ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 5, May 2015

An Improved Dot Diffusion Based Block Truncation Coding With Random Class Matrix

Divya Nair D[1], G. Suresh Kumar[2], Binoy Babu M[3]

PG Scholar, Dept. of ECE, Maharaja Prithvi Engineering College, Avinashi, Chennai, Tamilnadu,,India [1]

Assistant professor, Dept. of ECE, Maharaja Prithvi Engineering College, Avinashi, Chennai, Tamilnadu, India[2]

PG Scholar, Dept. of ECE,, Maharaja Prithvi Engineering College, Avinashi, Chennai, Tamilnadu, India[3]

Abstract: Block Truncation Coding (BTC) has been considered a highly efficient compression technique for decades. However, its inherent artifacts, blocking effect and false contour, caused by low bit rate configuration are the key problems. To deal with these, an improved BTC, namely Dot Diffused BTC (DDBTC) is used. In this paper we improved current method of Block Truncation Coding Using Optimized Dot Diffusion. Dot-diffused BTC image compression technique which can yield excellent image quality and processing speed. But the image Quality is depend on class matrix and diffusion kernel which is calculated by co optimization algorithm. . So in our method we generated class matrix randomly, which avoids co optimization algorithm and gives an opportunity to select better class matrix. And there is no need to compute diffusion kernel for each class matrix, Floyd's and Steinberg filter weights are used as diffusion kernel matrix. we also choose median as one bit quantizer threshold. it is proven that our method is 3 times faster than DDBTC and gives better HPSNR.

Keywords: Dot Diffused Block Truncation Coding, Class matrix, Processing time, HPSNR

I. INTRODUCTION

Block truncation coding (BTC),is a technique for image compression. The basic concept of this technique is to divide the original image into many non-overlapped blocks, each of which is represented by two distinct values. When a BTC image is transmitted, a pair of values (2 × 8 bits/block) and the corresponding bitmap which addresses the arrangement of the two values in each block (1 bit/pixel) are required. Since a BTC image normally accompanies with annoying false contour and blocking effect, an improved scheme, namely EDBTC was developed to cope with these two issues. The dot diffusion, was employed to cooperate with BTC to yield the proposed dot-diffused BTC (DDBTC) image compression technique. This method can be considered as a powerful candidate for use in most of the low power image/video codec system.Comparing with the EDBTC, DDBTC have high image quality.

DDBTC can yield excellent image quality and processing speed. But the image Quality is depend on class matrix and diffusion kernel which is calculated by co optimization algorithm. . So in our proposed method we generated class matrix randomly, and there is no need to compute diffusion kernel for each class matrix, Floyd's and Steinberg filter weights are used as diffusion kernel matrix. we also choose median as one bit quantizer threshold. it is proven that our method is 3 times faster than DDBTC and gives better HPSNR.

II. OVERVIEW OF BTC

BTC is a technique for image compression called Block.By using a finite Markov chain model, an exact closed form expression for the transient behaviour of an **M/M/l** queue. BTC algorithm uses a two-level (one-bit) non parametric quantizer that adapts to local properties **of** the image. The quantizer that shows great promise is one which preserves the local sample moments. This quantizer produces good quality images that appear to be enhanced at datarates of 1.5bits/picture element. No large data storage was required. This technique uses nonparametric quantizer adaptive over local regions of the image.

The image will be divided into **4** X 4 pixel blocks, and the quantizer will have 2 levels. If one uses the classical quantization design of Max which minimizes the mean square error, one must know, **Q** *priori,* the probability density function of the pixels in each block. This same knowledge is also required for the absolute error fidelity criteria of Kassam [IO]. Since in general it is not possible to

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 5, May 2015

find adequate density function models for typical imagery. Nonparametric quantizes for our coding schemes, that minimizes either mean square error (denoted MSE) or mean absolute error. After dividing the picture into ***n*** X ***n*** blocks *(n* = 4 for our examples), the blocks are coded individually, each into a two level signal. The levels for each block are chosen such that the first two sample moments are preserved.

In the presence of many channel errors, BTC was superior to other methods of transform coding and Hybrid coding. The mean square error and mean absolute error measure cannot correlated with photo analysts evaluations. In some cases, images with large mean square errors were evaluated higher than images with smaller mean square errors. It should be noted that BTC requires a significantly smaller computational load and much less memory.

III.EDBTC

In EDBTC-based data hiding a high image quality and data payload was used. The traditional BTC [1] divides an original image into many non-overlapped blocks of size $M \times N$, and each block can be processed independently .In this process, the first moment (x^{-}), second-moment $(x2)$, and variance $(\sigma\ 2)$ of the input block are calculated to yield the mean, high-mean, and low-mean for further BTC processing.

