Block truncation coding using pattern fitting

Bibhas Chandra Dhara[a], Bhabatosh Chanda[b,*]

[a] Department of Information Technology, Jadavpur University, Salt Lake Campus, Kolkata 700098, India
[b] Electronics and Communication Sciences Unit, Indian Statistical Institute, 203 B.T. Road, Kolkata 700108, India

Received 11 June 2003; received in revised form 23 February 2004; accepted 23 February 2004

Abstract

Block truncation coding (BTC) divides an image into blocks having given size and then encodes each block by two representative gray levels and a bit-pattern. In this work a modified scheme of BTC is proposed where the computed representative gray levels are the bias and the contrast in each block. Secondly, instead of determining bit-pattern for each block, an optimum bit-pattern is selected from a pattern-book. Thus the index of the optimum pattern is used to encode in lieu of the explicit pattern. Thirdly, if the contrast is low the block is assumed to be smooth and bit-pattern is not required to reconstruct the block. This leads to significant reduction in bit-rate (bpp). Finally, the contrast component and the predictive residual of the bias component are entropy coded to achieve further reduction in bpp. Performance of the proposed scheme is measured in terms of peak-signal-to-noise ratio and bpp, and is compared with other recently reported methods.

Keywords: Image coding; Block truncation coding; Vector quantization; Pattern fitting

1. Introduction

Data compression is the mapping of a data set into a bit stream to decrease the number of bits required to represent data set. Image compression is one type of data compression. Image compression and image sequence coding has huge applications in video conferencing, video phones, TV transmission. Since 1970s the block truncation coding (BTC) [1–3] has been studied a lot. It is a lossy but attractive image coding scheme for its simplicity, low computational cost and relatively high quality. Like other image compression methods, quality of the reconstructed image is measured in terms of peak-signal-to-noise ratio (PSNR) and the degree of compression by bits-per-pixel (bpp). In this method sharp gray-level transitions and textured areas are reconstructed well; whereas smooth gray-level transitions are less well preserved. The BTC output data set includes a bit-pattern, which defines the quantization bin of each pixel, and two reconstruction levels determined to preserve the original mean and variance of the pixel gray level over the block. To form the bit-pattern in this method, block mean is set as threshold. Instead of geometric moments of gray levels (i.e., mean and variance), absolute moments may also be used [2]. The main drawback of original BTC is the high bit-rate (2 bpp). Many modification have been proposed to reduce bit-rate, such as median filtering [4], adaptive coding [5], DCT-BTC [6], BTC with decimation and interpolation(D/I) [7], etc. In order to reduce the bit-rate vector quantization (VQ) [8,9] technique has also been applied to quantize the vector formed by the two values corresponding to the reconstruction levels generated by the BTC for each block [10]. To form bit-pattern BTC uses block-mean as threshold and is used in coding. Generated bit-pattern may be represented by one of a set of pre-selected bit-patterns [11] so that the index of the representing bit-pattern can be used to code the block. Thus bit-rate is further reduced. However, since the approximating bit-pattern is selected based on the minimum Hamming distance between the generated bit-pattern and the bit-patterns in the set, original gray levels in the block may not fall in appropriate quantization bin.

In the present study we try to fit two-level patterns or templates, taken from a collection called pattern-book, to

* Corresponding author. Tel.: +91-33-2575-2915; fax: +91-33-2577-6680.

E-mail addresses: bibhas@it.jusl.ac.in (B.C. Dhara), chanda@isical.ac.in (B. Chanda).

the gray levels present in a block in least-square error sense. Pattern fitting is an well-known concept in image processing and analysis, and were used to detect edges [12], local features [13], and object of known shape [14]. The pattern that best fits the block represents its bit-pattern and is used in BTC. Thus, the index of the selected pattern is sufficient to reconstruct the block. These patterns are designed based on spatial homogeneity as well as the expected nature of local features. Based on the best fit pattern two quantities, namely bias and contrast, are computed and that are used in the reconstruction of the block. Both these values are entropy coded to further reduce the bpp. The paper is organized as follows. Section 2 contains basic concept of BTC and the proposed method. How an image block is represented by a two-level pattern is discussed in Section 3. Coding of the reconstruction levels is studied in Section 4. A summary of test results appears in Section 5. Finally, conclusion are drawn in Section 6.

