Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 10 (2013) 699 – 706

Procedia Technology

International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013

Image Compression based on Block Truncation Coding using Clifford Algebra

Kartik Sau[a]*, Ratan Kumar Basak[a], Amitabha Chanda[b]

[a]Department of Computer Science and Engineering, Budge Budge Institute of Technology, Kolkata-137, West Bengal, India
[b]Department of Computer Science and Engineering, University of Calcutta, Kolkata – 700 009, West Bengal, India

Abstract

The Block Truncation Coding (BTC) is one of the lossy image compression algorithm in which first and second order moments are preserved in each image block. The present work investigates image compression based on Absolute Moment Block Truncation Coding (AMBTC) and Clifford Algebra. Here we develop a technique to express a positive integer as a sum of largest perfect square of positive integer. The largest square is computed from the given integer, and then the same process is repeated from the residual part of the integer successively. The proposed method gives very good performance in terms of PSNR values when compared to the conventional BTC and AMBTC. To assess image quality some parametric measures bring into service such as: Peak Signal to Noise Ratio (PSNR), Weighted Peak Signal to Noise Ratio (WPSNR), Bit Rate (BR), and Structural SIMilarity Index (SSIM). The comparative result shows that proposed algorithm is the better than BTC and AMBTC.

Keywords: Image Compression; BTC; AMBTC; PSNR; WPSNR; SSIM

1. Introduction

Persons who use digital image processing techniques usually need to control an ample volume of data. It refers to a need of large amount of storage space to keep this large amount of data for future work. The objective of an image

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000 .
E-mail address: kartik_sau2001@yahoo.co.in

2212-0173 © 2013 The Authors. Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
Selection and peer-review under responsibility of the University of Kalyani, Department of Computer Science & Engineering
doi:10.1016/j.protcy.2013.12.412

compression technique is to represent an image with smaller number of bits without introducing appreciable degradation of visual quality of decomposed image. The image compression techniques classified into two categories namely Lossy compression technique and Loss-less compression technique [1]. In lossless compression there is no difference between the reconstructed image and original image at each pixel. On the other hand, lossy compression the reconstructed image contains degradations relative to the original image. The quality of compressed image is good in loss-less compression techniques but compression ratio is less whereas lossy compression techniques provides less image quality with higher compression ratio. Block Truncation Coding and JPEG are lossy image compression technique. Block truncation coding (BTC) is a simple lossy image compression technique to compress monochrome image data, introduced by Delp and Mitchell [2]. It achieves 2 bits per pixel (bpp) with low computational complexity. The key idea of BTC is to perform moment preserving quantization for each non overlapping block of pixels so that the image quality will remain acceptable and at the same time the demand for the storage space will decrease. The quantizer of the original algorithm of BTC retains standard arithmetic mean and standard deviation for each non overlapping block as one bit quantized output. The statistical overhead is that each block needs to keep mean and standard deviation. Another image compression technique is the absolute moment block truncation coding (AMBTC) [3]. It retains the higher mean and lower mean of the non overlapping blocks as quantized output. In this paper, we have proposed a method based on Absolute Moment Block Truncation Coding (AMBTC) and Clifford Algebra [4]. It provides the better image quality than BTC and AMBTC. This paper is organised as follows. Section II describes algorithm of BTC. Section III describes our proposed method. Section IV describes parametric measures for image quality measurements. Section V introduces experimental results and section VI gives conclusion.

2. BTC Algorithm

Block truncation coding (BTC) is a popular lossy moment preserving quantization method for compressing digital images. It preservers first and second moments [5] of each block.

The BTC algorithm involves the following steps:

Step 1: The original image of size M × N is divided into non overlapping blocks (C) of the size m × m (let m = 4) and each block is processed separately. Let $f(x_i)$, $x_i \in C$ is the original intensity of the pixel where C represent the set of coordinate of the pixel of the image plane in that block. Also let $m^2=k$.

Step 2: Based on pixel intensities, the block C is partitioned into two sets of pixels C_0 and C_1 such that $C = C_0 \cup C_1$ and $C_0 \cap C_1 = \phi$ where $C_0 = \{x'_1, x'_2, x'_3, \ldots\ldots x'_{k'}\}$ and $C_1 = \{x''_1, x''_2, x''_3, \ldots\ldots x''_{k-k'}\}$. This partition is represented by assigning one of two levels; say 0 and 1 to the pixel. Thus the partition can be represented as the bit pattern of 0's and 1's based on the following principle:

$$C_0=\{0:f(xi) < m1\} \quad (1)$$
$$C_1=\{1:f(xi) \geq m1\}$$

Where m_1 represent the first order moment i.e.

$$m_1 = \frac{1}{k}\sum_{i=1}^{k} f(x_i) \quad (2)$$

Step 3: The block matrix (B) can be calculated from the C_0 and C_1 described as follows for each pixel of that block.

$$B = \begin{cases} 1 & x_i \geq x_{th} \\ 0 & x_i \prec x_{th} \end{cases} \quad (3)$$

Here 'th' is considering as m_1 which can be define by the equation (2).

