A Novel Image Compression Algorithm based on Discrete Wavelet Transform with Block Truncation Coding

Shyam Lal[1], Mahesh Chandra[2] & Gopal Krishna Upadhyay[3]
[1-]Department of E & C Engineering, Moradabad Institute of Technology, Moradabad-244001(U.P)-India
[2]Department of E & C Engineering, Birla institute of Technology, Mesra-Ranchi(Jharkhand)
[3] Director, SSITM,Kasganj, Moradabad (U.P.) –India
[shyam_rao24@rediffmail.com, shrotriya@bitmesra.ac.in,gkupadhyay2003@yahoo.com]

Abstract- This paper presents a novel image compression algorithm for gray scale image. Digital images contain large amount of information that need evolving effective techniques for storing and transmitting the ever increasing volumes of data. Image compression addresses the problem by reducing the amount of data required to represent a digital image. Image compression is achieved by removing data redundancy while preserving information content. In this paper a simplified and more efficient image compression algorithm is described and implemented. This paper proposed an efficient image compression algorithm which is based on block truncation coding (BTC). Simulation & Experimental results on benchmark test images demonstrate that the new approach attains competitive image compression performance, compared with sate-of-the-art image compression algorithms.

Keywords: Image compression, DWT, SPIHT, Haar Transform and Block Truncation Coding

1. Introduction

The basic idea behind this method of compression is to treat a digital image as an array of numbers i.e., a matrix. Each image consists of a fairly large number of little squares called pixels (picture elements). The matrix corresponding to a digital image assigns a whole number to each pixel. For example, in the case of a 256x256 pixel gray scale image, the image is stored as a 256x256 matrix, with each element of the matrix being a whole number ranging from 0 (for black) to 255 (for white).Image compression is used to minimize the amount of memory needed to represent an image. Images often require a large number of bits to represent them, and if the image needs to be transmitted or stored, it is impractical to do so without somehow reducing the number of bits. The problem of transmitting or storing an image affects all of us daily. TV and fax machines are both examples of image transmission, and digital video players and web pictures of Catherine Zeta-Jones are examples of image storage [1-3].

Image compression is a technique used to reduce the storage and transmission costs. The existing techniques used for compressing image files are broadly classified into two categories, namely lossless and lossy compression techniques. In lossy compression techniques, the original digital image is usually transformed through an invertible linear transform into another domain, where it is highly de-correlated by the transform. This de-correlation concentrates the important image information into a more compact form. The transformed coefficients are then quantized yielding bit-streams containing long stretches of zeros. Such bit-streams can be coded efficiently to remove the redundancy and store it into a compressed file. The decompression reverses this process to produce the recovered image [1-3].

Discrete Cosine Transform (DCT) is a powerful mathematical tool that took its place in many compression standards such as JPEG and MPEG. In the most general form, DCT can be expressed as matrix multiplication. The 2-D discrete cosine transform (DCT) is an invertible linear transform and is widely used in many practical image compression systems because of its compression performance and computational efficiency [1-3]. DCT converts data (image pixels) into sets of frequencies. The first frequencies in the set are the most meaningful; the latter, the least. The least meaningful frequencies can be stripped away based on allowable resolution loss. DCT-based image compression relies on two techniques to reduce data required to represent the image. The first is quantization of the image’s DCT coefficients; the second is entropy coding of the quantized coefficients [4]. Quantization is the process of reducing the number of possible values of a quantity, thereby reducing the number of bits needed to represent it. Quantization is a lossy process and implies in a reduction of the color information associated with each pixel in the image [4 -7].

Haar Wavelet Transform (HWT): The 1D Haar Transform can be easily extended to 2D. In the 2D case, we operate on an input matrix instead of an input vector. To transform the input matrix, we first apply the 1D Haar transform on each row. We take the resultant matrix, and then apply the 1D Haar transform on each column. This gives us the final transformed matrix. The 2D Haar transform is used extensively in efficient image compression, both lossless and lossy [8].