2. Basic concept

The BTC algorithm is a lossy fixed length compression method that uses a Q level quantizer to quantize a local region of the image. The quantizer levels are chosen such that a number of moments of gray levels over a local region in the image are preserved in the quantized output. For a two-level (1 bit) quantizer, one of the two intensity values is selected for a pixel in the block. These values are chosen such that the sample mean and variance of the reconstructed block are identical to those of the original block. After coding a bit-pattern of image block and two values a and b corresponding to two quantization levels are obtained. It is easy to understand that smaller data rate can be achieved by selecting bigger block size or by allocating less number of bits for reconstruction levels a and b. Suppose an image is divided into a number of $n \times n$ blocks. Let k be the number of pixels in each block ($k = n^2$). Let $f(\mathbf{x}_i)$, $\mathbf{x}_i \in C$ are the intensity values of the pixels in a block of the original image where C represents the set of coordinates of pixels in the block, i.e., $C = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$. The first two sample moments m_1 and m_2 are given by

$$m_1 = \frac{1}{k}\sum_{i=1}^{k} f(\mathbf{x}_i),$$

$$m_2 = \frac{1}{k}\sum_{i=1}^{k} f^2(\mathbf{x}_i),$$

where m_1 is the sample mean and the sample variance σ^2 of image block is given by

$$\sigma^2 = m_2 - m_1^2.$$

Suppose based on the pixel intensities the quantizer partitions the C, in other words, the block into two sets of pixels C_0 and C_1, such that $C = C_0 \cup C_1$ and $C_0 \cap C_1 = \emptyset$, where $C_0 = \{\mathbf{x}'_1, \mathbf{x}'_2, \ldots, \mathbf{x}'_{k'}\}$ and $C_1 = \{\mathbf{x}''_1, \mathbf{x}''_2, \ldots, \mathbf{x}''_{k-k'}\}$. Then $\{f(\mathbf{x}_i) | \mathbf{x}_i \in C_0\}$ defines the candidates of a quantization bin and $\{f(\mathbf{x}_i) | \mathbf{x}_i \in C_1\}$ that of the other bin. Suppose this partition is represented by assigning one of two labels, say 0 and 1, to the pixels. Without loosing generality, let the pixels of set the C_0 are marked by 0 and that of C_1 by 1. Thus the partition can be represented as a bit-pattern as stated above. During reconstruction, the pixels marked by 0 will be given the value $A - d$ and that marked by 1 will be given the value $A + d$. The values A and d represent the bias (low frequency component) and contrast (high frequency component), respectively, within the block. These values are to satisfy

$$km_1 = k'(A - d) + (k - k')(A + d), \tag{1}$$

$$km_2 = k'(A - d)^2 + (k - k')(A + d)^2. \tag{2}$$

Solving for A and d we get

$$A = m_1 + \frac{\sigma(2k' - k)}{2\sqrt{k'(k - k')}}, \tag{3}$$

$$d = \frac{\sigma k}{2\sqrt{k'(k - k')}}. \tag{4}$$

Like original BTC method in the proposed method too two gray levels $A + d$ and $A - d$ are used in image reconstruction at decoding phase. Hence, intensity $\hat{f}(\mathbf{x}_i)$ of the pixels of the corresponding block of the reconstructed image is given by

$$\hat{f}(\mathbf{x}_i) = \begin{cases} A + d & \text{if } \mathbf{x}_i \in C_1, \\ A - d & \text{if } \mathbf{x}_i \in C_0. \end{cases} \tag{5}$$

It is evident from Eqs. (3) and (4) that $a = A - d$ and $b = A + d$, where quantization levels a and b are used in conventional BTC. Advantage of using (A, d) instead of (a, b) is explained later. Only relevant question we are left with is how to define the partition, which is given in the following section.

3. Partitioning

In conventional BTC method to generate the bit-pattern representing the partition of an image block, block mean is used as threshold and each pixel is labeled independently depending only on its intensity value. Thus the bit-pattern is generated as

$$B(\mathbf{x}_i) = \begin{cases} 0 & \text{if } f(\mathbf{x}_i) \leqslant m_1, \\ 1 & \text{if } f(\mathbf{x}_i) > m_1. \end{cases} \tag{6}$$

So two partitions are defined $C_0 = \{\mathbf{x}_i | B(\mathbf{x}_i) = 0\}$ and $C_1 = \{\mathbf{x}_i | B(\mathbf{x}_i) = 1\}$. Fig. 1(a) shows original “LENA” image of size 512×512. Reconstructed image after conventional BTC is shown in Fig. 1(b) for which PSNR = 32.89 and bpp=2.

