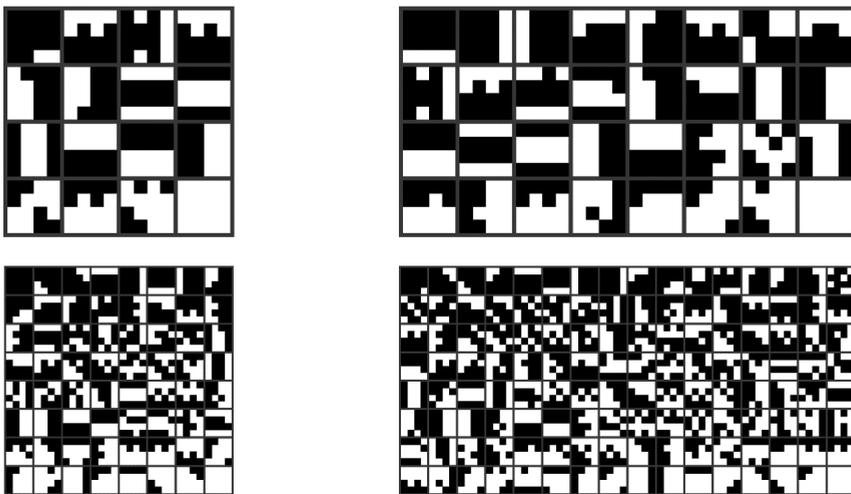


Figure 4.4: Schematic diagram of the proposed image retrieval framework.

indexed by incorporating the color codebook. Given the *min* and *max* color codebook (C_{min} and C_{max}), the indexing process of EDBTC *min* and *max* quantizer can be defined as follows:

$$\tilde{I}_{min}(i, j) = \arg \min_{k=1,2,\dots,N_{min}} \|q_{min}(i, j), c_k^{min}\|_2^2 \quad (4.11)$$

$$\tilde{I}_{max}(i, j) = \arg \min_{k=1,2,\dots,N_{max}} \|q_{max}(i, j), c_k^{max}\|_2^2 \quad (4.12)$$

for all $i = 1, 2, \dots, \frac{M}{m}$, and $j = 1, 2, \dots, \frac{N}{n}$. Where (i, j) denotes the index of image block. Given the color codebook C , the indexing process of concatenated *min* and *max* quantizer $[q_{min}; q_{max}]$ can be performed as follows:

$$\tilde{I}(i, j) = \arg \max_{k=1,2,\dots,N_c} \|[q_{min}(i, j); q_{max}(i, j)], c_k\|_2^2 \quad (4.13)$$

The stacking process of min and max quantizer reduces the computational time in the color indexing process and produces the lower feature dimensionality.

Given the bit pattern codebook B , the indexing process of EDBTC bitmap image $bm(i, j)$ is simply computed as follows:

$$\tilde{b}(i, j) = \arg \min_{k=1,2,\dots,N_b} \delta_H \{bm(i, j), B_k\} \quad (4.14)$$

for all image blocks $i = 1, 2, \dots, \frac{M}{m}$, and $j = 1, 2, \dots, \frac{N}{n}$. Where $\delta_H \{\cdot, \cdot\}$ denotes the Hamming distance between two binary patterns (vectors). The indexing process of color and bit pattern reduce the required bits in the EDBTC image compression. The entropy coding with lossless or lossy approaches can be further applied to reduce the required bits before transmitting into the decoder module.

4.3.2 Color Histogram Feature (CHF)

An image feature, namely Color Histogram Feature (CHF), can be simply obtained from the indexed EDBTC color quantizers. The CHF adequately describes an image brightness and its color distribution. This process can be viewed as computing the occurrence of specific indexed color quantizer in the whole image to produce a histogram. The CHF_{min} and CHF_{max} are the image feature descriptor derived from the min and max quantizer, respectively. While the CHF denotes the image feature descriptor generated by concatenating the min and max quantizer. Figure 4.5 illustrates an example of CHF computation. The CHF_{min} , CHF_{max} , and CHF can be formally described as follows:

$$CHF_{min}(k) = Pr \left\{ \tilde{I}_{min}(i, j) = k \mid i = 1, 2, \dots, \frac{M}{m}; j = 1, 2, \dots, \frac{N}{n} \right\}, \quad (4.15)$$

$$CHF_{max}(k) = Pr \left\{ \tilde{I}_{max}(i, j) = k \mid i = 1, 2, \dots, \frac{M}{m}; j = 1, 2, \dots, \frac{N}{n} \right\}, \quad (4.16)$$

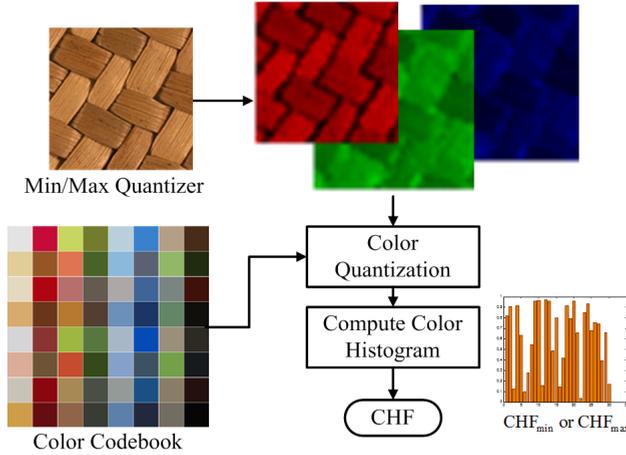


Figure 4.5: Illustration of CHF computation.

$$CHF(k) = Pr \left\{ \tilde{I}(i, j) = k \mid i = 1, 2, \dots, \frac{M}{m}; j = 1, 2, \dots, \frac{N}{n} \right\}, \quad (4.17)$$

where $k = 1, 2, \dots, N_{min}$ and $k = 1, 2, \dots, N_{max}$ for CHF_{min} and CHF_{max} , respectively. While CHF simply requests $k = 1, 2, \dots, N_c$. The feature dimensionality of CHF_{min} , CHF_{max} , and CHF are identical to the color codebook size, i.e. N_{min} , N_{max} , and N_c , respectively.

4.3.3 Bit Pattern Histogram Feature (BHF)

The other image feature descriptor which can be derived from EDBTC bitmap image is called as Bit Patter Histogram Feature (BHF). This feature is similar to the CHF which tabulates the occurrence of a specific indexed bit pattern in whole image. This feature represents an image textural information as well as visual pattern, line, and edge, etc. Figure 4.6 shows an illustration of BHF computation. The BHF of an image can be formally defined as follow

$$BHF(k) = Pr \left\{ \tilde{b}(i, j) = k \mid i = 1, 2, \dots, \frac{M}{m}; j = 1, 2, \dots, \frac{N}{n} \right\}, \quad (4.18)$$

for all $k = 1, 2, \dots, N_b$. The feature dimensionality of BHF is identical to the bit pattern codebook size, i.e. N_b . The prospective readers are suggested to refer [16] for detail explanation on effect of color and bit pattern codebook size in the image retrieval task.