The JPEG compression technique divides an image into 8x8 blocks and assigns a matrix to each block. One can use some linear algebra techniques to maximize compression of the image and maintain a suitable level of detail. JPEG (Joint Photographic Experts Group) is an international compression standard for continuous-tone still image, both grayscale and color. This standard is designed to support a wide variety of applications for continuous-tone images. Because of the distinct requirement for each of the applications, the JPEG standard has two basic compression methods. The DCT-based method is specified for lossy compression, and the predictive method is specified for lossless compression. A simple lossy technique called baseline, which is a DCT-based methods, has been widely used today and is sufficient for a large number of applications. In this paper, we will simply introduce the JPEG standard and focuses on the baseline method [9-10].

The Set Partition in Hierarchical Tree (SPHT) algorithm is unique in that it does not directly transmit the contents of the sets, the pixel values, or the pixel coordinates. What it does transmit is the decisions made in each step of the progression of the trees that define the structure of the image. Because only decisions are being transmitted, the pixel value is defined by what points the decisions are made and their outcomes, while the coordinates of the pixels are defined by which tree and what part of that tree the decision is being made on. The advantage to this is that the decoder can have an identical algorithm to be able to identify with each of the decisions and create identical sets along with the encoder [11].

Wavelet coding is proving to be a very effective technique for image compression, giving significantly better results than the JPEG standard algorithm with comparable computational efficiency. The standard steps in such compression are to perform the Discrete Wavelet Transform (DWT), quantize the resulting wavelet coefficients (either uniformly or with a human visual system weighting scheme), and losslessly encode the quantized coefficients. These coefficients are usually encoded in raster-scan order, although common variations are to encode each sub-block in a raster-scan order separately or to perform vector quantization within the various sub-blocks 12-16].

In this paper we present an efficient image compression algorithm by discrete wavelet transform DWT) block truncation coding (BTC) method. This algorithm gives better image compression performance.

The rest of paper is structured as follows. Section II briefly review the concept of discrete wavelet transform (DWT).Section-III presents the proposed image compression algorithm in detail. Section-IV presents the simulation & experimental results and section-V concludes the paper.

2. Discrete Wavelet Transform

Wavelets are mathematical functions that cut up data into different frequency components, and then study each component with a resolution matched to its scale. They have advantages over traditional Fourier methods in analyzing physical situations where the signal contains discontinuities and sharp spikes. Wavelets were developed independently in the field of mathematics, quantum physics, electrical engineering, and seismic geology. Interchanges between these fields during the last 20 years have led to many new wavelet applications such as image compression, turbulence, human vision, radar, and earthquake prediction. This paper introduces wavelets to the interested technical person outside of the digital signal processing field. Some researchers describe the history of wavelets beginning with Fourier, compare wavelet transforms with Fourier transforms, state properties and other special aspects of wavelets, and finish with some interesting applications such as image compression, musical tones, and de-noising noisy data. The fundamental idea behind wavelets is to analyze according to scale. Indeed, some researchers in the wavelet field feel that, by using wavelets, one is adopting a whole new mindset or perspective in processing data.

The Wavelets are functions that satisfy certain mathematical requirements and are used in representing data or other functions. This idea is not new. Approximation using superposition of functions has existed since the early 1800's, when Joseph Fourier discovered that he could superpose sines and cosines to represent other functions. However, in wavelet analysis, the scale that we use to look at data plays a special role. Wavelet algorithms process data at different scales or resolutions. If we look at a signal with a large “window" we would notice gross features. Similarly, if we look at a signal with a small “window" we would notice small features. The result in wavelet analysis is to see both the forest and the trees, so to speak. This makes wavelets interesting and useful [7][12-16].