Fig. 1. (a) Original image and (b) Reconstructed image with conventional BTC (PSNR= 32.89 and bpp=2).

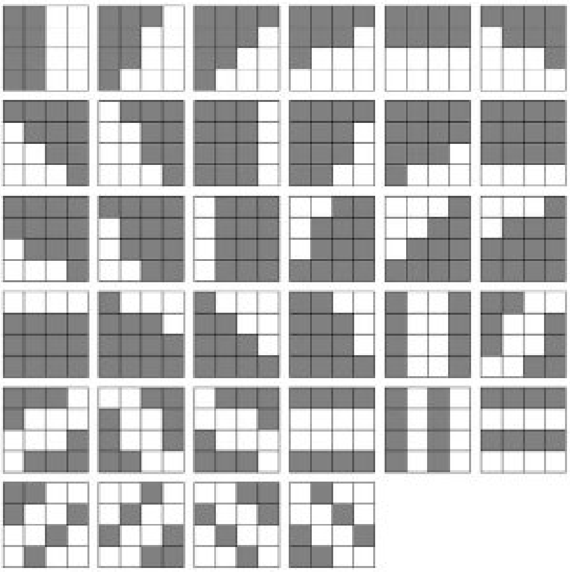

Fig. 2. Set of two-level patterns (also viewed as bit-patterns) used to fit with gray levels of image block in the proposed method.

Generally for any image there is spatial homogeneity in pixel intensity and this is not taken care of in the conventional BTC method. Some works [11,15] have implicitly exploited this characteristic through sub-sampling and interpolation of pixel intensity. Here, on the contrary, we use this characteristic explicitly in our partitioning method. To define partition we consider a pattern-book containing N two-level patterns (one level is represented by 1 and the other by 0). These patterns are designed based on the spatial homogeneity as well as finer details like edge, line etc. in different orientation. Then we try to fit the candidate image block to each of these patterns $P_j, j = 1, 2, \ldots, N$, say, in mean-square-error sense. For example, if we try to fit the image block to the pattern P_j, mean-square-error in fitting is computed as follows. Suppose $P_j = P_{j0} \cup P_{j1}$, where all pixels of the set P_{j0} have label 0 and that of P_{j1} have label 1. Then

$$e_{j0} = \frac{1}{k'} \sum_{\mathbf{x}_i \in P_{j0}} (f(\mathbf{x}_i) - \mu_0)^2, \tag{7}$$

$$e_{j1} = \frac{1}{k - k'} \sum_{\mathbf{x}_i \in P_{j1}} (f(\mathbf{x}_i) - \mu_1)^2, \tag{8}$$

where

$$\mu_0 = (1/k') \sum_{\mathbf{x}_i \in P_{j0}} f(\mathbf{x}_i) \quad \text{and}$$

$$\mu_1 = (1/(k - k')) \sum_{\mathbf{x}_i \in P_{j1}} f(\mathbf{x}_i).$$

Hence, total error in fit for the jth pattern is given by $e_j = e_{j0} + e_{j1}$. Finally, index m of best fit pattern is obtained for $m = \arg\min_j \{e_j\}$.

Hence, the proposed BTC sends the index m instead of the entire bit-pattern. That means only $\log_2 N$ is transmitted instead of k. If former one is much less than the latter (which is usually taken), a significant reduction in bpp can be achieved. Once the pattern is fixed, A and d are calculated using Eqs. (3) and (4). The encoder will transmit these two values and index of the selected pattern. In our experiment we have used first (top) 30 patterns of Fig. 2, their complements and last four patterns making $N = 64$. Result of BTC obtained from fitting patterns of Fig. 2 is shown in Fig. 3(a) where PSNR= 31.79 bpp= 1.375.

Fig. 3. Output images after pattern fitting: (a) A, d (PSNR= 31.79, bpp= 1.375) (b) A, d' (PSNR= 31.61, bpp= 0.88) (c) A', d' (PSNR= 31.57, bpp= 0.64).

