ISSN: 0976-9102(ONLINE)
DOI: 10.21917/ijivp.2012.0060

MULTISTAGE BITRATE REDUCTION IN ABSOLUTE MOMENT BLOCK TRUNCATION CODING FOR IMAGE COMPRESSION

S. Vimala
Department of Computer Science, Mother Teresa Women's University, India
Email: vimalaharini@gmail.com

Abstract
Absolute Moment Block Truncation Coding (AMBTC) is one of the lossy image compression techniques. The computational complexity involved is less and the quality of the reconstructed images is appreciable. The normal AMBTC method requires 2 bits per pixel (bpp). In this paper, two novel ideas have been incorporated as part of AMBTC method to improve the coding efficiency. Generally, the quality degrades with the reduction in the bit-rate. But in the proposed method, the quality of the reconstructed image increases with the decrease in the bit-rate. The proposed method has been tested with standard images like Lena, Barbara, Bridge, Boats and Cameraman. The results obtained are better than that of the existing AMBTC method in terms of bit-rate and the quality of the reconstructed images.

Keywords:
Compression, Bit-Rate, bpp, Block, Bit-Plane, High-Mean, Low-Mean

1. INTRODUCTION

With the continuing growth of modern communication technology, demand for image transmission and storage is increasing rapidly. Advances in computer technology for mass data storage and digital processing have paved the way for implementing advanced data compression techniques to improve the efficiency of transmission and storage of images [1]. Image data compression is concerned with minimization of the number of information carrying units used to represent an image. Image compression techniques are classified into two main types namely Lossy and Lossless Compression techniques. In lossless techniques, the reconstructed images must be the exact replica of the input images. Lossless compression techniques are suitable for fields such as Medical Imaging, where a minute loss in data may lead to risks to human life. Lossy techniques lead to loss of data when reconstructing the compressed image. These techniques are suitable for fields such as Television displays where human vision system is involved. Loss in image data does not make much difference when the reconstructed images are displayed. Block Truncation Coding (BTC) is one such lossy technique. BTC has the advantages of low computational load and less memory requirement.

The key idea of BTC is to perform moment preserving quantization for blocks of pixels so that the quality of the image will remain acceptable. Though the compression rate gained by JPEG compression algorithm is high [2], BTC has gained popularity due to its practical usefulness. Several improvements of the basic method have been recently proposed in the literature.

In BTC, the input image is broken into small blocks of size 4x4 pixels. The mean $\bar{x}$ and the standard deviation σ are computed for each block using the Eq.(1) and Eq.(2) respectively.

$$\bar{x} = \frac{1}{K}\sum_{i=1}^{k} x_i \tag{1}$$

$$\sigma = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(\bar{x} - X_i)^2} \tag{2}$$

Each pixel value x_i of the block is compared against the mean value $\bar{x}$. If the pixel value is greater than or equal to mean, it is coded as 1 else it is coded as 0. The bitplane of size 16 bits of the block is generated using the Eq.(3) as follows,

$$if\ x_i \geq \bar{x} \quad \text{then 1 else 0.} \tag{3}$$

Two quantizing levels q_1 and q_2 are computed using the Eq.(4) and Eq.(5) respectively as,

$$q_1 = \bar{x} + \sigma\sqrt{\frac{p}{(k-p)}} \tag{4}$$

where, p is the number of pixels whose values are greater than or equal to $\bar{x}$,

$$q_2 = \bar{x} - \sigma\sqrt{\frac{(k-p)}{p}}. \tag{5}$$

For each block of pixels, the bit-plane, the two quantizing levels q_1 and q_2 are transmitted or stored. This leads to a bit rate of 2 bits per pixel (bpp). A straight forward coding of $\bar{x}$ and σ by m bits yields a bit rate of,

$$\frac{m+m+k}{k} = 1 + \frac{2m}{k} \text{ bits per pixel (bpp).} \tag{6}$$

The compressed image is reconstructed using the bit plane as follows, the bit 1 in the bit-plane is replaced with q_1 and the bit 0 is replaced with q_2. The BTC method is fast, requires very little extra memory, easy to implement and has low computational demand [3].

Goeddel and Bass [4] proposed a heuristic selection criterion for the threshold:

$$x_{th} = \frac{(x_{\min} + x_{\max})}{2} \tag{7}$$

where, $x_{\max}$ and $x_{\min}$ stand for the minimal and maximal pixel values of a block. This criterion gives improved MSE values. Another variant of BTC is absolute moment block truncation coding (AMBTC). In AMBTC by Lema and Mitchell [5], the lower mean and the higher mean are computed using the Eq.(8) and Eq.(9) respectively.