Mathematically the Discrete wavelet transform pair for one dimensional can be defined as

$$W_\phi(j_0,k) = \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x)\tilde{\phi}_{j_0,k}(x) \qquad (1)$$

$$W_\psi(j,k) = \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x)\tilde{\psi}_{j,k}(x) \qquad (2)$$

for $j \geq j_0$ and

$$f(x) = \frac{1}{\sqrt{M}} \sum_k W_\phi(j_0,k)\phi_{j_0,k}(x) + \frac{1}{\sqrt{M}} \sum_{j=j_0}^{\infty} \sum_k W_\psi(j,k)\psi_{j,k}(x) \qquad (3)$$

Where $f(x)$, $\phi_{j_0,k}(x)$, and $\psi_{j,k}(x)$ are functions of discrete variable x = 0, 1, 2, ... ,

In two dimensions, a two-dimensional scaling function, $\phi(x,y)$, and three two-dimensional wavelet $\psi^H(x,y)$, $\psi^V(x,y)$ and $\psi^D(x,y)$, are required. Each is the product of a one-dimensional scaling function ϕ and corresponding wavelet ψ.

$$\phi(x,y) = \phi(x)\phi(y) \qquad (4)$$

$$\psi^H(x,y) = \psi(x)\phi(y) \qquad (5)$$

$$\psi^V(x,y) = \phi(y)\psi(x) \qquad (6)$$

$$\psi^D(x,y) = \psi(x)\psi(y) \qquad (7)$$

where ψ^H measures variations along columns (like horizontal edges), ψ^V responds to variations along rows (like vertical edges), and ψ^D corresponds to variations along diagonals. The two-dimensional DWT can be implemented using digital filters and down samplers and it is shown in the Figure 1 & 2, respectively.

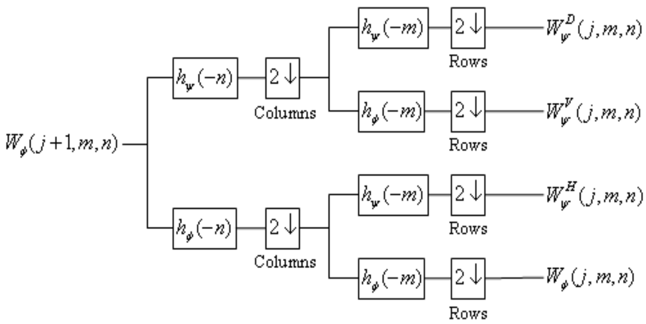


Fig. 1. The two-dimensional DWT— the analysis filter

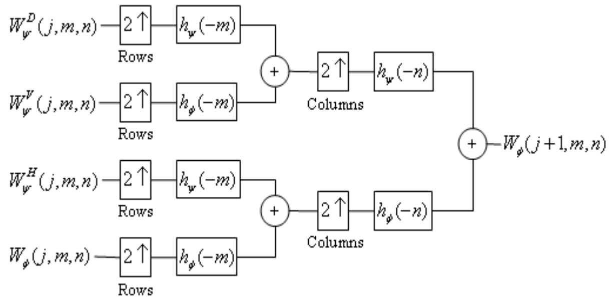


Fig. 2. The two-dimensional DWT— the synthesis filter

III. PROPOSED ALGORITHM

The block diagram of proposed algorithm is given below

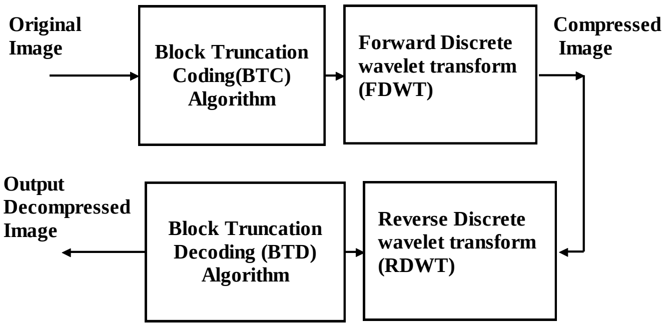


Fig.3. Block diagram of proposed image compression and decompression algorithm

A. Block Truncation Coding & Decoding Algorithm:

The main goal is to transmit or to store only data related to the statistics of the image blocks of the original image using a coding algorithm. A decoding algorithm is also needed to display the image using only the statistics of the blocks. Block Truncation Coding Algorithm (BTC) is simple, fast and essentially robust against transmission errors [17-18].