4. Encoding the block

In conventional BTC and also in absolute moment BTC (AMBTC) [2] the quantization data to be stored in the compressed file is either the pair (a,b) or $(a,b-a)$. They are usually expressed by $8+8$ bits, but they can also be stored by 10 bits using joint quantization [16,17]. Joint bit rate may also be reduced by Vector Quantization [10]. The pair can even be coded by discrete cosine transform (DCT) [6]. In our work $A-d$ and $A+d$ are used as quantization levels and are used in reconstruction of block. Here we store A and d. This gives advantage over the usual quantization level a and b in coding purpose because of the following reasons.

It is evident from Eqs. (3) and (4) that $A=(b+a)/2$ and $d=(b-a)/2$. In view of the Wavelet theory [18], A may be considered as the response of the scaling function or the low-resolution ($\frac{1}{4}$ times) representation of the block and d is the response of wavelets whose natures are represented by the patterns shown in Fig. 2. There may be many different pairs of (a,b) for which d or $(b-a)/2$ is same. So the standard deviation of d is smaller than that of a and b, and this leads to higher compression for d by entropy coding. Secondly, the contrast or d-value close to 0 means the gray levels within the block are more or less uniform. In that case the gray levels within the block can reliably be represented by the bias value, i.e., A only. From this point of view we apply a small threshold, d_{th}, to obtain

$$d' = \begin{cases} 0 & \text{if } d \leqslant d_{th}, \\ d - d_{th} & \text{otherwise,} \end{cases} \tag{9}$$

d_{th} is selected in such a way that the error introduced due to this approximation does not reduce the PSNR of that particular block below a fixed quantity, say, Z dB (see Eq. (14)).

Thus, in general, we need to transmit d', A and index of the corresponding pattern in the said order. However, it may be noted that if d' is zero, we need not transmit the pattern index. This leads to further reduction in bit rate. The result of using $d_{th}=4$ (corresponding Z values is 35 (approx.)) is shown in Fig. 3(b) for which PSNR $=$ 31.61 and bpp $=$ 0.88 considering entropy coding for d' and 8-bits for A.

The values of A do not usually cover the whole dynamic range spanned by 8 bits. The range of values also vary from image to image. So we transform the values of A to cover the range from 0 to 2^l, where l is the largest integer not exceeding $\log_2(A_{max}-A_{min})$. Hence, the transformed value is defined as

$$A' = \frac{2^l(A-A_{min})}{(A_{max}-A_{min})}. \tag{10}$$

Used for Prediction 2	Used for Prediction 3	Used for Prediction 4
Used for Prediction 1	Current Block C	

Fig. 4. Blocks used in Prediction of A in current block.

		ng_1	ng_2		
	*	p_1	p_2	*	
ng_8	p_8	$\tilde{A}$	$\tilde{A}$	p_3	ng_3
ng_7	p_7	$\tilde{A}$	$\tilde{A}$	p_4	ng_4
	*	p_6	p_5	*	
		ng_6	ng_5		

Fig. 5. Pixel intensity at the time of decoding of Block 4×4, if $\tilde{d}=0$.

It may be noted that this needs transmission of A_{min} and A_{max} once at the beginning. Corresponding output is shown in Fig. 3(c) for which PSNR=31.57. As mentioned earlier that A is bias or average intensity of the block, it has stronger correlation with that of its neighboring blocks compared to a or b as used in conventional BTC because of spatial homogeneity. So for further reduction in bpp, A' is coded by predictive coding. Fig. 4 shows the blocks that are used for prediction of A'. Suppose the A' values of the blocks numbered 1, 2, 3, 4, C are A'_1, A'_2, A'_3, A'_4 and A'_c, and are known. Assuming a linear model, the prediction error is

$$\Delta A' = A'_c - \sum_{i=1}^{4} w_i A'_i, \tag{11}$$

where w_1, w_2, w_3, w_4 are weights and are estimated considering a large number of images such that minimum $\Delta A'$ is achieved on an average. It is evident that the variance of $\Delta A'$ is much less than that of A'. Hence, through entropy coding $\Delta A'$ can be represented by less number of bits than that is needed to represent A' directly. This prediction coding does not introduce any further error in reconstructed image. So output is same as Fig. 3(c) for which PSNR=31.57 and the bpp is now reduced to 0.64 by predictive entropy coding for both d' and $\Delta A'$. During reconstruction the approximate value $\tilde{A}$ (resp. $\tilde{d}$) of A' (resp. d') used, where