Step 4: The block matrix (B), mean ($\overline{x}$) and σ have to be send for each block. Here σ represents standard deviation of a block.

Step 5: To reconstruct the image the pixel marked by 0 will be given the value 'a' and that marked by 1 will be given the value 'b'. The values 'a' and 'b' satisfy the following relation.

$$\left.\begin{aligned} km_1 &= ka + (k-k^{'})b \\ km_2 &= ka^2 + (k-k^{'})b^2 \end{aligned}\right\} \quad (4)$$

Solving the above equation (4) we get

$$\left.\begin{aligned} a &= m_1 - \sigma\sqrt{\frac{k}{k-k^{'}}} \\ b &= m_1 + \sigma\sqrt{\frac{k-k^{'}}{k}} \end{aligned}\right\} \tag{5}$$

For each block m1, σ, k, k′ is known to us from which we can compute the value of 'a' and 'b' where k′ is the number of 0's in the block matrix and k-k′ is the number of 1's in the block matrix. This k′ number of 0's are replaced by the value of 'a' and k-k′ number of 1s are replaced by the value of 'b'. If the block size is 4 × 4 then it will give the 32 bit compressed data (Size of the block matrix is 16 bits. For holding mean and SD is 8bits each) and hence the bit rate is 2bpp. For each block we have to compute the value of a and b. So it is time consumed. To overcome the above mentioned problem D' Lema and Robert Mitchell introduced the concept of Absolute moment block truncation. But AMBTC does not give the satisfactory result. Here we introduce a new concept based on Clifford algebra [6]. The experimental results show our proposed method is better than BTC and AMBTC.

Table. 1 The encoded and decoded image block.

2	19	12	15
2	11	11	9
2	3	12	15
3	3	4	14

$\bar{x}$=8.56
σ=5.59

0	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1

q=9

2	13	13	13
2	13	13	13
2	2	13	13
2	2	2	13

a=2.22
b=13.49

3. Proposed Method

To overcome the different difficulties in the above mentioned method, here we present a new technique based on Clifford Algebra [7] which was proposed by A. Chanda [8]. In this technique we express a positive integer as a sum of largest perfect square of positive integer. The largest square is computed from the given integer, and then the same process is repeated from the residual part of the integer successively.

Step 1: The image of size M × N in pixels is divided into sub-images so that the image is of smaller non overlapping blocks of the size m × m (normally 4 × 4 pixels) and then each block will process separately.

Step 2: Calculate the average gray level ($\bar{x}$) for each block of the whole image in concern $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ or, $\bar{x} = \frac{1}{n}(x_1 + x_2 + x_3 + \ldots x_n)$, we have modified this definition of average by introducing a kind of associative algebra named as Clifford Algebra which generalize the real numbers, complex numbers, quaternion's and several other hyper-complex number systems. The theory of Clifford algebras is intimately connected with the theory of quadratic forms and orthogonal transformations. To calculate the mean of a set of numbers we follow the following steps.

Step 2.1: Represent every element (say n) of that set into smallest possible set of summation of squared integer terms and create an integer vector in descending order. Those integer numbers are must belongs to 15 to 0.

Step 2.2: Choose the element which has maximum occurrence from n vectors index-wise and if the possibility of occurrence of each element is same then choose the element which is minimum valued integer.

Step 2.3: Create a final vector of integers in descending order or ascending order by the elements returned from step 2.2.

Step 2.4: Calculate summation of square of each element of final vector.

