International Journal of Pure and Applied Mathematical Sciences.
ISSN 0972-9828 Volume 10, Number 1 (2017), pp. 15-27

http://www.ripublication.com

IMAGE COMPRESSION USING CHEBYSHEV POLYNOMIAL SURFACE FIT

[1]S.Sajikumar and and [2]A.K.Anilkumar

[1] Department of Mathematics, College of Engineering,
Thiruvananthapuram, Kerala, India
E-mail: sajikumar.s@cet.ac.in

[2]Vikram Sarabhai Space Centre, Indian Space Research Organisation,
Thiruvananthapuram, Kerala, India
E-mail: ak_anilkumar@vssc.gov.in

ABSTRACT

A lossy image compression method based on block-by-block surface fit using bivariate polynomial is proposed. Chebyshev polynomials of first kind are used to generate the surface for each block. Data compression is achieved by representing the gray level variations across any block by the parameters of the fitted surface and these parameters are stored instead of the pixel values. Compression with three coefficients and four coefficients in the fit model is proposed. In standard lossy compression techniques compression is achieved by exploiting spatial redundancies of the input data. In this proposed method compression does not depends on redundant information but depends on the block size that can be predefined. The method is best suited for high compression with reasonable reconstructed image quality. Performance was tested on number of test images using Chebyshev polynomials of different orders.

Key Words: Lossy compression, surface fit, Chebyshev polynomial, MSE, PSNR.

1. Introduction

Image compression is essential in effective transmission and storage of image data. The explosion in the amount of image and multimedia data resulted in insufficient bandwidth of network and storage of memory device. Thus image data need to be compressed without or with small loss of quality [1]. If we take a pixel in an image at random there is a good chance that its neighbours will have the same intensity or very similar intensity. Typically hence, image compression is based on the fact that the neighbouring pixels are highly correlated ([2], [3]). Most image compression methods exploit this feature to obtain efficient compression.

Two broad classes of image compression techniques are lossy and lossless. The amount of information retained by an image after compression and decompression is the energy retained. If the energy retained is 100% then the compression is lossless, otherwise it is lossy. Lossless image compression finds application mainly in medical imaging and industrial applications. Lossless compression can be achieved with the techniques like Run Length Encoding (RLE), Huffman coding, Arithmetic coding etc ([4], [5], [6], [7]). Lossy techniques include transform coding methods such as Discrete Cosine Transform (DCT), JPEG, and JPEG2000etc ([8], [9], [10]). The proposed compression algorithm differs from the standard compression algorithms in that it does not rely on two dimensional correlations in the input data.

Lossy compression techniques are tested for their performance based on two commonly used measures, the Mean Squared Error (MSE) and Peak Signal to Noise Ratio (PSNR). The MSE between original image $f(x,y)$ and reconstructed image $\hat{f}(x,y)$ of size $M \times N$ is defined as

$$MSE = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[f(x,y) - \hat{f}(x,y)]^2$$

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right)(dB)$$

For the last forty years vigorous research works have been taken place in the field of data compression in which lossy compression methods got much attention. The state-of-the art coding methods are transform based algorithms [11]. Along with these developments polynomials are also used for image compression ([12], [13]). Two dimensional multiplicative autoregressive model for lossless compression of medical images was first introduced in [14]. Amir Helzer et al. [15] used implicit bivariate polynomial for image c

ompression. Adaptive geometric piecewise polynomial approximation, predictive coding, bivariate linear polynomial approximation ([16], [17], [18]) are the recent trends in image compression. Chebyshev polynomials are used in compression via Chebyshev Transform [19] and row and coloumn based image compression [20].

The organisation of this paper is as follows. Section II briefly introduces the Chebyshev polynomials. Adaptation of Chebyshev polynomials for image compression and polynomial based compression models are described in section III. Section IV presents the compression algorithm and experimental results. A comparison of proposed method with the plane fitting model and the four parameter 'xy' model are given in section V. Finally concluding remarks are given in section VI.