The concept of the BTC is to preserve the first- and second-moment characteristics of a block when the original block is substituted by its quantization levels. Thus, the following two conditions are maintained:

$$mx^{-} = (m- q)\ a + qb,$$
$$mx2 = (m- q)\ a\text{^}2 + qb\text{^}2,$$

Where the two variables a and$b$ denote the low-mean and high-mean, respectively. Since BTC is a one-bit quantizer, x^{-} is employed for binarizing the block. The result is called bitmap which is used to record the distributions of the two quantization levels, low-mean and high-mean.

$$yi, j = b, \quad \text{if } hi, j = 1$$
$$yi, j = a, \quad \text{if } hi, j = 0$$

Where hi, j denotes the element of the bitmap (H), and yi, j denotes the element of the compressed BTC image. Since a BTC image normally accompanies with annoying false contour and blocking effect, an improved scheme, namelyEDBTC, was developed to cope with these two issues. The *following* figure shows an example processed by the two methods,BTC and EDBTC, using the Elaine image The EDBTCexploits the inherent dithering property of the to overcome false contour problem. Moreover, the blocking effect can also be eased by its error kernel, since both sides of a boundary between any pair of resulting image blocks being correlation. In a block process, the raster scan path (from left to right and top to bottom) is applied to process each pixel. Compressed results of different BTC methods shown below

Fig 1. Original Elaine image.

Fig 2. Traditional BTC

Fig 3. EDBTC

This error-diffusion algorithm belongs to a family of fundamental algorithms for computer graphics since it can be applied to a large variety of applications where the visual quality of the output with a restricted palette plays an important role. It may be efficiently used in general-purpose visualization, in network-based imaging, in printing, in quantizer attached to compression. The efficiency of our algorithm is based on a deliberately restricted choice of the

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 5, May 2015

distribution coefficients. This algorithm produces fairly satisfactory output.

IV. DOT-DIFFUSED BLOCK TRUNCATION CODING

DDBTC was an improved version of the traditional BTC algorithm, thus the traditional algorithm will be firstly introduced for a better comprehension. Given an Original image of size $P \times Q$, and which is divided into Many non-overlapped blocks of size $M \times N$, then each block can be processed independently and eventually represented by two values. The independent processing property yields the additional excellent parallelism advantage. The first and, second-moment, and the corresponding variance are Obtained by

$$\overline{x} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} x_{i,j},$$

$$\overline{x^2} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} x_{i,j}^2,$$

$$\sigma^2 = \overline{x^2} - (\overline{x})^2,$$

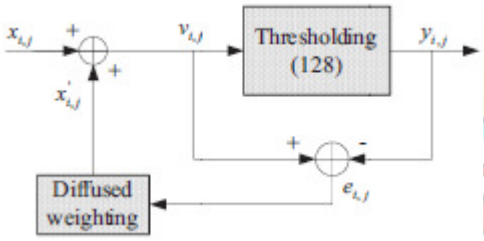

Fig 4. Dot Diffusion algorithm

In DDBTC, the maximum and minimum are obtained by,

Xmax = max (B)
Xmin = min (B)

Where the vector *B* denotes a divided original block. This algorithm has two main differences to that of the traditional BTC: 1) The high mean and low mean are replaced by the local maximum *(x*max*)* and minimum *(x*min*)* in a block, because the high dynamic range (*x*max *–x*min) can easily destroy the blocking effect and false contour, and 2) the manner of bitmap generation is replaced by the dot-diffused half toning.

The Gaussian filter with two parameters defined the standard deviation 1.3 and support region *R* of size 9 ×9, is adopted. According to the experimental results, slight modification of the parameter will not significantly change the results.