Step 2.5: The new mean named as C-mean represented by $\tilde{x}$ will be calculated with respect to the mean from formal definition and the result from step 2.4. The definition of C-mean is as follows:

$$\tilde{x} = avg + \alpha\left(avg - vs\right)$$

$$where,\ avg = \frac{1}{n}\sum_{i=1}^{n} x_i \ \ and$$

$$vs = sum\ of\ final\ vector\ elements$$

$$\alpha = 0.267$$

As an illustration suppose the values are {209, 168, 98, 96, 105, 202, 146, 92, 103, 107, 190, 115, 95, 95, and 109}

$209=14^2+3^2+2^2+0^2+0^2$	$202=14^2+2^2+1^2+1^2+0^2$	$190=13^2+4^2+2^2+1^2+0^2$
$168=12^2+4^2+2^2+2^2+0^2$	$146=12^2+1^2+1^2+0^2+0^2$	$115=10^2+3^2+2^2+1^2+1^2$
$098= 9^2+4^2+1^2+0^2+0^2$	$092= 9^2+3^2+1^2+1^2+0^2$	$095= 9^2+3^2+2^2+1^2+0^2$
$096= 9^2+3^2+2^2+1^2+1^2$	$103=10^2+1^2+1^2+1^2+0^2$	$095= 9^2+3^2+2^2+1^2+0^2$
$105=10^2+2^2+1^2+0^2+0^2$	$107=10^2+2^2+1^2+1^2+1^2$	$109=10^2+3^2+0^2+0^2+0^2$

So, desired value of column 1 = 10^2, column 2 = 3^2, column 3 = 1^2, column 4 = 1^2, column 5= 0^2

Therefore, sum of final vector elements $vs = 10^2 + 3^2 + 1^2 + 1^2 + 0^2 = 111$. $\tilde{x}$ = 120.625+0.267(120.625-111) = 123.1949. As the value of α is selected cleverly from various experimental results the c-mean deviates less from the original mean and it implies better PSNR.

Step 3: Quantization levels are then calculated and classified into two ranges of values. Higher c-mean is the c-mean of the set C_0 of those gray levels which are greater than or equal to $\tilde{x}$ and remaining C_1 are used to make Lower c-mean. Where $C = C_0 \cup C_1 \ \ and \ \ C_0 \cap C_1 = \phi$

The higher c-mean represented by X_H and lower c-mean represented by X_L are calculated as follows:

$$x_H = cmean\left(\left\{f\left(x_i\right) : f\left(x_i\right) \geq cth; f\left(x_i\right) \in C\right\}\right) \tag{6}$$

$$x_L = cmean\left(\left\{f\left(x_i\right) : f\left(x_i\right) < cth; f\left(x_i\right) \in C\right\}\right) \tag{7}$$

$$cth = cmean\left(\left\{f\left(x_i\right) : f\left(x_i\right) \in C\right\}\right) \tag{8}$$

Step 4: Let 'cth' is the threshold value. If the pixel value $f(x_i)$ is less than quantization threshold 'cth' then pixel values are quantized to 0 otherwise quantized to 1. A binary block named B is used to represent these quantization values of each pixel.

$$B = \begin{cases} 1 & f\left(x_i\right) \geq cth \\ 0 & f\left(x_i\right) < cth \end{cases} \tag{9}$$

Step 5: The block matrix (B), $x_H \ \ and \ \ x_L$ have to be send for each block.

Step 6: In the decoder each image block is reconstructed by replacing "1"s in the bit plane with x_H and the "0"s with x_L.

$$X = \begin{cases} x_L & B = 0 \\ x_H & B = 1 \end{cases} \tag{10}$$

In the proposed method we need 16 bit to code the bit plane which is same as the BTC but our proposed method required less calculation than BTC in the decoder side. Total number of bits required for a block is 32 and bit rate is 2 bpp

4. Image Quality Measure

There are several kind of parametric measure to measure the quality of an image which plays important role in various image-processing applications. The parametric measures of image quality are broadly classified into two classes - one is subjective measure and another is objective measure. In this paper we will concentrate in objective measures [9]-[10] such as Peak Signal to Noise Ratio (PSNR), Weighted Peak Signal to Noise Ratio (WPSNR), Bit Rate (BR), and Structural SIMilarity (SSIM) index. When an image compression algorithm has been implemented and designed to a system, it is important to be able to calculate its performance and this evaluation process should be done in such a manner that various kinds of compression techniques can compare their above mentioned measures.

1.1 Peak Signal to Noise Ratio (PSNR)

The peak signal-to-noise ratio (PSNR) is the ratio between the maximum possible power of a signal and the point wise difference between the original image and the processed image. PSNR is usually represented in terms of the logarithmic decibel scale. Thus PSNR is defined as follows:

$$PSNR = 10\log\left(\frac{L^2}{MSE}\right) \text{ dB} \tag{11}$$

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[y(i,j) - x(i,j)\right]^2 \tag{12}$$

Where L is the dynamic range of the pixel value. M and N is the dimension of the greyscale image. Where x(i,j) and y(i,j) represents original image and processed image. If reference image and processed image is very much closed to each other then the MSE will be small and the PSNR will be large, otherwise the PSNR value will be small. PSNR value may range from close to 0 and up to 100 dB. If the reference image and reconstructed image are identical to each other then PSNR value will be 100 dB, otherwise it will be less than 100 dB.