$$\overline{x_l} = \frac{1}{p}\sum_{x_i \geq \bar{x}} x_i \tag{8}$$

$$\overline{x_h} = \frac{1}{k-p}\sum_{x_i < \bar{x}} x_i. \tag{9}$$

This method minimizes the MSE value among the BTC variants that use $\bar{x}$ as a quantization threshold. The coding and decoding processes are very fast for AMBTC because square root and multiplication operations are omitted. Some improvements have been done to the existing AMBTC method to reduce the bit-rate without much affecting the quality of the reconstructed image. BTC achieves 2 bits per pixel (bpp) with low computational complexity [6]. Many techniques with modifications to BTC have been proposed to reduce the bit-rate obtained with normal BTC. Yung-Gi Wu [7] proposed Probability based Block Truncation Image Bit-plane Coding; Yu-Chen Hu [8] presented a Modified BTC with Predictive Technique and Bit-plane Coding with Edge Pattern. Various authors extended BTC to multi-spectrum images such as color images [9]–[11].The remaining part of the paper is organized as follows: In section 2, the proposed methods are explained. In section 3, the results are discussed and the conclusion is given in section 4.

2. PROPOSED METHOD

In the proposed method, compression is achieved in two stages.

2.1 STAGE 1

In the first stage, the input image blocks are classified into low detail blocks and high detail blocks. To do this classification, the mean of each block is computed using the Eq.(1). Then the sum of difference between the mean and the individual values of the block is computed using the Eq.(10).

$$S = \sum_{i=1}^{k}\left(\bar{x} - x_i\right) \tag{10}$$

This sum value is compared against a threshold value. If the sum exceeds the threshold value, the block is identified as a high detail block and if the sum value is less than the threshold value, the block is identified as a low detail block. For a low detail block, only the mean is stored. For a high detail block, the bit plane, the high mean and the low mean values are stored. A high detail block requires 32 bits to get stored and the low detail block requires only 8 bits to store the Mean value. The coding efficiency depends on the number of high and low detail blocks available. If the number of low detail blocks is high, the bit-rate is reduced to a significant level without much loss in the quality of the reconstructed image.

2.1.1 Algorithm:

Step 1: Input the image blocks one by one

Step 2: Compute the mean of the block using Eq.(1).

Step 3: Generate the bit-plane using the Eq.(3).

Step 4: Compute the sum using the Eq.(9).

Step 5: if Sum >= threshold value, compute $\overline{x_h}$ and $\overline{x_l}$ using the Eq.(7) and Eq.(8).
Else retain the mean value.

Step 6: end.

2.2 STAGE 2

In stage 2, a novel idea has been incorporated, where the values $\overline{x_h}$, $\overline{x_l}$ and $\bar{x}$ are divided by 4. Generally, these statistical moments need 8 bits to store them. Because the maximum gray scale is 256 which requires $Log_2(256)$ bits ($Log_2(256)=8$). But when divided by 4, the maximum possible value would be 64 and hence only 6 bits are needed to represent them, i.e. $Log_2(64)=6$ bits. This leads to reduction in the bit-rate by 0.25 bpp for the whole image.

The decoding stage is carried out in two steps: In step 1, the high mean and the low mean values of high detail blocks and the mean value stored for the low detailed blocks are multiplied by 4 to get back more or less the original values. In step 2, for a high detail block, each 1 in the bit-plane is replaced with the high mean and the 0 in the bit-plane is replaced with the low mean. For a low detail block, the whole block is coded with the mean of the block.

The algorithms are implemented in Matlab 7.0 on Windows XP Operating System. Hardware Used: Intel Core 2 E7400@ Duo 2.8 GHz Processor with 2 GB RAM. The experiments were carried out with the standard images Camera, Boats, Bridge, Baboon, Lena, Barbara and Kush of 256 x 256 pixels. The input images taken for the study are given in Fig.1.

3. RESULTS AND DISCUSSION

The results obtained in terms of bpp (bits per pixel) and PSNR (quality of the reconstructed image) with the conventional AMBTC are given in the first two columns of the Table.1. The bit-rate achieved is 2 bpp and the average PSNR obtained with the existing AMBTC is 35.60. The two novel ideas discussed in the proposed method are applied separately to the input images and the results are given under the columns *Stage 1* and *Stage 2*. For *cameraman* image, the bpp obtained is 1.17 with the PSNR of 32.09. When compared to the results of AMBTC, the bit rate is reduced by a value of 0.83 (41.5% reduction in bit-rate) and there is a very little degradation in the PSNR, i.e. by only a value of 0.08. In stage 1, the bpp varies for each image, as the bit-rate depends on the number of high and low detail blocks. In stage 2, a uniform bit-rate of 1.75 is achieved for all images. The results obtained by combining both these ideas are given under the column *Hybrid*. In this case, the output of stage 1 is given as input to the stage 2. Thus two level compressions is achieved.

For camera image, the bit-rate has been reduced to 0.99 bpp with the increased PSNR of 32.23. When we observe the results of the hybrid method, there is a significant improvement both in terms of bpp and PSNR. On an average, the bit-rate from 2 bpp of AMBTC method has been reduced to 1.31 bpp and the PSNR has been raised to 34.47. The proposed method outperforms the AMBTC method both in terms of bit-rate and the quality of the reconstructed images. The reconstructed images of the methods discussed in the paper are given in Fig.2 and Fig.3.