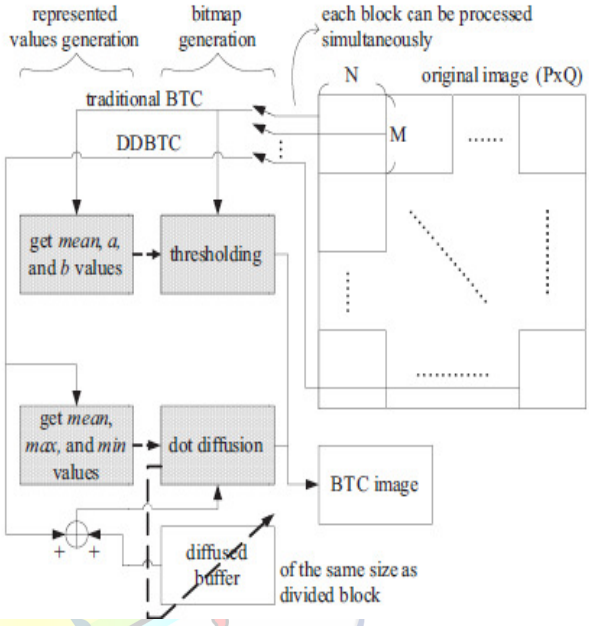

Fig 5. Algorithm Comparison of BTC and DDBTC

In the procedure of the co-optimization algorithm, the Mese-Vaidyanathan's class matrix is adopted as the initial Class matrix (*C*) in this optimization. Suppose the coefficients in the class matrix are collocated as a 1D sequence. Successively swap each member *C (i)*in the class matrix with one of the other 63 members *C (j)*(suppose the size of the class matrix is 8 ×8), where $i \_= j$ Generated all potential diffused weightings $k_{m,n} \in K$ by adjusting 10–6 within a range of 0 to 1. During the generation of diffused weighting, the nearest vertical and horizontal weights are fixed as 1, and the other four diagonals are kept at the same value.

Suppose the original image and the divided block are of sizes $P \times Q$ and $M \times N$, respectively, and each block can be processed independently. For each block, processing order of pixel is defined by class matrix .All of the pixel value associated with number zero in class matrix The equation corresponding to the process is given by,

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 5, May 2015

$$v_{i,j} = x_{i,j} + x'_{i,j}, \quad \text{where } x'_{i,j} = \sum_{(m,n)\in R} \frac{e_{i+m,j+n} \times k_m}{sum}$$

$$e_{i,j} = v_{i,j} - y_{i,j}, \quad \text{where } y_{i,j} = \begin{cases} 0, & \text{if } v_{i,j} < 128, \\ 255, & \text{if } v_{i,j} \geq 128 \end{cases}$$

Where the variable *xi, j* denotes the current input grayscale value, variable *x_i, j* denotes the diffused error accumulated from neighboring processed pixels, and variable *vi, j* denotes the modified grayscale output. The variable *yi, j* denotes the binary output in the bitmap, and variable *ei, j* denotes the difference between the modified grayscale output *vi, j* and the binary output *yi, j*. The variable *km* denotes the diffused weighting, and *R* denotes the support region diffused weighting, with a suggested size of 3 ×3 as in Knuth's and Mese-Vaidyanathan's dot diffusion.
The diffused weighting can be represented as

$$\begin{bmatrix} k_{-1,-1} & k_{-1,0} & k_{-1,1} \\ k_{0,-1} & x & k_{0,1} \\ k_{1,-1} & k_{1,0} & k_{1,1} \end{bmatrix}.$$

The error not only can diffuse to the self-block, but also can Diffuse to its neighboring blocks in DDBTC

V. DDBTC With Random Class Matrix

In the proposed system the class matrix is selected randomly. DDBTC has the disadvantage for each class matrix different Kernel is used. But in the proposed method, We will fix the Kernel matrix first.

The error can only diffuse to neighboring pixels that associates to the numbers in the class matrix with a greater value than its own associated value. These are the pixels that have yet to be threshold. The variable *sum* is the summation of the diffused weights corresponding to those unprocessed pixels.

$$sum = \sum_{m=-1}^{1} \sum_{n=-1}^{1} \begin{cases} k_{m,n}, & \text{if } c_{i+m,j+n} > c_{i,j} \\ 0, & \text{if } c_{i+m,j+n} < c_{i,j}, \end{cases}$$

Where the variable *ci, j* denotes the coefficient value in the class matrix.

In this following example, the central position with number 34 is the current processing position, and those numbers in gray represent the processed pixels.

51	54	60	20	28
43	56	55	19	33
59	41	34	12	62
13	11	2	42	47
22	9	50	61	57

The arrows represent the possible diffusing directions. Since the pixels with numbers smaller than 34 are processed, the total number of diffusing directions is four. For the sake of the co-optimization treatment, the DDBTC achieved even better image quality than that of the EDBTC. Notably, the bigger size in class matrix leads to less parallel advantage.

For example, the whole bitmap of an image can be obtained in 64 unit times when a class matrix of size 8 × 8 is employed, while when the size of the class matrix grows to 16 × 16, then 256 unit times are required to obtain the whole bitmap of an image. Hence, there is a tradeoff between the size of the class matrix and processing efficiency the error not only can diffuse to the self-block, but also can diffuse to its neighboring blocks. In the proposed method, we don't use any algorithms and class matrix is selected randomly.