1.2 Weighted Peak Signal to Noise Ratio (WPSNR)

The weighted PSNR (WPSNR) is a different quality metric that was suggested in [11]. It uses an additional parameter call the Noise Visibility Function (NVF). The value of NVF ranges from close to zero and up to one. The WPSNR can be defined as follows:

$$WPSNR = 10\log\left(\frac{L}{\sqrt{MSE}\times NVF}\right)^2 \text{ dB} \tag{13}$$

Where $NVF = \frac{1}{rc}\sum_{i=0}^{r}\sum_{j=0}^{c} NVF(i,j)$ (14)

$$NVF(i,j) = \frac{1}{1+\theta\sigma_x^2(i,j)} \tag{15}$$

$$\sigma_x^2(i,j) = \frac{1}{(2l+1)^2}\sum_{m=-l}^{l}\sum_{n=-l}^{l}\left(x(i+m,j+n) - \overline{x}(i,j)\right)^2 \tag{16}$$

$$\theta = \frac{D}{\sigma_{x\max}^2} \tag{17}$$

Where $\sigma_{x\max}^2$ is the maximum local variance of a given image in a window centred on the pixel with coordinates (i, j) and $D \in [50,100]$ is a determined parameter. If the reference image and reconstructed image are identical to each other then WPSNR will be 100 dB, otherwise it will be less than 100 dB.

1.3 Structural SIMilarity (SSIM) Index

The structural similarity (SSIM) index [12] is another metric, which is used to determine the similarity between reference image and reconstructed image. The SSIM index measures the image quality based on an initial uncompressed image as reference. SSIM index is calculated as follows:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{18}$$

Where $\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$ (19)

$$\sigma_x = \left(\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_x)^2\right)^{1/2} \tag{20}$$

$$\sigma_{xy} = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - \mu_x\right)\left(y_i - \mu_y\right) \tag{21}$$

$C_1 = \left(k_1 L\right)^2, k_1 \ll 1$ and $C_2 = \left(k_2 L\right)^2, k_2 \ll 1$

1.4 Bit Rate (BR)

The performance of image compression schemes can be specified in terms of compression efficiency. Compression efficiency is measured by the compression ratio or by the bit rate. Compression ratio is the ratio of the size of original image to the size of the compressed image; the bit rate is the number of bits per pixel required by the compressed image. CR = size of original image / size of the compressed image. The compression ratio and bit rate are related. Let b be the number of bits per pixel (bit depth) of the uncompressed image, CR the compression ratio, and BR the bit rate. The bit rate is given by

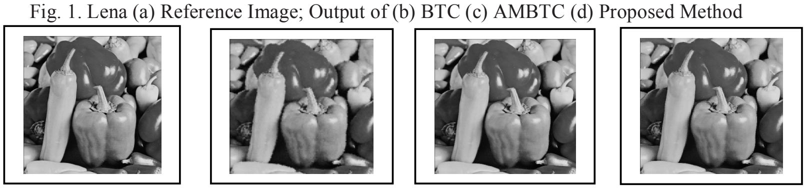
$$BR = \frac{b}{CR} \tag{22}$$

5. Experimental Results

Six test images of size 512 × 512 namely Lena, Peppers, Boat, Tank, Bridge and House are used to make a comparative study between different compression techniques. To compare different compression methods we are using three parameters namely PSNR, WPSNR and SSIM with the block size of 5 × 5. These values are listed in Table 2. From the parametric value of Table 2 it is sure that AMBTC provides better compression results compare with BTC. In the Fig. 1 ,2, 3, 4, 5 and 6 shows reference image and compressed image using BTC, AMBTC and Proposed Method.

a b c d

Fig. 1. Lena (a) Reference Image; Output of (b) BTC (c) AMBTC (d) Proposed Method



a b c d

Fig. 2. Peppers (a) Reference Image; Output of (b) BTC (c) AMBTC (d) Proposed Method

a b c d

Fig. 3. Boat (a) Reference Image; Output of (b) BTC (c) AMBTC (d) Proposed Method

a b c d

Fig. 4. Tank (a) Reference Image; Output of (b) BTC (c) AMBTC (d) Proposed Method

a b c d

Fig. 5. Bridge (a) Reference Image; Output of (b) BTC (c) AMBTC (d) Proposed Method

a b c d

Fig. 6. House (a) Reference Image; Output of (b) BTC (c) AMBTC (d) Proposed Method

Table 2. Comparative Study among BTC, AMBTC and Proposed Method.

