Color Image Compression Using Polynomial and Quadtree Coding Techniques

Dr. Ban N. Dhannoon
College of Science/Al-Nahrain University/ Iraq
drban_2001@yahoo.com

Dr. Loay E. George
College of Science/Baghdad University/ Iraq
loayedwar57@yahoo.com

Abstract – The aim of this research is to introduce a simple and fast hybrid image color which implies polynomial image technique and quadtree coding. Image blocks are approximated by polynomial functions, and the residue image signal is coded by a hierarchal quantization scheme followed by quadtree coding. Quadtree is applied to encode the nearly sparse blocks of quantized residue signal. The conducted tests of this proposed color image coding scheme have shown encouraging compression performance (in compression with other existing coding schemes). In addition to the good performance aspects the scheme is simple and fast.

KeyWords: polynomial coefficients, quadtree coding, image compression.

1. Introduction

Image coding for elimination of redundancy from the typical highly correlated image waveforms is an active area of research. The objective of image compression is to reduce both spatial and spectral redundancy of the image data in order to be able to store or transmit data in an efficient (i.e., as simple as bits) form [1].

For low bit rate compression applications, segmentation based coding methods generally provide high compression ratios when compared with traditional transform, vector quantization (VQ) and subband (SB) coding schemes.

Quadtree based image compression, which recursively divides the image into simple geometric regions has been one of the most popular segmentation based coding schemes investigated by researchers [2].

2. Previous works

D. M. Bethel and D. M. Monro [3] report a novel image coder which is a hybrid of fractal coding and vector quantization. The approach to image compression is to form an approximate image by one method and clear up errors by another. In this realization, image blocks are approximated by polynomial functions, and the residual image blocks (RIBs) are coded by vector quantization into a code book which is small enough to transmit with an image.

The results are found to be intermediate between fractal and JPEG coding in their rate/distortion performance.

M.F. Fahmy, et.al. [4] devise an algorithm to construct a set of M orthogonal bases along which images can be decomposed. Image compression is achieved through keeping only the coefficients of the linear prediction polynomials, as well as the weights of the decomposition bases, that represent each block of the image. The bases and weights of the singular vectors of the dominate singular values of the image's singular value decomposition are subsequently used in image reconstruction. Simulations results have revealed that the proposed compression scheme, competes very well with compression schemes like JPEG or SPHIT coders.

G. Li and C. Wen [5] present a method for signal reconstruction using orthogonal transform based on discrete Legendre polynomials. They extend the discrete Legendre polynomials to two-dimensional discrete Legendre polynomials for reconstructing and compressing an image. Simulation results illustrate that the error results from compression is usually low with a satisfactory compression ratio by using the proposed method.

L.E. George and B. A. Sultan [6] use polynomial approximation to prune the smoothing component of the image bands produced from decomposing the image signal using wavelet transform. The proposed method give good compression performance while preserving the image quality level.

3. Polynomial Surface Representation

Two dimensional type of polynomials is commonly used as a computer graphics tool to display surfaces with high degree of smoothness [7, 8]. Moreover, polynomial fitting is utilized in two different ways, either for finding the surface that passes through a set of given points (i.e., interpolation) or for finding a surface that passes near a set of given points (i.e., approximation). The polynomial approximation represents the core of our suggested image coding method for reducing the local spatial slow variation component associated with image data.

The idea of using the polynomial representation as a compact technique is to reduce the spatial redundancy existing in the image.

The 2-D signal (I) decomposed into two components (I_1) and (I_2). The first component (I_1) represents the result of approximating the original signal. While, the second component (I_2) represents the residue left by subtracting the polynomial signal (I_1) from the original signal. For compression purpose, the following relevant facts are considered.