VI. Results And Discussion

In this proposed system, image compression with high quality has been presented. This approach is based on the random class matrix. Fig. 6 is the original Elaine image

Fig 6. Original Elaine Image

In DDBTC, class matrix is selected according to the kernel matrix, different Kernel was used for different class matrix. So it was complex and the processing time was high and efficiency was low.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 5, May 2015

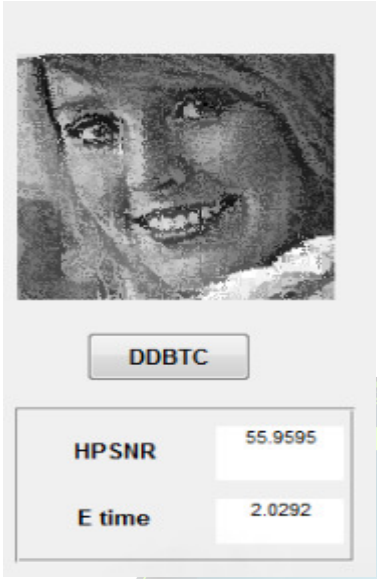

Fig 7. Image after DDBTC

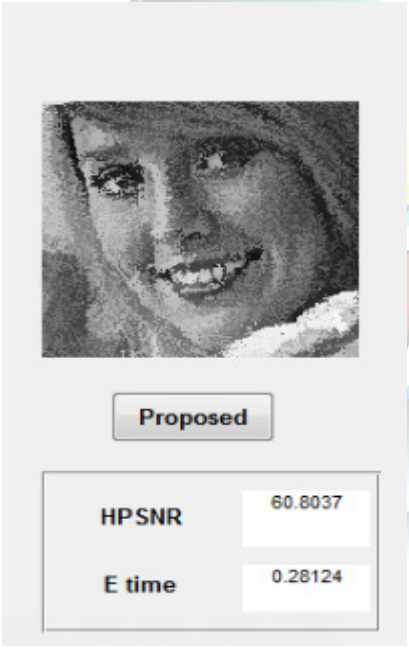

Fig 8. Image after DDBTC with Random class Matrix

In the proposed method random class matrix is selected. Algorithm is not used here. The result can be known easily. Here Kernel is fixed. So processing time will be low and image quality will be high.

VII. CONCLUSION

In this paper we have presented a high image compression technique with random class matrix. We are not using any type of algorithm here. Result can be known easily and image quality is high. The proposed method has HPSNR 60.8037 and the DDBTC has HPSNR of 55.9595.The execution time of proposed method is 0.28124 and the DDBTC has the execution time of 2.0292.The proposed method offers high image quality and high processing efficiency compared to EDBTC and DDBTC. Although the Proposed method provides high image quality with high speed, future work can be put to develop a better algorithm to avoid class matrix and diffused matrix being trapped in local optimal states.

REFERENCES

[1]. J.M Guo and Y.F Liu, 'Improved block truncation coding using optimized dot diffusion,'inProc. IEEE Int. Symp. Circuits Syst 2013.

[2]. L.G Chen and Y.C Liu, 'A high quality MC-BTC codec processing' IEEE Trans. Circuits Syst. Video Technol., vol. 4 2009

[3]. P Stucki , 'MECCA: A multiple-error correcting computation algorithm for bilevel image hardcopy reproduction' IBM Res. Lab., Zurich, Switzerland 2000.

[4]. D,EKnuth ,'Digital halftones by dot diffusion' ACM Trans. Graph.1999

[5]. PMese and P. PVaidyanathan , 'Optimized halftoning using dot diffusion and methods for inverse halftoning' IEEE Trans 2000. ImageProcessing, vol. 9,2000

[6]. P Li and J. P Allebach , 'Block interlaced pinwheel error diffusion,'J. Electron. Imag., vol. 14, 2005

[7]. E.J Delp and O.R Mitchell 'Image compression usingblock truncation coding' IEEE Trans. Commun., vol. 27,2012

[8]. D.R Halverson ,N.C Griswold , and G.L Wise ,'A generalizedblock truncation coding algorithm for image compression' IEEE Trans.Acoust., Speech, Signal Process., vol. 32,2008

[9]. P.Udpikar and J.P Raina, 'Modified algorithm for block truncationcoding of monochrome images' Electron. Lett., vol. 21,2001

[10]. S Horbelt and J Crowcroft , 'A hybrid BTC/ADCT video codecsimulation bench, 1998