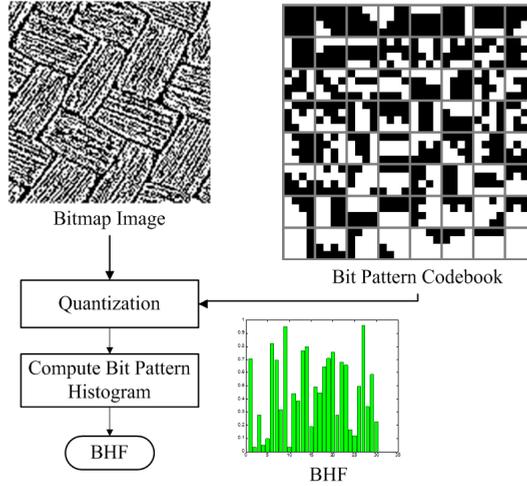


Figure 4.6: Illustration of BHF computation.

4.3.4 Image Retrieval with EDBTC Feature

The similarity distance computation plays an important role in the overall performance of image retrieval and classification task. Different choice of similarity distance influences the performance accuracy. The similarity degree between two images can be measured using similarity distance score between their descriptor in which smaller distance indicates more similarity. In our proposed method, the choice of similarity distance can be determined by experiment since the CHF and BHF are in the different modalities. In image retrieval system, a set of retrieved images are returned to the user in ascending order based on their similarity distance scores. Several similarity distance can be formally defined as follows to measure similarity degree between the query and target image:

- L_1 distance:

$$\delta(query, target) = \alpha_1 \sum_{k=1}^{N_{min}} |CHF_{min}^{query}(k) - CHF_{min}^{target}(k)| + \alpha_2 \sum_{k=1}^{N_{max}} |CHF_{max}^{query}(k) - CHF_{max}^{target}(k)| + \alpha_3 \sum_{k=1}^{N_b} |BHF^{query}(k) - BHF^{target}(k)|, \quad (4.19)$$

- L_2 distance:

$$\delta(query, target) = \left[\alpha_1 \sum_{k=1}^{N_{min}} (CHF_{min}^{query}(k) - CHF_{min}^{target}(k))^2 + \alpha_2 \sum_{k=1}^{N_{max}} (CHF_{max}^{query}(k) - CHF_{max}^{target}(k))^2 + \alpha_3 \sum_{k=1}^{N_b} (BHF^{query}(k) - BHF^{target}(k))^2 \right]^{1/2}, \quad (4.20)$$

- χ^2 distance:

$$\begin{aligned} \delta(query, target) = & \alpha_1 \sum_{k=1}^{N_{min}} \left(\frac{CHF_{min}^{query}(k) - CHF_{min}^{target}(k)}{CHF_{min}^{query}(k) + CHF_{min}^{target}(k) + \varepsilon} \right)^2 + \\ & \alpha_2 \sum_{k=1}^{N_{max}} \left(\frac{CHF_{max}^{query}(k) - CHF_{max}^{target}(k)}{CHF_{max}^{query}(k) + CHF_{max}^{target}(k) + \varepsilon} \right)^2 + \\ & \alpha_3 \sum_{k=1}^{N_b} \left(\frac{BHF^{query}(k) - BHF^{target}(k)}{BHF^{query}(k) + BHF^{target}(k) + \varepsilon} \right)^2, \end{aligned} \quad (4.21)$$

- Fu distance:

$$\delta(query, target) = \frac{L_2 \text{ distance}}{|CHF_{min}| + |CHF_{max}| + |BHF|}, \quad (4.22)$$

- Modified Canberra distance:

$$\begin{aligned} \delta(query, target) = & \alpha_1 \sum_{k=1}^{N_{min}} \frac{|CHF_{min}^{query}(k) - CHF_{min}^{target}(k)|}{CHF_{min}^{query}(k) + CHF_{min}^{target}(k) + \varepsilon} + \\ & \alpha_2 \sum_{k=1}^{N_{max}} \frac{|CHF_{max}^{query}(k) - CHF_{max}^{target}(k)|}{CHF_{max}^{query}(k) + CHF_{max}^{target}(k) + \varepsilon} + \\ & \alpha_3 \sum_{k=1}^{N_b} \frac{|BHF^{query}(k) - BHF^{target}(k)|}{BHF^{query}(k) + BHF^{target}(k) + \varepsilon}, \end{aligned} \quad (4.23)$$

where $\{\alpha_1, \alpha_2, \alpha_3\}$ denotes the similarity weighting constants indicating the percentage contribution of the CHF and BHF in the similarity distance computation. Higher value of similarity weighting constants indicates the higher contribution of image feature descriptor usage in similarity distance computation. A small number ε is added into denominator to avoid the mathematical division error.

4.4 Experimental Results

This section reports some extensive experiments for demonstrating the effectiveness and usefulness of the proposed image feature descriptor for content-based image retrieval and classification under the textural and natural images in the grayscale or color space. In this research, two image feature descriptors and two color spaces are involved for performing the retrieval and classification task. Two features are the Color Histogram Feature (CHF) and Bit Pattern Histogram Feature (BHF), while the Red-Green-Blue (RGB) and YCbCr color spaces are investigated for the CBIR system and image classification. The EDBTC-1 refers to the proposed method which employs the CHF_{min} , CHF_{max} , and BHF as feature descriptor. Herein, CHF_{min} and CHF_{max} are the feature descriptor obtained from the min and max quantizer, respectively. Thus, the feature dimensionality of EDBTC-1 is $N_{min} + N_{max} + N_b$, where N_{min} , N_{max} , and N_b denoting the color codebook size of min and max quantizer, and the bit pattern codebook size, respectively. To further reduce the feature dimensionality, the EDBTC-2 simply utilizes the proposed feature descriptor CHF and BHF. Where, CHF is obtained from the histogram of VQ-indexed by concatenating of min and max quantizer. The feature dimensionality of EDBTC-2 is $N_c + N_b$, where the N_c and N_b denote the color and bit pattern codebook size, respectively. In case of YCbCr color space, an image in RGB color space is firstly converted into YCbCr channel. The

EDBTC decomposes this image for obtaining the two color quantizers in the YCbCr space. The EDBTC generates the bitmap image from the Y color channel.