2. Chebyshev Polynomial

The Chebyshev polynomial $T_n(x)$ of degree n is defined by $T_n(x) = \cos(n \arccos x)$ for $x \in [-1, 1]$ and $n = 0, 1, 2, \ldots$. $T_n(x)$ satisfy the Chebyshev differential equation

$(1-x^2)y'' - xy' + n^2 y = 0$ are the Chebyshev polynomials of the first kind.These are orthogonal polynomials of degree n on the interval $-1 \le x \le 1$, with the weight $\dfrac{1}{\sqrt{1-x^2}}$.

That is,

$$\int_{-1}^{1} \frac{T_i(x)T_j(x)}{\sqrt{1-x^2}}\,dx = \begin{cases} 0\,, & i \neq j \\ \dfrac{\pi}{2}\,, & i = j \neq 0 \\ \pi\,, & i = j = 0 \end{cases}$$

The polynomial $T_n(x)$ has n zeros on the interval [-1, 1] at $x_k = \cos\left(\dfrac{(2k-1)\pi/2}{n} \right)$ for

$k = 1, 2, \ldots, n$ .Chebyshev polynomials are used in compression techniques bcause they are orthogonal and numerically stable. $T_n(x)$ are normalised polynomials such that $T_n(1) = 1$.

Chebyshev polynomials are connected by the recurrence relation:

$$T_0(x)=1,$$
$$T_1(x)=x,$$
$$T_{n+1}(x)=2xT_n(x)-T_{n-1}(x) \ \ for\ n>1.$$

First few Chebyshev polynomials are

$$T_0(x)=1,$$
$$T_1(x)=x,$$
$$T_2(x)=2x^2-1,$$
$$T_3(x)=4x^3-3x,$$
$$T_4(x)=8x^4-8x^2+1,$$
$$T_5(x)=16x^5-20x^3+5x.$$

3. Proposed Method

The input image is partitioned into nonoverlapping blocks of predetermined size n $\times$ n

(2 $\times$ 2, 4 $\times$ 4, 8 $\times$ 8, etc). For each block a surface is fitted with a bivariate polynomial of the form

$$P(x,y)=a_0T_0(x)+a_1T_n(x)+a_2T_n(y)$$

with three fit coefficients $[a_0, a_1, a_2]$ or

$$P(x,y)=a_0T_0(x)+a_1T_n(x)+a_2T_n(y)+a_3T_n(x)T_n(y)$$

with four coefficients $[a_0, a_1, a_2, a_3]$, where $T_n(x)$ are Chebyshev polynomials of degree n. A threshold on MSE can be set to determine the degree of the Chebyshev polynomial $T_n(x)$ in the fit model $P(x,y)$. For each block we get fit coefficients which are being stored for reconstruction. Thus compression is achieved. Gray values of each block are recalculated using the stored parameters of the fitted surface. The bivariate linear polynomial

$$P(x,y)=a_0T_0(x)+a_1T_1(x)+a_2T_1(y)$$

is best suited for 2 $\times$ 2 blocks and in this case by storing three coefficients instead of four gray values for each block 25% compression is achieved and the compression ratio (CR) is 75% . But as the number of gray values in a block increases higher order

$T_n(x)$ generally shows better reconstruction quality. For a 4×4 block compressed with three coefficients CR is 81.25% and with four coefficients it is 75%. For an 8×8 block compressed with three coefficients and four coefficients compression ratios are 95.31% and 93.75% respectively. Computation of the fit coefficients can be done in *MATLAB*.

Block artefacts reduces reconstructed imge quality in any block based lossy compression technique and are more visible in highly compressed images. Block artefacts are jumps in the pixel values from one block edge to adjacent block edge in the decompressed image. This can be reduced by replacing adjacent pixel values of neighbouring blocks by their averages.

If $f(x, y)$ is the original image and $g(x, y)$ is the approximated image,

$$r(x, y) = f(x, y) - g(x, y)$$

is the residual image. By applying a lossless compression method RLE or Huffman coding or Arithmetic coding scheme on residual image and adding this with the reconstructed image, the original image can be compressed losslessly.

As the indexing of blocks is done in an ordered manner, it is only necessary to store the coefficients and the reconstruction can consider the ordered storing of coefficients.

4. Algorithm and Experimental Results

Step 1 Load an input gray image of size N $\times$ N.

Step 2 Partion the image matrix into blocks of desired size n $\times$ n.

Step 3 Fit a polynomial surface to each block with a threshold on MSE and store the the fit coefficients for reconstruction.