For the study presented here the image (NXN) will be divided into 4X4 pixel blocks, and the quantizer will have 2 levels. After dividing the image into MXM blocks *(M* = 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. Let $M = N^2$ and let $X_1, X_2, \ldots X_M$ be the values of the pixels in a block of the original picture.

Then the first and second sample moments and the sample variance are respectively

$$\overline{X} = \frac{1}{M}\sum_{i=1}^{M} X_i \tag{8}$$

$$\overline{X^2} = \frac{1}{M}\sum_{i=1}^{M} X_i^2 \tag{9}$$

$$\overline{\sigma}^2 = \overline{X^2} - (\overline{X})^2 \tag{10}$$

As with the design of any one bit quantizer, we find a threshold X_{th}, and two output levels, *A* and *B* ,such that

$$\left.\begin{array}{ll} if\ X_i \ge X_{th} & output = B \\ if\ X_i < X_{th} & output = A \end{array}\right\} \quad for\ i = 1, 2, \ldots\ldots, M$$

For our first quantizer, we set $X_{th} = \overline{X}$ and the output levels *A* and *B* are found by solving the following equations:

Let Q=number of X_i's greater than $X_{th} (= \overline{X})$ then to preserve $\overline{X}$ and $\overline{X^2}$

$$M\,\overline{X} = (M-Q)A + QB \tag{13}$$

and

$$M\,\overline{X^2} = (M-Q)A^2 + QB^2 \tag{14}$$

Solving Equation (13) & (14) for *A* and *B*

$$A = \overline{X} - \overline{\sigma}\sqrt{\left[\frac{Q}{M-Q}\right]} \tag{15}$$

$$B = \overline{X} + \overline{\sigma}\sqrt{\left[\frac{M-Q}{Q}\right]} \tag{16}$$

Where $\overline{\sigma} = \sqrt{\overline{X^2} - \left(\overline{X}\right)^2}$

Each block is then described by the values of $\overline{X}, \overline{\sigma}$ and an *N* X *N* bit plane consisting of 1's and 0's indicating whether pixels are above or below X_{th}. Assuming 8-bits each to $\overline{X}$ and $\overline{\sigma}$ results in a data rate of 2 bits/pixel.

a. The coding Algorithm Implementation procedure:

1. The original image (NxN) is broken down into small blocks of size (MxM) pixels with M << N; usually M = 4.
2. For each block, the mean value $\overline{X}$ and the mean square value $\overline{X^2}$ are computed, as well as $\overline{\sigma}$.
3. Each pixel value of the block is then compared to a threshold value.

$$\left.\begin{array}{ll} if\ X_i \ge X_{th} & output = B \\ if\ X_i < X_{th} & output = A \end{array}\right\} \quad for\ i = 1, 2, \ldots\ldots, M \tag{17}$$

The block is replaced with a block of A's and B's. The image is therefore converted to an image with pixels values equal to A or B.

4. The code table is

$$T(i,j) = \begin{cases} 1 & if\ A \\ 0 & if\ B \end{cases} \tag{18}$$

5. The mean value $\overline{X}$ and $\overline{\sigma}$ are then coded with 8 bits.
6. For each block, $\overline{X}$, $\overline{\sigma}$, and T(i,j) are transmitted.
7. Repeat for each block of the original image.

b. The decoding Algorithm Implementation procedure:

1. The received data, $\overline{X}$, $\overline{\sigma}$, and T(i,j) for each block are converted to real numbers.
2. Then, for each block the values A and B are computed as

$$A = \overline{X} - \overline{\sigma}\sqrt{\left[\frac{Q}{M-Q}\right]} \tag{19}$$

$$B = \overline{X} + \overline{\sigma}\sqrt{\left[\frac{M-Q}{Q}\right]} \qquad (20)$$

The block is then reconstructed as

Y(i,j)=A.I(i,j)+(B-A).T(i,j) (21)

for i=1,2,3,4 & j=1,2,3,4.