In our proposed scheme, all images in database are firstly decomposed using EDBTC encoding scheme to yield the two color quantizers and its corresponding bitmap image. Subsequently, the image feature descriptor of all images are derived from these EDBTC data stream using EDBTC-1 or EDBTC-2 methods. These image feature are then stored in database for later usage. All images in database are regarded as a set of target images. This process can be performed in offline manner. While a query image is turned by a user, an identical feature extraction procedure as conducted for the target image, is applied for this query image. The similarity criterion between the query and target image is simply measured with the similarity distance computation based on their feature descriptor. At the end of image retrieval, a set of retrieved images are returned to the user in ascending order sorted based on their similarity distance values.

The identical strategy in image retrieval system can also be adopted for the image classification task in which the nearest neighbor classifier is incorporated for determining the class label of a given input image. In all experiments, the EDBTC image block size is set at 4×4 .

4.4.1 Performance Measurement

In this subsection, two quantitative metrics are introduced to examine the successfulness and effectiveness of the proposed image feature descriptor in the image retrieval domain. The image retrieval performance of the proposed method is accessed in terms of Average Precision Rate (APR) and Average Retrieval Rate (ARR). These two values indicate the percentage of relevant images returned by the system in a specific number of retrieved images L . The APR and ARR are formally defined as

$$APR = \frac{1}{N_t L} \sum_{q=1}^{N_t} n_q(L), \quad (4.24)$$

$$ARR = \frac{1}{N_t N_R} \sum_{q=1}^{N_t} n_q(N_R), \quad (4.25)$$

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

The n_q denotes the number of relevant images against a query image q . A higher value of APR and ARR exhibits the better image retrieval performance.

The performance of the proposed method in image classification task is simply examined using the proportion of correct classification (accuracy rate) in which the total correct classification is computed among all the testing images. The image classification utilizes an identical procedure as in the image retrieval task for deciding a class label of testing image. A nearest neighbor classifier is employed by performing similarity matching between the testing image and training images. A classifier decides

Table 4.1: Summary of image databases used in the experiment.

Database Name	Image Size	Number of Classes	Number of Images per Class	Total Images
Corel [52]	384×256	10	100	1000
Brodatz-1856 [47]	128×128	116	16	1856
Vistex-640 [35]	128×128	40	16	640
Stex [48]	128×128	476	16	7616
ALOT [3]	192×128	250	16	4000
Vistex-864 [35]	128×128	54	16	864
USPTex [1]	128×128	191	12	2292
Outex TC00013 [39]	128×128	68	20	1360
KTH-TIPS [23]	200×200	10	81	810
KTH-TIPS 2A [23]	200×200	11	vary	4395

a class label of a testing image based on the class label of the training images which is returned by the system based on their similarity distance score.

4.4.2 Experimental Setup

In this experiment, several image databases are involved for assessing the effectiveness and usability of the proposed image feature descriptor. The image databases include the natural and textural images in the grayscale as well as in color spaces. We employ two color spaces, i.e. RGB and YCbCr, for performing comparison of the proposed image feature descriptor. Table 4.1 gives a summary of the image databases used in this experiment. The image databases consist of various image sizes and different database conditions such as different number of class, total images, and various number of image in each class, etc. Figure 4.7 shows some image samples of each image database. The database names are given from the first to the last row as Corel, Brodatz, Vistex, Stex, ALOT, USPTex, Outex TC00013, KTH-TIPS, and KTH-TIPS 2A. Various image databases give different image content and appearance. The Brodatz image database is in the grayscale mode, while the other databases are in the RGB color space. For performing image retrieval in YCbCr color space, all images in database are firstly converted from the RGB color space into the YCbCr color channel for subsequent feature extraction process. For assessing the superiority of the proposed method compared to the former schemes in image retrieval task, the performance of the proposed image feature is examined in terms of APR and ARR using the Corel, Brodatz, Vistex-640, Stex, and ALOT image databases. In addition, the performance of the proposed method is compared to the former schemes in image classification task in terms of classification rate under Vistex-640, USPTex, Outex TC00013, KTH-TIPS, and KTH-TIPS 2A image databases. The experimental setting of the proposed method is maintained identically to the former schemes including the nearest neighbor condition, fold cross validation, the number of retrieved image, etc., to make a fair comparison against the other competing schemes.

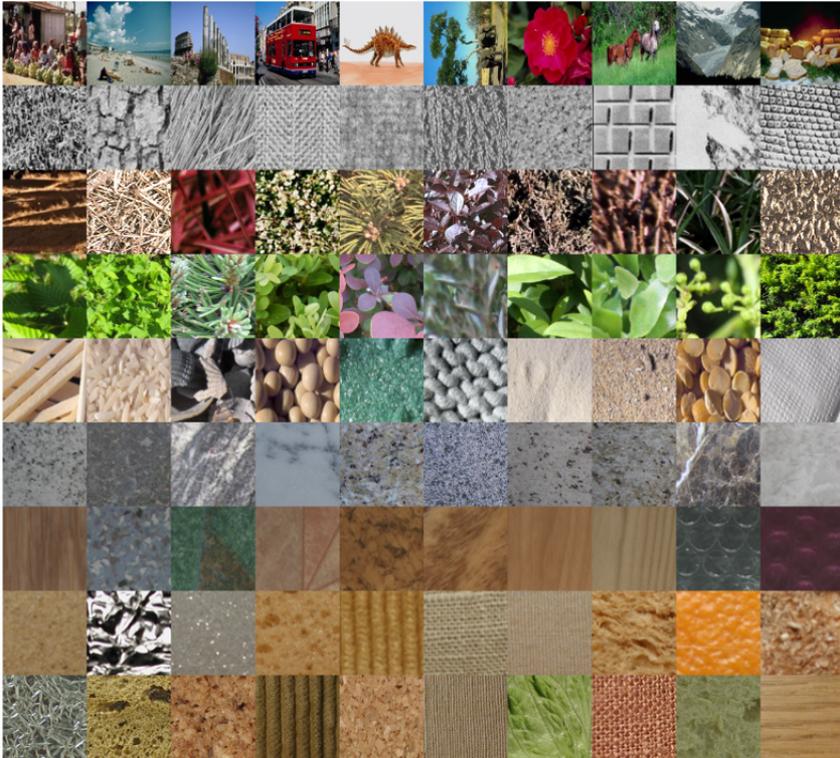


Figure 4.7: Image samples over various image databases. The database name from the first to last row: Corel, Brodatz, Vistex, Stex, ALOT, USPTex, Outex TC00013, KTH-TIPS and KTH-TIPS 2A.