Step 4 Decode the image using fit coefficients.

Step 5 Reduce block artefacts.

Step 6 Reconstruct the image.

Performance testing of the proposed method is carried out on standard test images. Images of 256 gray levels (8 bits per pixel) of size 256×256 are used. Block sizes 2×2, 4×4 and 8×8 are used in test cases. With 4×4 and 8×8 blocks, compression with four coefficients shows better performance than that with three coefficients. For 4×4 blocks 16 gray values are replaced by 4 coefficients and hence the achieved compression is 75% and compression ratio is 25%. Similarly for 8 x 8 blocks 64 gray values are replaced by 4 coefficients and the achieved compression is 93.75% and compression ratio is 6.25%. Experimental results are given in Table 1, 2 and 3 and figures (1-4).

From the result obtained in Table 1 it is clear that proposed method with three coefficients

shows an increase in PSNR value as compared to the linear fit model

$P(x, y) = a_0T_0(x) + a_1T_1(x) + a_2T_1(y)$ When 4×4 and 8×8 block size are used. Result analysis of Table 2 and 3 shows PSNR is varying with Chebyshev polynomials $T_n(x)$ of different orders with respect to a given threshold on MSE.

5. Comparison with Plane fitting model and $'xy'$ model

In the plane fitting model $z = a + bx + cy$ with three parameters $[a, b, c]$ for fixed block size ([21], [22], [23]), it is given as the best fit model for any block size 2×2 or 4×4 or 8×8. But our results in three parameter case shows that the plane fit model is best suited only for 2×2 blocks. In 4×4 and 8×8 blocks higher order Chebyshev polynomials outperforms the plane model. An increase of 0.4972dB with 4×4 blocks and 0.7714dB with 8×8 blocks in PSNR value is noticed in the case of test image rice. An increase of 0.2426dB with lena and 0.185dB with cameraman is also observed with 8×8 block.

An increase of 0.667dB for rice image is noticed with 8 x 8 block and slight increase in PSNR for other images as compared with the four parameter $'xy'$ model $z = a + bx + cy + dxy$. Details are given in Tables 4 and 5. Results are given without quantization of fit coefficients at the encoding stage and post processing after reconstruction.

Table 1: PSNR and MSE values of Test images for different block size with three coefficients

Polynomial	Block size	Rice		Lena		Cameraman	
		PSNR	MSE	PSNR	MSE	PSNR	MSE
$a_0T_0(x) + a_1T_1(x) + a_2T_1(y)$	2×2	28.9417	83	26.179	157	23.8796	266
$a_0T_0(x) + a_1T_3(x) + a_2T_3(y)$	2×2	28.9417	83	26.179	157	23.8796	266
$a_0T_0(x) + a_1T_5(x) + a_2T_5(y)$	2×2	28.9417	83	26.179	157	23.8796	266
$a_0T_0(x) + a_1T_7(x) + a_2T_7(y)$	2×2	28.9417	83	26.179	157	23.8796	266
$a_0T_0(x) + a_1T_9(x) + a_2T_9(y)$	2×2	28.9417	83	26.179	157	23.8796	266
$a_0T_0(x) + a_1T_1(x) + a_2T_1(y)$	4×4	24.4904	231	23.2617	307	21.3521	476
$a_0T_0(x) + a_1T_3(x) + a_2T_3(y)$	4×4	24.5819	226	23.2438	308	21.3066	481
$a_0T_0(x) + a_1T_5(x) + a_2T_5(y)$	4×4	24.7701	217	23.2612	307	21.3161	480
$a_0T_0(x) + a_1T_7(x) + a_2T_7(y)$	4×4	24.9059	210	23.2835	305	21.3374	478
$a_0T_0(x) + a_1T_9(x) + a_2T_9(y)$	4×4	24.9876	206	23.2985	304	21.3257	476
$a_0T_0(x) + a_1T_1(x) + a_2T_1(y)$	8×8	19.0794	804	19.8798	669	19.011	817
$a_0T_0(x) + a_1T_3(x) + a_2T_3(y)$	8×8	19.3178	761	19.9435	659	19.0365	812
$a_0T_0(x) + a_1T_5(x) + a_2T_5(y)$	8×8	19.7005	697	20.1018	635	19.1549	790
$a_0T_0(x) + a_1T_7(x) + a_2T_7(y)$	8×8	19.8508	673	20.1224	632	19.1960	782
$a_0T_0(x) + a_1T_9(x) + a_2T_9(y)$	8×8	19.3692	752	19.8794	669	19.0006	819