1. The polynomial component (I_1) shows a large scale (or low frequency) variation, while the reminder part (I_2) shows short scale (i.e., high frequency) variation. Thus we can say that the polynomial signal represents the smoothed part of the original signal, while the residue represents the fluctuations and spikes that are existed within the original signal.
2. Since the mathematical representation of polynomial component require a few number of bits to describe its coefficients, this lead to the fact that a high compression ratio could be gained if the polynomial formula is used to describe the smoothed part of the signal.
3. The histogram width of the residue component is very narrow in comparison with the original signal. So the average number of bits (i.e., the average codeword length) required to the residue component is significantly smaller than that required for the original data.
4. Therefore, the overall number of bits required to encode the smooth and residue components, in separated manner will be smaller than that required to directly encode the original signal.

It is worth to mention that the residue component mainly consists of edges and those image contents having short wavelength characteristics. Since the human visual system (HVS) shows a poor contrast sensitivity to discriminate the variation which may occur at the locations showing high contrast. Thus, a coarse quantization may be applied on the residue component, to offer a better opportunity to achieve a good compression performance and without big error in image subjective quality.

4. Proposed System Layout

In the first step, the image data is decomposed into (RGB) color components, and then apply one-to-one transformation to give more amenable to an efficient compression than the raw data image.

The color transform is utilized in image compression schemes, because it helpful to reduce the spectral redundancy, also it exploits some of the characteristics of the human vision system to improve the compression performance.

Many compression standards convert (R, G, B) bands to (Y, U, V) bands. In this kind of color model; around 90% of the image information are concentrated into the band (Y), the remaining information will be in the other two bands. And since they holds only 10% of the whole image information, then down sampling them will cause insignificant subjective distortions in the color image.

Figure 1 presents the layout of the proposed compression system applied on each sub-band which consists of two coding techniques, polynomial and quadtree technique, the following steps illustrate each module of the proposed compression system:

a. Decompose the subband into fixed non overlapped blocks of L*L.
b. Extract polynomial coefficients.
c. Perform scalar quantization on the coefficients and save them in a file.
d. Reconstruct a smoothed image from the quantized coefficients.
e. Subtract each smoothed block from the original block to produce the residue.
f. Apply quadtree coding on the residue and save the result in the file.

ISSN 2229-5518

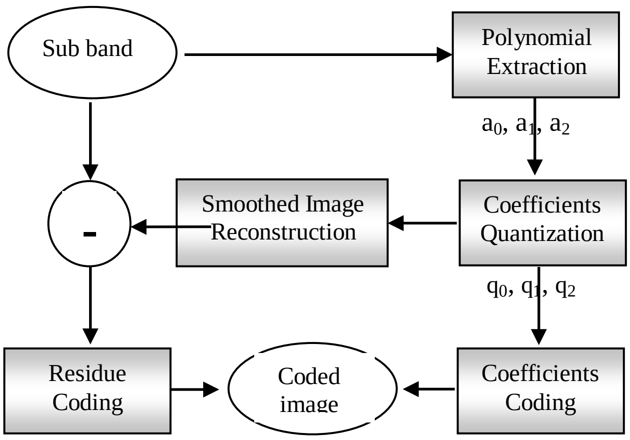


Figure 1. Block diagram of the proposed image coding system

ori

a. Load the quantized coefficients from the file.
b. Reconstruct the smoothed image from the quantized coefficients.
c. Load residue from the file
d. Decode the loaded residue using quadtree.
e. Add the residue with the smoothed image to reconstruct the original image with some loose of its information.

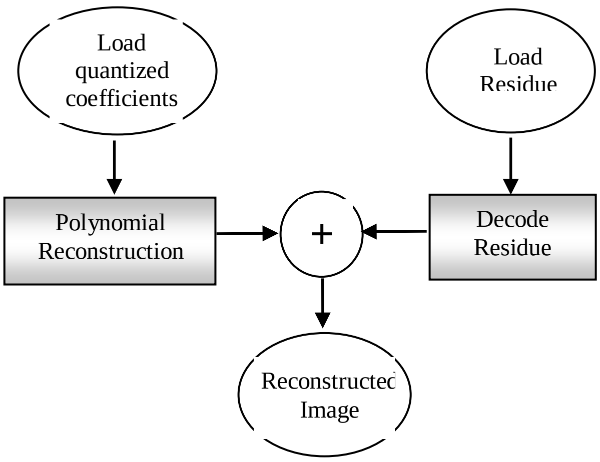


Figure 2. Block diagram of the proposed image decoding system

4.1 Polynomial Extraction

The mathematical formula for representing a 2-D polynomial coefficient of the 1st degree is given as:

$$G(x, y) = a_0 + a_1 x' + a_2 y' \quad \dots (1)$$

Where

$$y_c = x_c = (L-1) / 2 \quad \dots (2)$$

$$x' = (x - x_c) / x_c \quad \dots (3)$$

$$y' = (y - y_c) / y_c \quad \dots (4)$$

L : is the block length.

$$a_0 = \frac{\sum\sum G(x,y)}{L^2} \quad \dots (5)$$

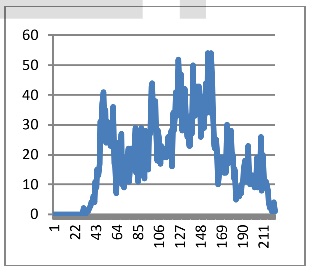
$$a_1 = \frac{\sum\sum x' G(x,y)}{\sum\sum x'^2} \quad \dots (6)$$

$$a_2 = \frac{\sum\sum y' G(x,y)}{\sum\sum y'^2} \quad \dots (7)$$

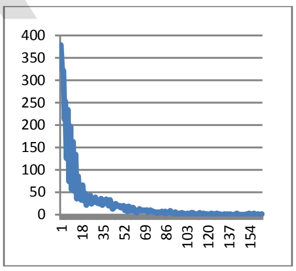
4.2 Polynomial Coefficients

Figure (3) shows the spatial correlation of the polynomial coefficients of the 1st order polynomial surfaces for the image and for block size (4 x 4). It is clear that all the coefficients, except (a_0) are highly peaked and thus, an entropy encoding is suitable for near optimal coding for the polynomial coefficients.

For improving the compression, a delta pulse code modulation (DPCM) was utilized on a_0 to compute the difference of the corresponding polynomial coefficients of the adjacent blocks. The result is more suitable than utilizing the actual values of a_0 coefficients.



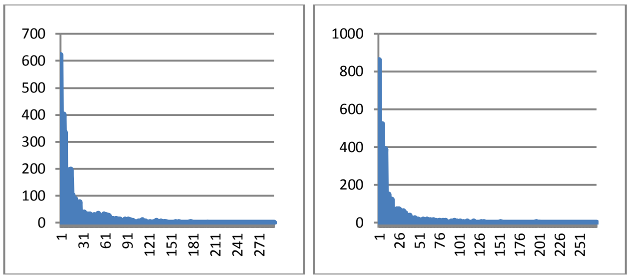
a_0



a_0 after DPCM

P_0,P_1

P_2



a_1 a_2

Figure (3): Spatial correlation of the polynomial coefficients (a_0, a_1, a_2).

4.2.1 Coefficient Quantization

A reduction in stored image size could be gained by discretizing images coarsely, a process called quantization. The quantization process is an irreversible process. Its objective is to reduce the number of code words needed to encode images, at the price of a small amount of distortion.