4.4.3 Example of Image Retrieval System

This sub-section illustrates the usability and correctness of the proposed image retrieval system using our proposed feature descriptor. The correctness of the proposed feature descriptor is simply judged based on the similarity content and appearance between the query image and target images. Two images are regarded as visually similar while these two images own the similarity to the color, shape, textural information or the other criterions. Figure 4.8 shows an example of the image retrieval result over various image databases using EDBTC-1 in RGB color space. In this experiment, the color and bit pattern codebook size for all image databases (except the Stex and ALOT image databases) are chosen as $N_{min} = N_{max} = N_b = 64$. The Stex and ALOT image database employ the color and bit pattern codebook size as $N_{min} = N_{max} = N_b = 16$. The similarity weighting constants are set as $\{\alpha_1 = \alpha_2 = \alpha_3 = 1\}$ indicating that all feature descriptors are involved in the similarity distance computation. The first column of Fig.(4.8) is the query image, while the subsequent images from left to right are a set of retrieved images in ascending order based on the similarity distance score. The image database from the first to the last row are Corel, Brodatz, Vistex, Stex, ALOT, USPTex, Outex TC-00013, KTH-TIPS, and KTH-TIPS 2A. As it can be seen from Fig.(4.8), the proposed feature descriptor performs well in the CBIR system which can retrieve a set of retrieved images with the similar content appearance. The proposed method offers a promising result in image retrieval task making it a good candidate as competing image feature descriptor.

4.4.4 Effect on Different Similarity Distance

This sub-section reports the effect of different similarity distance usage in CBIR system. Five similarity distance computation are considered in this experiment, i.e. L_1 , L_2 , χ^2 , Fu, and modified Canberra distance. We experimentally investigate the similarity distance computation over various distance metric in the image retrieval task. The experiments are conducted for all image databases except for Brodatz and KTH-TIPS 2A, since the Brodatz image database contains the grayscale image, whereas the KTH-TIPS 2A database consists of textural image with various number of image in each class. The image feature descriptor is derived using EDBTC-1 under the RGB and YCbCr color spaces. In this experiment, the color and bit pattern codebook size are set at $N_{min} = N_{max} = N_b = 64$ for all image databases (except Stex and ALOT image databases). The Stex and ALOT image databases simply utilize $N_{min} = N_{max} = N_b = 8$ for performing image retrieval task. Herein, the similarity weighting constants are chosen as $\{\alpha_1 = \alpha_2 = \alpha_3 = 1\}$. The number of retrieved image for Corel database is set at $L = 20$, while the other image databases exploit the number of retrieved image as identical with the number of images in each class. Table 4.2 delivers the performance of the proposed method under different similarity distance over various image databases in RGB color space as well as YCbCr color channel. The image retrieval performance using L_2 distance is identical to that of using the Fu distance. Since Fu distance is simply normalized version of L_2 distance and the sum of the proposed feature descriptor is 1 making the denominator of Fu distance are equivalent to 2 for all images. As it can be seen in this table, the modified Canberra distance

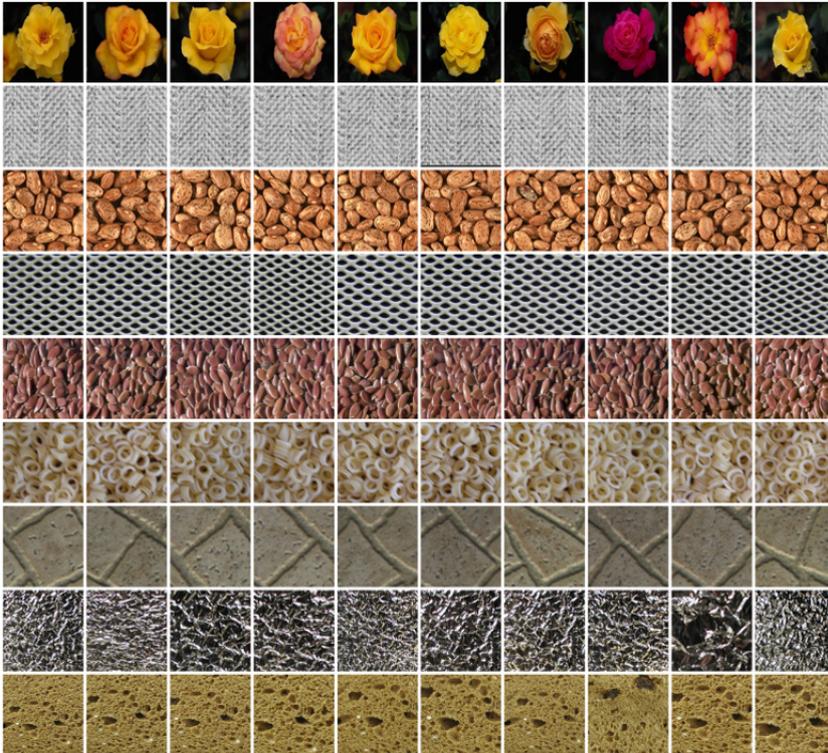


Figure 4.8: Example of image retrieval result over various image databases. An image in the first column is a query image where a set of subsequent images are retrieved result in ascending order from left to right. The database name from the first to last row: Corel, Brodatz, Vistex, Stex, ALOT, USPTex, Outex TC00013, KTH-TIPS and KTH-TIPS 2A.

gives the best image retrieval performance for almost all image databases (except KTH-TIPS image database). The image retrieval performance of RGB and YCbCr yields a comparable performance under different setting of similarity distances. The modified Canberra distance is more suitable compared to the other distance metrics for computing the similarity degree between the query and target image in the CBIR system.

4.4.5 Effect on Different Similarity Weighting Constants

An additional experiment were carried to further investigate the effect of similarity distance computation in the image retrieval task. In this experiment, we experimentally explore the effect of similarity weighting constants in the similarity distance calculation over various image databases under RGB and YCbCr color spaces. The similarity weighting constants indicate the percentage contribution of specific image feature descriptor in the similarity distance computation. Higher value of similarity weighting constants denotes higher contribution of specific image feature descriptor in the distance calculation. Herein, the similarity weighting constants are set at $\alpha_i = \{0.1, 0.2, \dots, 1\}$ for $i = 1, \dots, 3$. The modified Canberra distance is utilized in this experiment. The color, bit pattern codebook size, and number of retrieved images are identically set as in the previous sub-section. The optimized result is obtained by considering the combination of similarity weighting constants which gives the best APR value. The unoptimized result involves the similarity weighting constants as $\{\alpha_1 = \alpha_2 = \alpha_3 = 1\}$. Table 4.3 summarizes the image retrieval performance under the optimized and unoptimized setting of similarity weighting constants over various image databases in RGB and YCbCr color spaces. This table indicates that the image retrieval performance can be improved by setting the suitable similarity weighting constants. The optimized similarity weighting constants yield the better image retrieval result compared to that of unoptimized setting.