Table 2: PSNR and MSE values of test images for block size 4 x 4 with four coefficients

Polynomial	Rice		Lena		Cameraman	
	PSNR	MSE	PSNR	MSE	PSNR	MSE
$a_0T_0(x) + a_1T_1(x) + a_2T_1(y) + a_3T_1(x)T_1(y)$	24.8358	214	23.6619	280	21.6622	443
$a_0T_0(x) + a_1T_3(x) + a_2T_3(y) + a_3T_3(x)T_3(y)$	24.8672	212	23.5914	284	21.5744	453
$a_0T_0(x) + a_1T_5(x) + a_2T_5(y) + a_3T_5(x)T_5(y$	24.7190	219	23.3895	298	21.4462	466
$a_0T_0(x) + a_1T_7(x) + a_2T_7(y) + a_3T_7(x)T_7(y$	23.0595	321	22.1308	398	20.6176	564
$a_0T_0(x) + a_1T_9(x) + a_2T_9(y) + a_3T_9(x)T_9(y)$	23.226	309	22.2047	391	20.5636	571

Table 3: PSNR and MSE values of test images for block size 8 x 8 with four coefficients

Polynomial	Rice		Lena		Cameraman	
	PSNR	MSE	PSNR	MSE	PSNR	MSE
$a_0T_0(x) + a_1T_1(x) + a_2T_1(y) + a_3T_1(x)T_1(y)$	19.4165	744	20.2515	614	19.2591	771
$a_0T_0(x) + a_1T_3(x) + a_2T_3(y) + a_3T_3(x)T_3(y)$	19.595	714	20.2373	616	19.2533	772
$a_0T_0(x) + a_1T_5(x) + a_2T_5(y) + a_3T_5(x)T_5(y$	19.425	742	19.8754	669	18.9226	833
$a_0T_0(x) + a_1T_7(x) + a_2T_7(y) + a_3T_7(x)T_7(y$	20.0835	638	20.3415	601	19.3473	756
$a_0T_0(x) + a_1T_9(x) + a_2T_9(y) + a_3T_9(x)T_9(y)$	9.8625	6768	10.3452	6006	8.7065	8759

Table 4: Three parameter comparison

Test Image	Block size	Plane Model PSNR(dB)	Proposed Method PSNR(dB)
Rice	4×4	24.4904	24.9876
	8×8	19.0794	19.8508
Lena	4×4	23.2617	23.2985
	8×8	19.8798	20.1224
Cameraman	4×4	21.3521	21.3257
	8×8	19.011	19.1960

Table 5: Four parameter comparison

Test Image	Block size	‘xy' Model PSNR(dB)	Proposed Method PSNR(dB)
Rice	4×4	24.8358	24.8672
	8×8	19.4165	20.0835
Lena	4×4	23.6619	23.5914
	8×8	20.2515	20.3415
Cameraman	4×4	21.6622	21.5744
	8×8	19.2591	19.3473

Figure 1: Original and reconstructed rice image, CR=75%, PSNR= 28.9417dB block size

2 x 2 and 3 coefficients

Figure 2: Original and reconstructed lena image, CR=75%, PSNR=26.179dB, block size

2 x 2 and 3 coefficients

Figure 3: Original and reconstructed cameraman image, CR=75%, PSNR=23.8796dB, block

size 2 x 2 and 3 coefficients

Figure 4: Reconstructed images of rice, lena and cameraman at CR=25%. Block size 4 x 4

and 4 coefficients

6. Conclusion

This paper introduces a simple lossy compression technique based on Chebyshev polynomial with fast decompression and low computational complexity. For a given block size and a threshold on mean squared error compression depends only on the order of the Chebyshev polynomial in the fit model. Simulation results shows an increase in reconstructed image quality as compared to the bivariate linear polynomial fit model $z = a + bx + cy$ with three coefficients and the four coefficient fit model $z = a + bx + cy + dxy$.