Where I(i,j) is the identity block.

3. Repeat for each block received.
4. The last step is to rebuild the whole image from the reconstructed blocks.

B. DWT Image Compression Algorithm :

The idea behind this method is that further it performs image compression very efficiently and gives higher compression ratio and better image qualities.

The following implementation steps have been made for the image compression using discrete wavelet transform, which is based on 2-D-wavelet.

- Reading coded image from the output of BTC
- Decomposition of images using wavelets for the level N.
- Selecting and assigning a wavelet for compression.
- Generating threshold coefficients using Birge-Massart strategy.
- Performing the image compression using wavelets.
- Computing and displaying the results such as compressed image, retained energy and Zero coefficients.
- Decompression of the image based on the wavelet decomposition structure.
- Plotting the reconstructed image.
- Computing and displaying the size of original image, compressed image and decompressed image.

3. Simulation & Experimental Results

The 8-bit images of dimensions MlxM2 (= 256x256) pixels is used for simulation. For testing the performance of proposed image compression algorithm, the following test images shown in Fig.4 were used for experiments.

Fig.4.(A).Lena

Fig.4.(B). Bridge

Fig.4.(C). Clown

Fig.4.(D) .Couple

Fig.4.(E).Fruits

Fig.4.(G).House

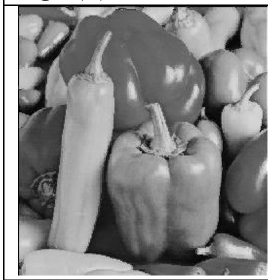


Fig.4.(H).Peppers

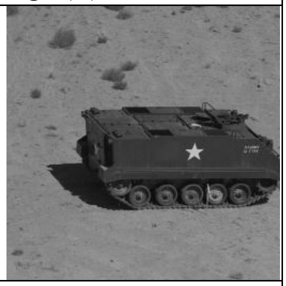


Fig.4.(I).Tank

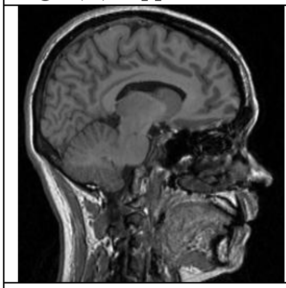


Fig.4.(J).Brain

Fig.4.(K).Cameraman

The proposed image compression algorithm can be viewed as completion and extension of discrete wavelet transform.We compared performance of proposed algorithm with five state-of-the-art image compression: Haar wavelet transform, JPEG, SPIHT and DWT. Test gray –sacle image(size:256X256) used in our experiments. We Evaluated and compared performance of different image compression algorithms by using three measures: Compression ratio(CR), Peak signal to noise ratio(PSNR),and mean squared error(MSE). Although compression ratio measure the rate of image compression and PNSR can measure the intensity difference between two images, It is well known that it may fail to describe the visual perception quality of image. The superiority of proposed algorithm is demonstrated by conducting two experiments. Compression ratio (CR) is defined in equation (22) and peak signal to noise ratio (PSNR) in dB as defined in equation (23) and Mean squared error (MSE) are the metrics used to compare the performance of proposed image compression algorithm with existing algorithms.

1. Compression Ratio: The image compression ratio is defined as

$$CR = \frac{Number\ of\ bits\ in\ Original\ image}{Number\ of\ bits\ in\ Compressed\ image} \qquad (22)$$

2. Peak signal to noise ratio (PSNR): The PSNR between the filtered output image y(i, j) and the original image s(i, j) of dimensions MlxM2 pixels is defined as:

$$PSNR = 10 * \log_{10}\left(\frac{MAX_I^2}{\sqrt{MSE}}\right) \qquad (23)$$

Where MAX_I the maximum pixel value of the image and MSE is is is mean squared error and it is defined as

$$MSE = \frac{\sum_{i}^{M1}\sum_{j}^{M2}[y(i,j)-s(i,j)]^2}{M1 \times M2} \qquad (24)$$