Over the last decades, the quantizer design has attracted the attention of many workers in the field of image coding. Due to their vast studies, different strategies was derived to deal with the quantizer design problem, most of them are based on objective fidelity criteria (i.e., the mean square error MSR), especially in the case of transform coding, while the little techniques take into consideration the properties of human visual system (HVS) upon the quantization error [9]. The uniform quantizer can be easily specified by its lower bound and step size. Also, the implementation of uniform quantization is easier than non-uniform quantization. In this work, a uniform quantization is applied on polynomial coefficients using equations 8, 9, and 10.

$$q_0 = \frac{a_0 - \mathrm{Min}_0}{\mathrm{stp}_0} \quad \dots (8)$$

$$q_1 = \frac{a_1 - \mathrm{Min}_1}{\mathrm{stp}_1} \quad \dots (9)$$

$$q_2 = \frac{a_2 - \mathrm{Min}_2}{\mathrm{stp}_2} \quad \dots (10)$$

where

$$\mathrm{stp}_0 = \frac{max_0 - \mathrm{min}_0}{2^{\mathrm{bit}_0} - 1}$$

$$\mathrm{stp}_1 = \frac{max_1 - \mathrm{min}_1}{2^{\mathrm{bit}_1} - 1}$$

$$\mathrm{stp}_2 = \frac{max_2 - \mathrm{min}_2}{2^{\mathrm{bit}_2} - 1}$$

4.2.2 Coefficient Coding

After performing uniform quantizer, an entropy coder could be used to get a near-optimal coefficients encoding. In this work, shift coding was utilized to compute the minimum code word length. An efficient shift coding require a code-unit length 2 to 3 bits for the polynomial coefficients except the q_0 coefficients which require 4 to 5 bits code-unit length.

After applying shift coding, the minimum number of bits required for q_0, q_1, q_2 in Y, U, and V band are listed in table (1). And the total number of bytes needed to store the coefficients is 11.4KB.

Table (1) Minimum no. of bits required to code coefficients in Y, U, and V coefficients

Band	q_0	q_1	q_2
Y	5	3	3
U	4	2	2
V	4	3	2

4.3 Smoothed Image Reconstruction

A smoothed image could be reconstructed from the quantized coefficients by applying eq. (11).

$$G2(x, y) = a_{0p}(k) + a_{1p}(k) + a_{2p}(k) \quad \dots (11)$$

where

$$a_{0p} = q_0 * \mathrm{stp}_0 \quad \dots (12)$$
$$a_{1p} = q_1 * \mathrm{stp}_1 \quad \dots (13)$$
$$a_{2p} = q_2 * \mathrm{stp}_2 \quad \dots (14)$$

Then the residue blocks will be constructed by subtracting the original blocks from the reconstructed smoothed blocks.

$$Res\,(x, y) = (G(x, y) - G2(x, y)) / QntR \quad \dots (12)$$

Where

$$QntR = L * (x')^2$$

4.3 Residue Coding

Apply quadtree coding directly on the residue components. The quadtree method scans the residue component, area by area, looking for areas composed of identical pixels (uniform areas).

The output is a tree (a quadtree, where each node is either a leaf or has exactly four children). The size of the quadtree depends on the complexity of the image [10].

Quadtree encoder has adaptive nature and, where the number of necessary computations depends on the image details. The image is partitioned into non-overlapped sub-square blocks.

ISSN 2229-5518

Each block may be broken up into four equal-sized sub-squares when it has some sort of details [11]. This process repeated, recursively, starting from the whole image and continuing until reaching the smallest block size (i.e., 2x2 square) is reached.

5. Experimental Results

Table (2) shows the effect of different quantization parameters used in this work on Lena. Bmp image of size 192KB.

Table (2) The effect of quantization parameters from the PSNR and the compressed file size