$$\tilde{A} = \frac{A'(A_{max}-A_{min})}{2^l} + A_{min}, \tag{12}$$

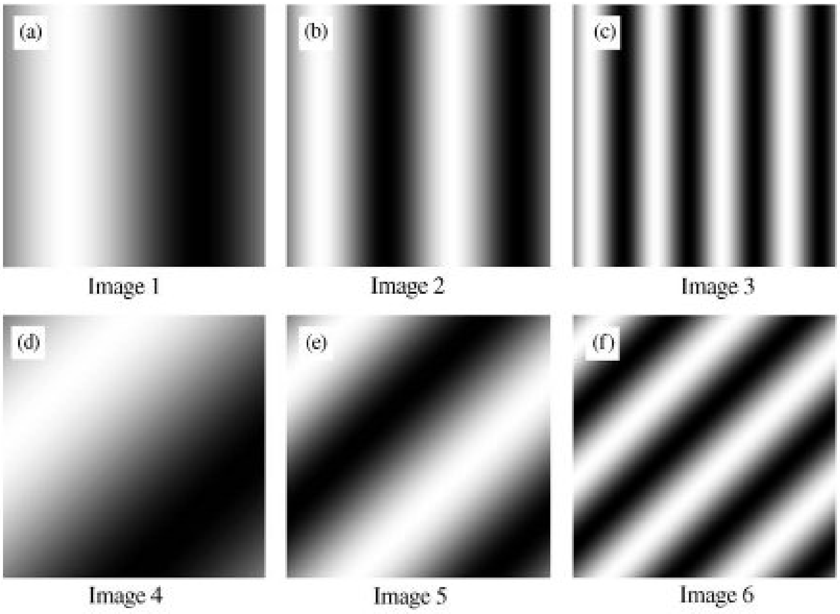

Fig. 6. Synthetic images used to compare INTM and CONM.

Table 1
Results of INTM and CONM method

Image	PSNR	
	INTM	CONM
Image1	47.29	44.17
Image2	45.53	39.64
Image3	40.47	38.57
Image4	48.26	45.93
Image5	46.95	42.08
Image6	41.60	38.96

$$\tilde{d} = \begin{cases} 0 & \text{if } d' = 0, \\ d' + d_{th} & \text{if } d' > 0. \end{cases} \tag{13}$$

At decoding phase when a block is smooth ($\tilde{d} = 0$) the intensity of all pixels are set to $\tilde{A}$. Let us call this method CONM. However, instead of setting all pixels to $\tilde{A}$, we propose to set only central 2×2 block (see Fig. 5) to $\tilde{A}$. In the second pass intensity of pixel p_i, for $i = 1, 2, \ldots, 8$ is determined by linear interpolation using horizontal and vertical set of data depending on context. In the next pass the intensity of pixels marked with '$*$' are computed by bi-linear interpolation. Let us call this method INTM. It is expected that INTM should give higher PSNR as well as smoother visual appearance than the method CONM. We have experimentally verified this by applying both the methods on six synthetic images as shown in Fig. 6, where intensities are generated as two-dimensional sine wave of various frequency and orientations. Table 1 shows the comparative performance of these two methods.

5. Experimental results

We have studied the nature of d', A' and $\Delta A'$ on several images namely, Airplane, Baboon, Barbara, Boat, Couple, Lake, Lena, Man, Peppers and Zelda as shown in Fig. 7. The used images are monochrome still images of size 512×512 and intensity resolution is 8 bit. According to the average distribution obtained from the study, the weights w_1, w_2, w_3, w_4 are determined and two Huffman codebooks for d' and $\Delta A'$ are developed. This is done to make the system applicable for a wide variety of images.

The block size is 4×4 and d_{th} value is set to 4. Here number (N) of two-level patterns used is 64 as shown in Fig. 2 (first 30 patterns and their compliments plus the last 4 patterns). The performance is evaluated by bpp and PSNR. PSNR is defined as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}} \, \text{d}B, \tag{14}$$

where MSE (mean square error) for a reconstructed image $\hat{f}(\mathbf{x}_i)$ is defined as

$$\text{MSE} = \frac{1}{M} \sum_i [f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i)]^2,$$

where M is the total number of pixels in the image.

Fig. 7. The original images used in the experiment.