A. Experiment 1:

Table–I gives the image compression performance in terms of compression ratio. It can be seen from Table-I that proposed algorithm gives higher compression ratio as compared to existing lossy compression techniques. Table-II gives the image compression performance in terms of peak signal to noise ratio (PSNR) in dB. It can be seen from Table-II that proposed algorithm gives higher PSNR as compared to HWT & DWT and lower as compared to JPEG & SPIHT. Table –III gives the image compression performance in terms of mean squared error (MSE).It can be seen from Table-III that proposed algorithm gives lower MSE as compared to HWT & DWT and higher as compared to JPEG and SPIHT.
So we have to trade-off between compression ratio and Peak to signal to noise ratio (PSNR).

B .Experiment 2:
To visualize the image quality of decompressed image of proposed image compression algorithm is compared with existing image compression techniques such as HWT,JPEG, SPIHT and DWT. Fig.5,6,7,8,9,10,11,12,13,and 14 gives visual appearance of Lena, Bridge, Fruits, Clown, Couple, House, Peppers, Tank, Brain and Cameraman by HWT, JPEG, SPIHT, DWT and proposed algorithm.

Table-I. Compression Ratio performance of various image compression algorithms

Method/Image	HWT	JPEG	SPIHT	DWT	Proposed
Lena(256X256)	5.73	35.17	53.20	50.73	56.33
Bridge(256X256)	3.00	52.50	53.80	59.00	59.18
Clown(256X256)	5.94	52.80	54.43	54.60	55.27
Fruit(256X256)	11.48	38.57	49.73	50.79	50.94
Couple(256X256)	5.94	40.80	44.11	52.35	53.48
House(256X256)	9.20	40.53	40.53	40.53	40.63
Peppers(256X256)	6.43	31.9	36.34	39.40	42.50
Tank(256X256)	8.80	25.10	30.21	33.04	33.81
Brain(256X256)	5.04	12.38	8.75	36.65	37.16
Cameraman(256X256)	4.28	14.74	30.65	34.85	39.13

Table-II. PSNR performance of various image compression algorithms

Method/Image	HWT	JPEG	SPIHT	DWT	Proposed
Lena(256X256)	24.26	29.36	35.79	26.43	27.73
Bridge(256X256)	23.00	26.16	28.32	23.56	24.90
Clown(256X256)	24.26	30.00	35.02	25.31	26.77
Fruit(256X256)	23.81	32.71	38.57	29.61	31.16
Couple(256X256)	23.81	28.70	33.36	25.84	27.19
House(256X256)	24.87	32.42	38.66	30.74	32.34
Peppers(256X256)	24.39	29.60	36.03	26.49	27.52
Tank(256X256)	26.30	32.76	40.35	33.67	34.86
Brain(256X256)	23.63	30.00	35.98	25.45	26.86
Cameraman(256X256)	25.33	27.86	34.70	26.04	27.17

Table-III. MSE performance of various image compression algorithm

Method/Image	HWT	JPEG	SPIHT	DWT	Proposed
Lena(256X256)	219.38	75.30	17.12	147.85	56.33
Bridge(256X256)	325.65	157.16	95.52	286.20	210.36
Clown(256X256)	243.78	64.87	20.43	191.23	136.61

Fruit(256X256)	270.46	34.78	9.03	70.17	49.74
Couple(256X256)	270.76	87.12	29.90	169.25	124.17
House(256X256)	211.76	37.19	8.85	54.33	37.86
Peppers(256X256)	236.25	31.90	16.21	145.62	115.05
Tank(256X256)	152.37	34.37	5.98	27.92	21.19
Brain(256X256)	281.71	62.30	16.38	155.20	132.96
Cameraman(256X256)	190.66	106.23	21.99	161.73	124.47