4.4.6 Performance on Image Retrieval System

This sub-section presents the performance comparison between the proposed method and former schemes in the image retrieval task. The comparison is conducted using five image databases, i.e. Corel, Brodatz, Vistex-640, Stex and ALOT image databases. The performance of image retrieval system is examined under APR value for Corel image database, and ARR value for the other image databases. The color and bit pattern codebook size are appropriately chosen to yield comparable feature dimensionality as in the former schemes. Herein, we utilize EDBTC-1 and EDBTC-2 for performing image retrieval strategy. EDBTC-1 employs CHF_{min} , CHF_{max} , and BHF. Whereas the EDBTC-2 utilizes CHF and BHF causing the feature dimensionality of EDBTC-2 is lower than EDBTC-1. For simplicity, the similarity weighting constants are set at $\{\alpha_1 = \alpha_2 = \alpha_3 = 1\}$. The number of retrieved images are set at $L = 20$ for Corel image database and identical to the number of images in each class for the other image databases. Table IV tabulates the image retrieval performance in terms of APR value between the proposed method and former schemes using Corel image database. The APR value of the proposed method is given as A/B indicating that A and B are

Table 4.2: Effect of different similarity distances over various image databases in terms of average precision rate.

Database Name	Distance Name	RGB Color Space	YCbCr Color Space
Corel	L_1	72.89	72.47
	L_2	64.31	63.61
	χ^2	74.69	75.00
	Fu	64.31	63.61
	Modified Canberra	76.87	76.72
Vistex-640	L_1	90.36	88.51
	L_2	83.13	80.97
	χ^2	91.63	91.47
	Fu	83.13	80.97
	Modified Canberra	91.67	91.60
Stex	L_1	41.57	40.91
	L_2	38.56	37.94
	χ^2	40.59	39.94
	Fu	38.56	37.94
	Modified Canberra	45.74	45.21
ALOT	L_1	45.95	45.55
	L_2	43.39	43.16
	χ^2	43.14	43.49
	Fu	43.39	43.16
	Modified Canberra	48.62	48.70
Vistex-864	L_1	84.53	82.49
	L_2	76.11	73.79
	χ^2	87.82	88.00
	Fu	76.11	73.79
	Modified Canberra	88.10	88.22
USPTex	L_1	63.11	62.85
	L_2	55.35	55.06
	χ^2	73.20	74.01
	Fu	55.35	55.06
	Modified Canberra	74.53	75.35
Outex TC00013	L_1	61.94	59.74
	L_2	57.52	54.54
	χ^2	64.59	65.16
	Fu	57.52	54.54
	Modified Canberra	66.10	66.48
KTH-TIPS	L_1	64.21	65.48
	L_2	59.69	61.81
	χ^2	66.21	64.81
	Fu	59.69	61.81
	Modified Canberra	64.76	63.34

Table 4.3: Effect of different similarity weight over various image databases in terms of average precision rate.

Database Name	RGB Color Space		YCbCr Color Space	
	Unoptimized	Optimized	Unoptimized	Optimized
Corel	76.87	79.07 $\{\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 0.3\}$	76.72	78.56 $\{\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 0.3\}$
Vistex-640	91.67	91.93 $\{\alpha_1 = 0.3, \alpha_2 = 0.3, \alpha_3 = 0.4\}$	91.60	91.92 $\{\alpha_1 = 0.7, \alpha_2 = 0.5, \alpha_3 = 0.8\}$
Stex	45.74	51.42 $\{\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 0.4\}$	45.21	50.82 $\{\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 0.4\}$
ALOT	48.62	52.36 $\{\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 0.4\}$	48.70	52.51 $\{\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 0.4\}$
Vistex-864	88.10	88.72 $\{\alpha_1 = 0.4, \alpha_2 = 0.5, \alpha_3 = 0.8\}$	88.22	88.63 $\{\alpha_1 = 0.3, \alpha_2 = 0.3, \alpha_3 = 0.4\}$
USPTex	74.53	75.63 $\{\alpha_1 = 0.8, \alpha_2 = 0.5, \alpha_3 = 1.0\}$	75.35	76.42 $\{\alpha_1 = 0.5, \alpha_2 = 0.3, \alpha_3 = 0.7\}$
Outex_TC00013	66.10	66.40 $\{\alpha_1 = 0.7, \alpha_2 = 0.4, \alpha_3 = 0.5\}$	66.48	66.87 $\{\alpha_1 = 0.6, \alpha_2 = 0.3, \alpha_3 = 0.4\}$
KTH-TIPS	64.76	66.67 $\{\alpha_1 = 0.5, \alpha_2 = 0.6, \alpha_3 = 0.2\}$	63.34	66.67 $\{\alpha_1 = 0.2, \alpha_2 = 0.5, \alpha_3 = 0.1\}$

Table 4.4: Comparisons among the proposed method and former schemes in terms of average precision rate for Corel image database.

Methods	Average
Gahroudi [9]	0.396
Jhanwar [21]	0.526
Huang [20]	0.532
Chiang [6]	0.533
Silakari [46]	0.560
Qiu [44]	0.595
Z.M. Lu [33]	0.600
Lu [32]	0.665
Yu [54]	0.717
Lin [27]	0.727
Poursistani [43]	0.743
EDBTC-1, 96 $\{N_{min} = N_{max} = N_b = 32\}$	0.740 / 0.738
EDBTC-2, 64 $\{N_c = N_b = 32\}$	0.750 / 0.749
EDBTC-1, 192 $\{N_{min} = N_{max} = N_b = 64\}$	0.769 / 0.767
EDBTC-2, 128 $\{N_c = N_b = 64\}$	0.767 / 0.776

image retrieval performance of using the RGB and YCbCr color space, respectively. As shown in this Table 4.4, the proposed method yields better performance compared to the former schemes using the Corel image database under the same or lower feature dimensionality. The feature dimensionality, color and bit pattern codebook size of the proposed scheme are also given in this table. The competing scheme [43] employs an image feature descriptor with dimensionality 256.

Table 4.5 and 4.6 show the performance comparison between the proposed method and former schemes under the textural image databases, i.e. Brodatz, Vistex-640, Stex and ALOT database. The feature dimensionality of the proposed method is set as comparable dimensionality with the former schemes to conduct a fair comparison. The ARR value in these tables is also delivered in form A/B indicating the ARR value using RGB and YCbCr color spaces, respectively. As it can be seen from this table, the proposed method offers a promising result in the textural image retrieval system

Table 4.5: Comparisons among the proposed method and former schemes in terms of ARR for Brodatz-1856 and Vistex-640 image databases.