The results of the proposed method, and two other recently proposed modified BTC techniques namely VQ-CIBTC [19] and Adapt.D/I [15] are compared in Table 2. The reconstructed images using proposed method are shown in Fig. 8. In VQ-CIBTC method we have set Thr1 to 10 and Thr2 to 50 as suggested in the original paper [19] and used 2 bits for identifying the class of a block. In Adapt.D/I method Thr1 and Thr2 have been set to 6 and 5, respectively, to achieve the PSNR close to what was reported in the original paper [15] and bit-rate has been measured without as well as with predictive entropy coding.

6. Conclusion

BTC is known for maintaining high PSNR, but compression ratio is low. In this paper a modified scheme of BTC is presented to achieve higher compression ratio, i.e.,

Table 2
Experimental result of proposed method and others

Image	Proposed method		VQ-CIBTC [19]		Adapt.D/I [15]		
	PSNR	bpp	PSNR	bpp	PSNR	bpp	
						Without predictive coding	With predictive coding
Airplane	30.52	0.60	30.15	0.76	31.59	1.24	1.17
Baboon	23.79	1.10	23.70	0.95	26.87	2.01	1.81
Barb	26.18	0.77	26.12	0.84	29.53	1.54	1.39
Boat	30.03	0.70	29.48	0.80	29.93	1.36	1.30
Couple	31.65	0.75	30.61	0.82	32.46	1.52	1.30
Lake	26.16	0.94	25.70	0.89	27.87	1.94	1.69
Lena	31.59	0.64	30.92	0.78	31.87	1.44	1.20
Man	29.96	0.80	29.33	0.84	31.23	1.59	1.36
Peppers	31.69	0.68	30.76	0.79	32.69	1.60	1.30
Zelda	34.89	0.60	33.57	0.76	34.14	1.39	1.10
Overall average	29.65	0.76	29.03	0.82	30.82	1.56	1.36

lower bpp keeping PSNR as high as possible. Here instead of determining bit-pattern for a block, an optimum bit-pattern is selected from a pattern-book. So in lieu of the explicit bit-pattern only the index of the optimum pattern is used to encode the block. The bit-patterns stored in the pattern-book are designed considering spatial homogeneity as well as fine structures like edge, line, etc. However, these patterns are designed heuristically based on extensive observation and experimentation guided by intuition. So a well-devised technique for determining the pattern-book need to be developed to avoid human bias and error. We have computed representative gray levels in such a way that they represent the bias and the contrast in each block. The advantage is that the low contrast component indicates that the block is smooth and, in that case, only bias value (and no bit-pattern) is needed to reconstruct the block. However, instead of assigning the bias value to all the pixels of the reconstructed block, we assign this value only to the central part of the block and the remaining pixel values are determined by linear interpolation using the gray levels of the neighboring blocks to achieve high PSNR and smoother visual appearance. This of course, incurs higher computational cost during reconstruction. Another advantage is that the bias component shows much stronger spatial correlation than the conventional representating gray levels. So predictive residual of the bias component shows a very strong unimodal p.d.f. which suggest very efficient coding by, say, Huffman coding scheme. In this work we have adopted a simple linear prediction model. However, a better prediction model may be searched out. The performance of the proposed scheme is measured in terms of PSNR and bpp. In case of most of the images, it shows reasonably high PSNR and low bpp. Comparison with other methods shows its acceptability/superiority.

7. Summary

Block truncation coding is a well-known lossy image compression technique and is being used for last two decades. BTC divides an image into a number of blocks of fixed size, and encodes each block independently by two representative values and a bit-pattern of size equal to that of the block. The technique is known to preserve high PSNR, but achieves low compression ratio. In this work a modified scheme of BTC is proposed to achieve high compression ratio keeping PSNR as high as possible. Here instead of determining bit-pattern for a block an optimum bit-pattern is selected from a pattern-book. Pattern-book is designed considering the spatial homogeneity as well as fine structure like edges, line, etc. For encoding purpose the index of the optimum pattern is used. Here representative levels are computed in such a way that they represent the bias and contrast of the block. One advantage of these representative levels is that the bias components of neighboring blocks are strongly correlated. So predictive coding can successfully be used. Contrast component and the predictive error of bias component are entropy coded to achieve more compression. Finally, low contrast component indicate that block is smooth, in that case, only bias value is needed to reconstruct the block, no bit-pattern is needed. Instead of assigning bias value to all pixels of the reconstructed block, this value is assigned to central part of the block and remaining pixel values are determined by interpolation

Fig. 8. The reconstructed images using proposed method.

using gray levels of neighboring blocks. Advantages are achieving higher PSNR value and maintaining spatial homogeneity in gray levels across the border of the blocks.