Block length	Qnt	Stp_0	Stp_1	Stp_2	PSNR	MSE	File size in KB
2	25	2	3	1	34.604	22.525	63.8
2	25	2	6	1	34.384	23.697	49.59
2	30	2	3	1	34.396	23.63	53.72
2	30	2	6	1	34.191	24.773	49.44
4	15	2	1	1	33.722	27.596	18.64
4	15	2	2	1	33.627	28.209	17.87
4	15	2	4	1	33.348	30.079	17.52
4	15	2	8	1	32.786	34.239	17.91
4	20	1	1	1	6.41	14861.98	17.45
4	20	2	1	1	33.002	32.576	16.69
4	20	2	2	1	32.921	33.187	15.58
4	20	2	3	1	32.810	34.046	15.71
4	20	2	4	1	32.676	35.119	15.43
4	20	2	6	1	32.378	37.611	15.31
4	20	2	8	1	32.079	40.285	15.33
4	20	2	10	1	31.860	42.363	15.54
4	20	2	2	2	25.551	181.14	18.06
4	20	2	4	2	25.392	187.885	17.48
4	20	2	8	2	25.018	204.791	17.41
4	20	4	2	1	7.592	11322.2	15.347
4	25	2	2	1	31.844	42.533	14.53
4	25	2	4	1	31.637	44.601	13.97
4	25	2	6	1	31.402	47.081	13.08
4	25	2	8	1	31.135	50.074	13.73
4	30	2	6	1	30.361	59.836	15.34
4	28	2	6	1	30.373	59.677	15.74
4	30	2	3	1	30.476	58.28	15.79
8	15	2	2	1	17.045	1283.21	18.25
8	20	2	2	1	19.144	791.927	13.35
8	25	2	2	1	20.958	512.55	11.1
8	30	2	2	1	22.448	370.06	9.39

From table (2), we can conclude that the best quantized parameters are block-size=4, QntR=20, stp_0=2, stp_1=2, and stp_2=1 or block-size=4, QntR=25, stp_0=2, stp_1=6, and stp_2=1. Figure (4) shows the original Lena image and its reconstructed image with PSNR=32.921 and the compressed file size = 15.58KB.

Figure (4) the original and the reconstructed image

The proposed compression scheme is fast. The required time for building the compressed an image of size 192KB is 0.54 second, while the reconstructing process takes 0.42 seconds.

6. Conclusions

The test results indicate that the proposed image compression system works in efficient ways. The use of wavelet to encode the data of YUV-bands is necessary to efficiently encode any possible high details may appear in this band.

Also, the RLE encoder which is a simple method and with a little modification, it became very useful to improve the compression performance of the YUV-bands.

References

[1] Ying Chen, Pengwei Hao, ”Integer Reversible Transformation to Make JPEG Lossless”, Center for Information Science, Peking University, Beijing, 100871, China Department of Computer Science, ICSP’04 Proceedings, 2004.

[2] G.J. Sullivan and R.L. Baker, “Efficient quadtree coding of images and video”. IEEE Trans. Image Proc., vol. 3, no. 3, pp. 327-331, May, 1994.

[3] D. M. Bethel and D. M. Monro, "Polynomial image coding with vector quantized error compensation". Acoustics, Speech, and Signal Processing, ICASSP-95., vol. 4, pp. 2499-2502, 1995.

[4] M.F. Fahmy, Y.M.Y. Hasan, and G.M.A. El-Raheem, "Image Compression Via Orthogonal Space Decomposition",

Proceedings of the Twenty Third National Radio Science Conference, pp. 1-8, 2006.

[5] Guoqi Li and Changyun Wen, "Legendre polynomials in signal reconstruction and compression", 5th IEEE Conference on Industrial Electronics and Applications, pp. 1636-1640, 2010.

[6] L.E. George and B.A. Sultan, "Image Compression Based On Wavelet, Polynomial and Quadtree", Journal of Applied Computer Science & Mathematics, no. 11 (5) /2011.

[7] P. Burger and D. Gillies, "Interactive computer graphics: Fuctional, Procedural and Device-level methods", Addison Wesley Publication Company, Wokingham, Reading, England, 1989.

[8] T.P. Pavlidis, "Algorithms for graphics and image processing", Springer-Verlag, Berlin-Heidelberg, 1982.

[9] A. Netravali and J. Limb, "Picture coding: A review", Proc. IEEE, vol. 69, no. 3, pp. 349-389, March 1981.

[10] D. Salmon, "Handbook of Data Compression", 5th ed., 2010.

[11] Y. Fisher, "Fractal Image Compression", Springer Verlag, New York, 1995.