Methods	Feature Dimension	Brodatz-1856	Vistex-640
GT with GGD&KLD [34]	$4 \times 6 \times 2 = 36$	17.19	76.57
DT-CWT [22]	$(3 \times 6 + 2) \times 2 = 40$	74.73	80.78
DT-RCWT [22]	$(3 \times 6 + 2) \times 2 = 40$	71.17	75.78
DT-CWT+DT-RCWT [22]	$2 \times (3 \times 6 + 2) \times 2 = 80$	77.75	82.34
LBP [37]	59	79.97	82.23
LTP [51]	$2 \times 59 = 118$	82.51	87.52
GLBP [17]	$6 \times 59 = 354$	75.21	84.74
LMEBP [49]	$8 \times 512 = 4096$	83.28	87.77
GLMEBP [49]	$3 \times 4 \times 512 = 6144$	82.01	87.93
LDP [55]	$4 \times 59 = 236$	79.91	87.27
LTrP [36]	$13 \times 59 = 767$	85.3	90.02
GLTrP [36]	$3 \times 13 \times 59 = 2301$	81.97	90.16
GLDP [55]	$6 \times 4 \times 59 = 1416$	79.11	87.51
GLTP [51]	$6 \times 2 \times 59 = 708$	78.75	80.98
MCMCM+DBPSP [50]	$9 \times 7 \times 7 + 6 = 477$	-	86.17
EDBTC-1	$48 \{N_{min} = N_{max} = N_b = 16\}$	66.71	85.64 / 85.72
EDBTC-2	$32 \{N_c = N_b = 16\}$	-	83.13 / 83.32
EDBTC-1	$96 \{N_{min} = N_{max} = N_b = 32\}$	70.14	89.56 / 89.63
EDBTC-2	$64 \{N_c = N_b = 32\}$	-	87.03 / 88.26
EDBTC-1	$192 \{N_{min} = N_{max} = N_b = 64\}$	80.74	91.67 / 91.60
EDBTC-2	$128 \{N_c = N_b = 64\}$	-	89.95 / 90.38
EDBTC-1	$384 \{N_{min} = N_{max} = N_b = 128\}$	90.09	92.45 / 92.63
EDBTC-2	$256 \{N_c = N_b = 128\}$	-	90.79 / 91.74

with the same or lower feature dimensionality. The proposed method is slightly inferior compared to that of the former scheme in [26] using Stex image database under the same feature dimensionality. However, the proposed method performs well for the other image databases.

4.4.7 Performance on Image Classification System

Additional experiments were conducted to further investigate and examine the proposed feature descriptor in the image classification system. The nearest classifier assigns a class label of testing image using the proposed EDBTC-1 or EDBTC-2 feature descriptor. The similarity weighting constants are simply set at $\{\alpha_1 = \alpha_2 = \alpha_3 = 1\}$. The dimensionality of the proposed descriptor is set as comparable to the other methods. Herein, the experimental setting is set as identical to the former schemes for a fair comparison. The image classification performance is assessed in terms of correct classification rate over all testing images.

Tables 4.7-4.9 summarize the image classification performance using Vistex-864, USPTex, Outex TC-00013, KTH-TIPS, and KTH-TIPS 2A image databases. Except for the Outex TC-00013 image database, the proposed method outperforms the former existing methods for the other image databases using the lower or identical feature dimensionality. The proposed method is slightly inferior compared to the other schemes under the Outex TC-00013 image database. From this experiment, it can be concluded

Table 4.6: Comparisons among the proposed method and former schemes in terms of ARR for Stex and ALOT image databases.

Methods	Feature Dimension	Stex	ALOT
Wbl-DT-CWT-1 scale [25]	$3 \times 2 = 6$	34.18	23.28
Wbl-DT-CWT-2 scale [25]	$6 \times 2 = 12$	45.38	33.56
Wbl-DT-CWT-3 scale [25]	$9 \times 2 = 18$	51.87	40.01
GG-DT-CWT-1 scale [25]	$3 \times 2 = 6$	34.82	23.86
GG-DT-CWT-2 scale [25]	$6 \times 2 = 12$	44.97	33.38
GG-DT-CWT-3 scale [25]	$9 \times 2 = 18$	50.75	39.33
GC-MWbl-DT-CWT-1 scale [24, 25]	$1 + 1 + 4 \times 4 = 18$	40.44	27.69
GC-MWbl-DT-CWT-2 scale [24, 25]	$1 + 1 + 4 \times 4 = 18$	49.78	37.68
GC-MWbl-DT-CWT-3 scale [24, 25]	$1 + 1 + 4 \times 4 = 18$	55.35	43.25
GC-MGG-DT-CWT-1 scale [26]	$1 + 1 + 4 \times 4 = 18$	46.42	30.58
GC-MGG-DT-CWT-2 scale [26]	$1 + 1 + 4 \times 4 = 18$	53.34	38.20
GC-MGG-DT-CWT-3 scale [26]	$1 + 1 + 4 \times 4 = 18$	57.24	43.06
EDBTC-1	$12 \{N_{min} = N_{max} = N_b = 4\}$	27.63 / 27.56	30.35 / 32.68
EDBTC-2	$8 \{N_c = N_b = 4\}$	20.55 / 20.77	25.09 / 24.43
EDBTC-1	$16 \{N_{min} = N_{max} = 4, N_b = 8\}$	34.59 / 34.89	38.25 / 37.51
EDBTC-2	$12 \{N_c = 4, N_b = 8\}$	29.06 / 29.64	32.20 / 31.66
EDBTC-1	$20 \{N_{min} = N_{max} = 8, N_b = 4\}$	40.29 / 39.41	45.22 / 45.18
EDBTC-2	$12 \{N_c = 8, N_b = 4\}$	32.20 / 35.98	40.37 / 42.93
EDBTC-1	$24 \{N_{min} = N_{max} = N_b = 8\}$	45.74 / 45.21	48.62 / 48.70
EDBTC-2	$16 \{N_c = N_b = 8\}$	39.66 / 44.75	45.26 / 48.13

Table 4.7: Classification performance among the proposed method and former schemes under Vistex 864, USPTex, and Outex TC-00013 image databases.