Images	Parameters	BTC	AMBTC	Proposed Method
Lena	PSNR	64.3859	72.8041	**73.184662**
	WPSNR	87.1948	94.0578	**94.086266**
	SSIM	0.9775	0.9907	**0.990976**
Pepper	PSNR	62.8157	72.5370	**72.849098**
	WPSNR	85.8155	92.4189	**92.422546**
	SSIM	0.9824	0.9934	**0.993540**
Boat	PSNR	60.5091	69.0160	**69.307755**
	WPSNR	83.1132	90.0319	**90.079124**
	SSIM	0.9626	0.9849	**0.985224**
Tank	PSNR	65.8189	73.2537	**73.405823**
	WPSNR	91.38882	98.1495	**98.107323**
	SSIM	0.9635	0.9833	**0.983780**
Bridge	PSNR	55.5984	63.6956	**63.816933**
	WPSNR	85.4379	92.2467	**92.068092**
	SSIM	0.9552	0.9812	**0.981345**
House	PSNR	59.3568	67.9825	**68.127289**
	WPSNR	87.6878	94.9401	**94.547020**
	SSIM	0.9598	0.9838	**0.983911**

6. Conclusion

Image compression using block truncation coding has been shown in Table 2 that parametric measure of proposed method outperforms than BTC and AMBTC algorithm. Original Block Truncation Coding (BTC), Absolute Moment Block Truncation Coding (AMBTC) and Proposed method these algorithms all are based on dividing the whole image into non-overlapping blocks and the use two quantization level for each block. In our database all reference images are gray scale image of the size of 512 × 512 and 256 gray levels. Parametric measures are used to compare and assess the image quality. In this paper, the Clifford algebra is incorporated in the conventional AMBTC method and this has lead to improvement in the PSNR and WPSNR of the reconstructed images. The value of alpha is chosen so cleverly so that PSNR of the reconstructed images give promising results. Both in terms of PSNR and WPSNR, Proposed Method outperforms the other techniques.

References

[1] Rafale C. Gonzalez, Richard Eugene. *Digital Image Processing*. Pearson, Edition 3, 2012.

[2] E.J. Delp, O.R. Mitchell, Image Compression using Block Truncation Coding. *IEEE, Trans. Communications,* Vol . 27, pp.1335-1342, September 1979.

[3] Maximo D.Lema and O.Robert Mitchell, Absolute Moment Block Truncation Coding and its Application to Color Images. *IEEE Transactions on Communications*, Vol. COM-32, NO. 10, October 1984.

[4] Lounesto P. *Clifford Algebras and Spinors*. Cambridge University Press, Cambridge, 1997.

[5] Franti P, O. Nevalainen and T. Kaukoranta. Compression of Digital Images by Block Truncation Coding: A Survey. *The Computer Journal,* Vol. 37, No. 4, 1994.

[6] Snygg J. *Clifford algebra: a computational tool for physicists*. Oxford University press, 1997.

[7] Hestenes D. and G. Sobczyk. *Clifford algebra to Geometric Calculus*. D. Riedel publishing Co. Holland. 1984.

[8] Chanda A. A New Signal Processing Tool Developed with the Help of the Clifford Algebra, *Proc. of the 22nd IASTED International Conference on Modelling and Simulation*, pp. 314-321 , 2011, Calgary, Canada.

[9] Eskicioglu A. M. and P.S. Fisher. Image quality measures and their performance. *IEEE Trans. Communications*, vol. 34, pp. 2959-2965, Dec. 1995.

[10] Yamsang N. and S. Udomhunsakul. Image Quality Scale (IQS) for compressed images quality measurement. *Proceedings of the International Multiconference of Engineers and Computer Scientists*, vol. 1, pp. 789- 794, 2009.

[11] Voloshynovskiy S., S. Pereira, A. Herrigel, N. Baumgartner and T. Pun. A Stochastic Approach to Content Adaptive Digital Image Watermarking. *Proceedings of the Third International Workshop on Information Hiding*, pp.211-236, 1999.

[12] Wang Z., A. C. Bovik, R. Sheikh and E. P. Simoncelli, Image quality assessment: From error measurement to structural similarity. *IEEE Transactions on Image Processing*, vol. 13, No. 1, January 2004.

[13] Somasundaram K. and MS.S.Vimala, Efficient Block Truncation Coding: *(IJCSE) International Journal on Computer Science and Engineering.* Vol. 02, No. 06, 2010, 2163-2166.