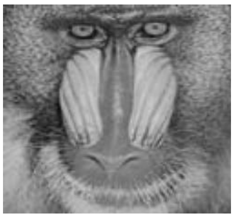
Table.1. Comparison of bpp and PSNR with respect to AMBTC and the proposed method in two stages

Image	1st Level		2nd Level				3rd Level	
			Stage 1		Stage 2		Hybrid	
	bpp	PSNR	bpp	PSNR	bpp	PSNR	bpp	PSNR
Camera	2.00	32.17	1.17	32.09	1.75	32.12	0.99	32.23
Boats	2.00	33.20	1.42	33.14	1.75	33.14	1.21	33.31
Bridge	2.00	30.99	1.96	30.98	1.75	30.95	1.71	31.22
Baboon	2.00	36.90	1.71	36.78	1.75	36.77	1.48	37.09
Lena	2.00	34.90	1.36	34.74	1.75	34.80	1.16	34.96
Barbara	2.00	36.74	1.48	36.53	1.75	36.60	1.28	36.77
Kush	2.00	35.60	1.59	36.41	1.75	35.60	1.37	35.69
Average	**2.00**	**34.36**	**1.53**	**34.38**	**1.75**	**34.27**	**1.31**	**34.47**


(a) Cameraman


(b) Boats


(c) Bridge


(d) Baboon


(e) Lena


(f) Barbara


(g) Kush

Fig.1. Input images taken for the study


(a) Original
bpp : 8.00


(b) Method 1
bpp: 1.42
PSNR: 33.14


(c) Method 2
bpp: 1.75
PSNR: 33.14


(d) Hybrid
bpp: 1.21
PSNR: 33.31

Fig.2. Reconstructed image of Boats


(a) Original
bpp: 8.00


(b) Method 1
bpp: 1.36
PSNR: 34.74


(c) Method 2
bpp: 1.75
PSNR: 34.80


(d) Hybrid
bpp: 1.16
PSNR: 34.96

Fig.3. Reconstructed images of Lena

4. CONCLUSION

The combination of the proposed methods is efficient in reducing the bits required to store the images. The number of bits reduced depends on the nature of the images taken for the study. The proposed method performs well for all types of images. When compared to conventional AMBTC method, the bit-rate is reduced on an average from 2.00 to 1.31 bpp and the PSNR is also raised to 34.47. These methods can also be applied to color images as the gray scale images from the base for color images. The computational complexity involved is less and hence it is easy to implement this method. The proposed method is very well suited for hand-held devices.

REFERENCES

[1] Anil K. Jain, "Image Data Compression: A Review", *Proceedings of the IEEE Conference*, Vol. 69, No. 3, pp. 349-372, 1981.

[2] G. Arce and N. Jr. Gallagher, "BTC Image Coding using Median Filter Roots", *IEEE Transactions on Communication*, Vol. 31, No. 6, pp. 784-793, 1983.

[3] Kruger .A, "Block Truncation Compression", *Dr. Dobb's Journal*, pp.48-55 and pp.104-106, 1992.

[4] Goeddel Tand Bass S, "A Two Dimensional Quantizer for Coding Digital Imagery", *IEEE Transactions on Communications,* Vol. 29, No. 1, pp. 60-67, 1981.

[5] Lema M.D and Mitchell O.R, "Absolute Moment Block Truncation Coding and its Application to Color Images", *IEEE Transactions on Communications*, Vol. 32, No. 10, pp. 1148-1157, 1984.

[6] K. Somasundaram and I. Kaspar Raj, "Low Computational Image Compression Scheme based on Absolute Moment Block Truncation Coding", *World Academy of Science, Engineering and Technoloy*, Vol. 19, pp. 166-171, 2006.

[7] Yung-Gi Wu, "Block Truncation Image Bitplane Coding", *Journal of Optical Engineering*", Vol. 41, No. 10, pp. 2476-2478, 2002.

[8] Yu-Chen Hu, "Predictive Moment Preserving Block Truncation Coding for Gray Level Image Compression", *Journal of Electronic Imaging*, Vol. 13, No. 4, pp. 871-877, 2004.

[9] M.D. Lema and O.R. Mitchell, "Absolute moment block truncation coding and its application to color images", *IEEE Transactions on Communications*, Vol. 32, No. 10, pp. 1148–1157, 1984.

[10] T. Kurita and N. Otsu, "A method of block truncation coding for color image compression", *IEEE Transactions on Communication*, Vol. 41, No. 9, pp. 1270–1274, 1993.

[11] Y. Wu and D.C. Coll, "Single bit-map block truncation coding of color images", *IEEE Journal on Selected Areas in Communications*, Vol. 10, No. 5, pp. 952–959, 1992.