Methods	Feature Dimension	Vistex 864	USPTex	TC-00013
Average RGB	3	58.78	36.19	76.49
LBP+Haralick [42]	10	91.59	73.17	77.89
MSD [28]	72	82.07	51.29	49.83
Multilayer CCR [2]	640	94.74	82.08	68.50
HRF [40]	-	58.89	49.86	40.00
Gabor EEE [18, 19]	192	96.92	92.58	75.86
Shortest Graph [7]	96	87.62	66.71	88.06
EDBTC-1	$48 \{N_{min} = N_{max} = N_b = 16\}$	98.61 / 98.38	88.22 / 87.48	81.91 / 81.32
EDBTC-2	$32 \{N_c = N_b = 16\}$	96.53 / 96.88	80.72 / 85.25	70.88 / 79.41
EDBTC-1	$96 \{N_{min} = N_{max} = N_b = 32\}$	99.19 / 99.31	91.97 / 93.19	83.31 / 84.41
EDBTC-2	$64 \{N_c = N_b = 32\}$	98.96 / 98.50	89.09 / 90.49	79.19 / 80.81
EDBTC-1	$192 \{N_{min} = N_{max} = N_b = 64\}$	99.42 / 99.65	95.16 / 95.20	87.57 / 86.69
EDBTC-2	$128 \{N_c = N_b = 64\}$	98.61 / 99.19	92.23 / 92.84	77.87 / 81.32
EDBTC-1	$384 \{N_{min} = N_{max} = N_b = 128\}$	99.19 / 99.54	94.94 / 95.94	85.88 / 85.22
EDBTC-2	$256 \{N_c = N_b = 128\}$	98.96 / 99.31	92.06 / 92.89	78.24 / 79.78

Table 4.8: Classification performance among the proposed method and former schemes under KTH-TIPS image database.

Methods	Feature Dimension	KTH-TIPS
CK-1 [4]	-	86.00
Sparse [10]	-	84.50
EDBTC-1	$48 \{N_{min} = N_{max} = N_b = 16\}$	98.52 / 97.53
EDBTC-2	$32 \{N_c = N_b = 16\}$	96.17 / 96.05
EDBTC-1	$96 \{N_{min} = N_{max} = N_b = 32\}$	99.26 / 99.38
EDBTC-2	$64 \{N_c = N_b = 32\}$	98.52 / 99.01
EDBTC-1	$192 \{N_{min} = N_{max} = N_b = 64\}$	99.26 / 99.26
EDBTC-2	$128 \{N_c = N_b = 64\}$	98.52 / 98.77
EDBTC-1	$384 \{N_{min} = N_{max} = N_b = 128\}$	99.63 / 100
EDBTC-2	$256 \{N_c = N_b = 128\}$	99.63 / 100

Table 4.9: Classification performance among the proposed method and former schemes under KTH-TIPS 2A image database.

Methods	Feature Dimension	KTH-TIPS 2A
SIFT [31]	-	52.70
LBP [38]	-	49.90
WLD [5]	-	56.40
DRLBP [45]	60	59.00
DRLTP [45]	176	62.60
EDBTC-1	$48 \{N_{min} = N_{max} = N_b = 16\}$	69.49 / 65.93
EDBTC-2	$32 \{N_c = N_b = 16\}$	65.59 / 67.15
EDBTC-1	$96 \{N_{min} = N_{max} = N_b = 32\}$	71.14 / 74.35
EDBTC-2	$64 \{N_c = N_b = 32\}$	69.78 / 72.90
EDBTC-1	$192 \{N_{min} = N_{max} = N_b = 64\}$	74.23 / 75.31
EDBTC-2	$128 \{N_c = N_b = 64\}$	70.08 / 74.58
EDBTC-1	$384 \{N_{min} = N_{max} = N_b = 128\}$	71.92 / 71.55
EDBTC-2	$256 \{N_c = N_b = 128\}$	70.31 / 71.94

that the proposed feature descriptor is suitable for performing an image retrieval and classification task.

4.5 Conclusions

A quantitative comparison of EDBTC image feature for color image retrieval and classification has been conducted and reported in this chapter. In the image retrieval and classification task, an image feature descriptor is simply derived and constructed from the EDBTC encoded data stream, i.e. two color quantizers and the bitmap image. The two EDBTC color quantizers produce the CHF which is effective for representing the color distribution of an image, whereas the bitmap image results the BHF for characterizing the image textural information as well as an image edges, lines, shapes, etc. The experimental results show that the proposed method offers a promising result in the image retrieval and classification task, and at the same time, the proposed method outperforms the former existing methods. The EDBTC image retrieval and classification system can be extended for the video processing.

References

- [1] A. R. Backes, D. Casanova, and O. M. Bruno. Color texture analysis based on fractal descriptors. *Pattern Recognition*, 45(5):1984–1992, 2012.
- [2] F. Bianconi, A. Fernandez, E. Gonzalez, D. Caride, and A. Calvino. Rotation-invariant colour texture classification through multilayer CCR. *Pattern Recognition Letters*, 30(8):765–773, 2009.
- [3] G. J. Burghouts and J. M. Geusebroek. Material-specific adaptation of color invariant features. *Pattern Recognition Letters*, 30(3):306–313, 2009.
- [4] B. J. L. Campana and E. J. Keogh. A compression-based distance measure for texture. *Statistical Analy Data Mining*, 3(6):381–398, 2010.
- [5] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikäinen, X. Chen, and W. Gao. WLD: a robust local image descriptor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1705–1720, 2010.
- [6] T. W. Chiang and T. W. Tsai. Content-based image retrieval via the multiresolution wavelet features of interest. *Journal of Information Technology and Applications*, 1(3):205–214, 2006.
- [7] J. J. de Mesquita Sa Junior, P. C. Cortez, and A. R. Backes. Color texture classification using shortest paths in graphs. *IEEE Transactions on Image Processing*, 23(9):3751–3761, 2014.
- [8] E. J. Delp and O. R. Mitchell. Image coding using block truncation coding. *IEEE Transactions on Communications*, 27(9):1335–1342, 1979.
- [9] M. R. Gahroudi and M. R. Sarshar. Image retrieval based on texture and color method in BTC-VQ compressed domain. In *9th International Symposium on Signal Processing and Its Applications (ISSPA)*, pages 1–4, 2007.
- [10] T. Guha and R. K. Ward. Image similarity using sparse representation and compression distance. *IEEE Transactions on Multimedia*, 16(4):980–987, 2014.