.

Acknowledgment

The authors would like to thank Abhay Kumar, Group Director AFDG, VSSC, Trivandrum for encouragement and support and also for providing computational facilities.

REFERENCES

[1] M. A. Joshi, M. S. Raval, Y. H. Dandawate, K. R. Joshi, and S. P. Metkar, Image and Video Compression: Fundamentals, Techniques, and Applications. CRC Press, 2014.

[2] D. Salomon, A concise introduction to data compression. Springer Science & Busi-ness Media, 2007.

[3] K. Sayood, Introduction to data compression. Newnes, 2012.

[4] R. Gallager, “Variations on a theme by Huffman", IEEE Transactions on Information Theory, vol. 24, no. 6, pp. 668-674, 1978.

[5] A. K. Jain, Fundamentals of digital image processing. Prentice-Hall, Inc., 1989.

[6] D. Taubman and M. Marcellin, JPEG2000 Image Compression Fundamentals, Standards and Practice: Image Compression Fundamentals, Standards and Practice. Springer Science & Business Media, 2012, vol. 642.

[7] I. H. Witten, R. M. Neal, and J. G. Cleary, “Arithmetic coding for data compression," Communications of the ACM, vol. 30, no. 6, pp. 520-540, 1987.

[8] W. B. Pennebaker and J. L. Mitchell, JPEG: Still image data compression standard. Springer Science & Business Media, 1992.

[9] R. C. Gonzalez and R. E. Woods, "Digital image processing," Nueva Jersey, 2008.

[10] V. K. Goyal, “Theoretical foundations of transform coding," IEEE Signal Processing Magazine, vol. 18, no. 5, pp. 9-21, 2001.

[11] J. Kovacevic and M. Vetterli, “Transform coding: past, present and future," IEEE Signal Processing Magazine, vol. 18, no. LCAV-ARTICLE-2001-003, p. 6, 2001.

[12] I. Sadeh, “Polynomial approximation of images," Computers & Mathematics with Applications, vol. 32, no. 5, pp. 99-115, 1996.

[13] M. Eden, M. Unser, and R. Leonardi, “Polynomial representation of pictures," Signal Processing, vol. 10, no. 4, pp. 385-393, 1986.

[14] M. Das and S. Burgett, “Lossless compression of medical images using two-dimensional multiplicative autoregressive models," IEEE transactions on medical imaging, vol. 12, no. 4, pp. 721-726, 1993.

[15] A. Helzer, M. Barzohar, and D. Malah, “Using implicit polynomials for image compression," in the 21st IEEE convention of the Electrical and electronic engineers in Israel, 2000. IEEE, 2000, pp. 384-388.

[16] R. Kazinnik, S. Dekel, and N. Dyn, “Low bit-rate image coding using adaptive geometric piecewise polynomial approximation," IEEE Transactions on Image Processing, vol. 16, no. 9, pp. 2225-2233, 2007.

[17] G. Al-Khafaji, “Image compression based on quad tree and polynomial," International Journal of Computer Applications, vol. 76, no. 3, 2013.

[18] G. Al-Khafaji and L. E. George, “Fast lossless compression of medical images based on polynomial," International Journal of Computer Applications, vol. 70, no. 15, 2013.

[19] A. F. Cheng, S. E. Hawkins III, L. Nguyen, C. A. Monaco, and G. G. Seagrave, “Data compression using chebyshev transform," Jul. 24 2007, US Patent 7,249,153.

[20] N. Al-Mutawwa and M. Zemerly, “Image compression using adaptive chebyshev polynomials," WSEAS Trans, pp. 417-422, 2004.

[21] S. Ameer and O. Basir, “Image compression through optimized linear mapping and parametrically generated features," in Communications, Control and Signal Processing, 2008. ISCCSP 2008. 3rd International Symposium on. IEEE, 2008, pp. 867-870.

[22] S. Ameer and O. A. Basir, “A simple three-parameter surface fitting scheme for image compression." in VISAPP (1), 2006, pp. 101-106.

[23] S. Ameer and O. Basir, “Image compression using plane fitting with inter-block prediction," Image and Vision Computing, vol. 27, no. 4, pp. 385-390, 2009.