References

[1] E.J. Delp, O.R. Mitchell, Image compression using block truncation coding, IEEE Trans. Commun. 27 (1997) 1335–1342.

[2] M. Lema, O. Mitchell, Absolute moment block truncation coding and applications to color images, IEEE Trans. Commun. 32 (1984) 1148–1157.

[3] P. Frani, O. Nevalainen, Block truncation coding with entropy coding, IEEE Trans. Commun. 43 (1995) 1677–1685.

[4] G. Arce, N.J. Gallagher, Btc image coding using median filter roots, IEEE Trans. Commun. 31 (1983) 784–793.

[5] R.P. Naciopoulos, D. Morse, Adaptive compression coding, IEEE Trans. Commun. 39 (1991) 1245–1254.

[6] Y. Wu, D.C. Coll, Btc-vq-dct hybrid coding of digital images, IEEE Trans. Commun. 39 (1991) 1283–1287.

[7] B. Zeng, Y. Neuvo, Interpolative btc image coding with vector quantization, IEEE Trans. Commun. 41 (1993) 1436–1438.
[8] A.Y. Linde, R.M. Gray, An algorithm for vector quantization, IEEE Trans. Commun. 28 (1980) 84–95.
[9] M.N. Nasrabadi, R.B. King, Image coding using vector quantization: a review, IEEE Trans. Commun. 36 (1988) 957–971.
[10] V. Udpikar, J.P. Raina, Btc image coding using vector quantization, IEEE Trans. Commun. 35 (1987) 352–355.
[11] S.I. Olsen, Block truncation and planar image coding, Pattern Recognition Lett. 21 (2000) 1141–1148.
[12] M. Hueckel, An operator which locates edges in digitized pictures, J. Assoc. Comput. 18 (1971) 113–125.
[13] R. Haralick, L. Shapiro, Computer and Robot Vision, Addison-Wesley, Reading, 1992 (Chapter 8).
[14] A. Rosenfeld, A. Kak, Digital Picture Processing, 2nd Edition, Academic Press, New York, 1982 (Chapter 9).
[15] Y. Wang, G. Tu, Block truncation coding with adaptive decimation and interpolation, Proceedings of SPIE, International Conference on Visual Communication and Image Processing 2000 (VCIP 2000), Vol. 4067, Australia, 2000, pp. 430–437.
[16] D.J. Healy, O.R. Mitchell, Digital video bandwidth compression using btc, IEEE Trans. Commun. 29 (1981) 1809–1817.
[17] O. Mitchell, E.J. Delp, Multilevel graphics representation using block truncation coding, Proc. IEEE 68 (1980) 868–873.
[18] C. Valens, A really friendly guide to wavelets, available at: http://perso.wanadoo.fr/, 1999.
[19] C.H. Kuo, C.F. Chen, W. Hsia, A compression algorithm based on classified interpolative block truncation coding and vector quantization, J. Inf. Sci. Eng. 15 (1999) 1–9.

About the Author—BIBHAS CHANDRA DHARA was born in 1976. He received B.Sc. (Hons) in Mathematics from University of Calcutta in 1997. He received B.Tech in Computer Science & Engineering from University of Calcutta in 2000. He received M.Tech in Computer Science from Indian Statistical Institute in 2002. Currently, he is working as a Lecturer in Department of Information Technology, Jadavpur University, Kolkata, India. His research area and interest include Image Processing and Video Compression.

About the Author—BHABATOSH CHANDA was born in 1957. He received B.E. in Electronics and Telecommunication Engineering and Ph.D. in Electrical Engineering from University of Calcutta in 1979 and 1988, respectively. He received 'Young Scientist Medal' of Indian National Science Academy in 1989 and 'Computer Engineering Division Medal' of the Institution of Engineers (India) in 1998. He is also recipient of UN fellowship and , UNESCO-INRIA fellowship. He is a Fellow of Institute of Electronics and Telecommunication Engineers and the National Academy of Science, India. He also received Shri Hari Om Prerit Vikram Sarabhai Research Awards in 2001. He has published more than 70 technical articles. His research interest includes Image Processing, Pattern Recognition, Computer Vision and Mathematical Morphology, Video Compression, and the application domain includes document page image, satellite image, biomedical image and indoor scenes. He has been working as a Professor at Indian Statistical Institute, Kolkata, India since 1996.