- [11] J. M. Guo. Improved block truncation coding using modified error diffusion. *Electronics Letters*, 44(7):462–464, 2008.
- [12] J. M. Guo and Y. F. Liu. Joint compression/watermarking scheme using majority-parity guidance and halftoning-based block truncation coding. *IEEE Transactions on Image Processing*, 19(8):2056–2069, 2010.
- [13] J. M. Guo and Y. F. Liu. Halftone-image security improving using overall minimal-error searching. *IEEE Transactions on Image Processing*, 20(10):2800–2812, 2011.
- [14] J. M. Guo and Y. F. Liu. High capacity data hiding for error-diffused block truncation coding. *IEEE Transactions on Image Processing*, 21(12):4808–4818, 2012.
- [15] J. M. Guo, S. C. Pei, and H. Lee. Watermarking in halftone images with parity-matched error diffusion. *Signal Processing*, 91(1):126–135, 2011.
- [16] J. M. Guo, H. Prasetyo, and J. H. Chen. Content-based image retrieval using error diffusion block truncation coding features. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(3):466–481, 2015.
- [17] Z. Guo, L. Zhang, and D. Zhang. Rotation invariant texture classification using LBP variance with global matching. *Pattern Recognition*, 43(3):706–716, 2010.
- [18] M. A. Hoang and J. M. Geusebroek. Measurement of color texture. *International Workshop on Texture Analysis and Synthesis (IWTAS)*, pages 73–76, 2002.
- [19] M. A. Hoang, J.M. Geusebroek, and A. W. M. Smeulders. Color texture measurement and segmentation. *Signal Processing*, 85(2):265–275, 2005.
- [20] P. W. Huang and S. K. Dai. Image retrieval by texture similarity. *Pattern Recognition*, 36(3):665–679, 2003.
- [21] N. Jhanwar, S. Chaudhurib, G. Seetharamanc, and B. Zavidovique. Content based image retrieval using motif co-occurrence matrix. *Image and Vision Computing*, 22(14):1211–1220, 2004.
- [22] M. Kokare, P. K. Biswas, and B. N. Chatterji. Texture image retrieval using new rotated complex wavelet filters. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 35(6):1168–1178, 2005.
- [23] KTH-TIPS. KTH-TIPS texture image database. <http://www.nada.kth.se/cvap/databases/kth-tips/index.html>.
- [24] R. Kwitt and A. Uhl. Image similarity measurement by Kullback-Leibler divergences between complex wavelet subband statistics for texture retrieval. In *15th IEEE International Conference on Image Processing (ICIP)*, pages 933–936, 2008.
- [25] R. Kwitt and A. Uhl. Lightweight probabilistic texture retrieval. *IEEE Transactions on Image Processing*, 19(1):241–253, 2010.
- [26] N. E. Lasmar and Y. Berthoumieu. Gaussian copula multivariate modeling for texture image retrieval using wavelet transforms. *IEEE Transactions on Image Processing*, 23(5):2246–2261, 2014.
- [27] 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*, 27(6):658–665, 2009.
- [28] G. H. Liu, Z. Li, Z. Lei, and Y. Xu. Image retrieval based on micro-structure descriptor. *Pattern Recognition Letters*, 44(9):2123–2133, 2011.
- [29] Y. F. Liu, J. M. Guo, and J. D. Lee. Halftone image classification using LMS

- algorithm and naïve Bayes. *IEEE Transactions on Image Processing*, 20(10):2837–2847, 2011.
- [30] Y. F. Liu, J. M. Guo, and J. D. Lee. Inverse halftoning based on the Bayesian theorem. *IEEE Transactions on Image Processing*, 20(4):1077–1084, 2011.
- [31] D. Lowe. Distinctive image feature from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [32] T. C. Lu and C. C. Chang. Color image retrieval technique based on color features and image bitmap. *Information Processing & Management*, 43(2):461–472, 2007.
- [33] Z. M. Lu and H. Burkhardt. Colour image retrieval based on DCT-domain vector quantization index histograms. *Electronics Letters*, 41(17):956–957, 2005.
- [34] B. S. Manjunath and W. Y. Ma. Texture feature for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.
- [35] MIT-VT. MIT-Vision Texture (VisTex) image database. <http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html>.
- [36] S. Murala, R. P. Maheshwari, and R. Balasubramanian. Local tetra patterns: a new feature descriptor for content-based image retrieval. *IEEE Transactions on Image Processing*, 21(5):2874–2886, 2012.
- [37] T. Ojala, M. Pietikainen, and Harwood D. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1):51–59, 1996.
- [38] T. Ojala, K. Valkealahti, E. Oja, and M. Pietikainen. Texture discrimination with multidimensional distributions of signed gray-level differences. *Pattern Recognition*, 34(3):727–739, 2001.
- [39] Outex. Outex texture image database. http://www.outex.oulu.fi/index.php?page=outex_home.
- [40] G. Paschos and M. Petrou. Histogram ratio features for color texture classification. *Pattern Recognition*, 24(1-3):309–314, 2003.
- [41] S. C. Pei and J. M. Guo. Hybrid pixel-based data hiding and block-based watermarking for error-diffused halftone images. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(8):867–884, 2003.
- [42] A. Porebski, N. Vandenbroucke, and L. Macaire. Haralick feature extraction from LBP images for color texture classification. In *1st Workshops on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–8, 2008.
- [43] P. Poursistani, H. Nezamabadi-pour, R. A. Moghadam, , and M. Saeed. Image indexing and retrieval in JPEG compressed domain based on vector quantization. *Mathematical and Computer Modelling*, 57(5-6):1005–1017, 2013.
- [44] G. Qiu. Color image indexing using BTC. *IEEE Transactions on Image Processing*, 12(1):93–101, 2003.
- [45] A. Satpathy, X. Jiang, and H. L. Eng. LBP-based edge-texture features for object recognition. *IEEE Transactions on Image Processing*, 23(5):1953–1964, 2014.
- [46] S. Silakari, M. Motwani, and M. Maheshwari. Color image clustering using block truncation algorithm. *International Journal of Computer Science Issues*, 4(2):31–35, 2009.
- [47] SIPI-USC. Brodatz texture image database. <http://sipi.usc.edu/database/database.php?volume=textures>.

- [48] Stex. Salzburg texture (Stex) image database. <http://www.wavelab.at/sources/STex/>.
- [49] M. Subrahmanyam, R. P. Maheswari, and R. Balasubramanian. Local maximum edge binary patterns: a new descriptor for image retrieval and object tracking. *Signal Processing*, 92(6):1467–1479, 2012.
- [50] M. Subrahmanyam, Q. M. J. Wu, R. P. Maheshwari, and R. Balasubramanian. Modified color motif co-occurrence matrix for image indexing. *Computers & Electrical Engineering*, 39(3):762–774, 2013.
- [51] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*, 19(6):1635–1650, 2010.
- [52] J.Z. Wang, J. Li, and G. Wiederhold. SIMPLcity: semantics-sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.
- [53] Y. G. Wu and S. C. Tai. An efficient BTC image compression technique. *IEEE Transactions on Consumer Electronics*, 44(2):317–325, 1998.
- [54] F. X. Yu, H. Luo, and Z.M. Lu. Colour image retrieval using pattern co-occurrence matrices based on BTC and VQ. *Electronics Letters*, 47(2):100–101, 2011.
- [55] 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 Transactions on Image Processing*, 19(2):533–544, 2010.