Fig.5.(A).Original

Fig.5.(B).Output of HWT

Fig.5.(C).Output of SPIHT

Fig.5.(D).Output of JPEG

Fig.5.(E).Output of DWT

Fig.5.(F).Output of proposed

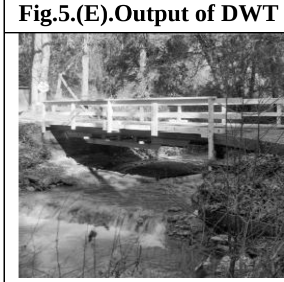


Fig.6.(A).Original

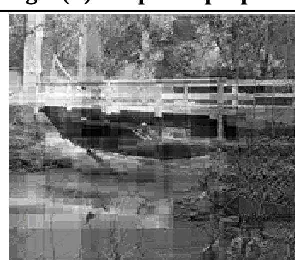


Fig.6.(B).Output of HWT

Fig.6.(C).Output of SPIHT

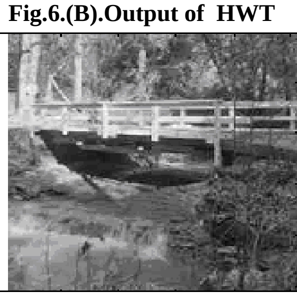
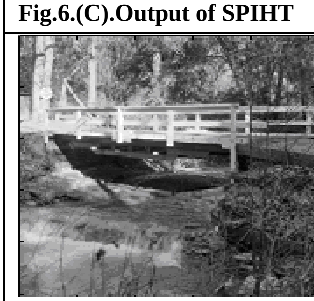


Fig.6.(D).Output of JPEG



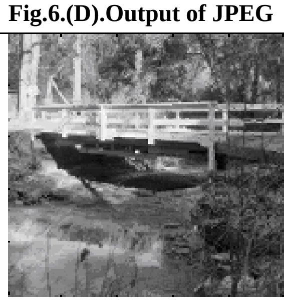


Fig.6.(E).Output of DWT

Fig.6.(F).Output of proposed

Fig.7.(A).Original

Fig.7.(B).Output of HWT

Fig.7.(C).Output of SPIHT

Fig.7.(D).Output of JPEG

Fig.7.(E).Output of DWT

Fig.7.(F).Output of proposed

Fig.8.(A).Original

Fig.8.(B).Output of HWT

Fig.8.(C).Output of SPIHT

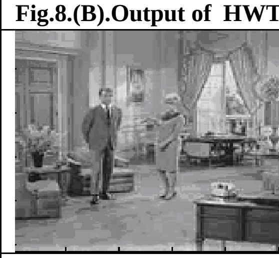


Fig.8.(D).Output of JPEG

Fig.8.(E).Output of DWT	Fig.8.(F).Output of proposed


Fig.9.(A).Original


Fig.9.(B).Output of HWT

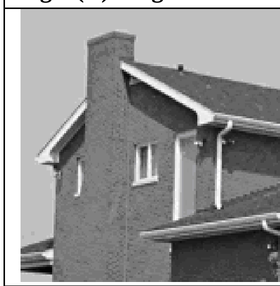


Fig.9.(C).Output of SPIHT


Fig.9.(D).Output of JPEG

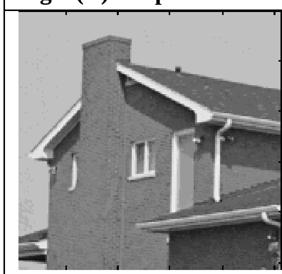


Fig.9.(E).Output of DWT

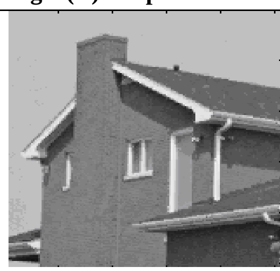


Fig.9.(F).Output of proposed

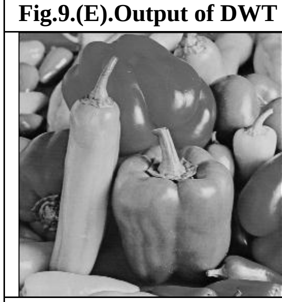


Fig.10.(A).Original

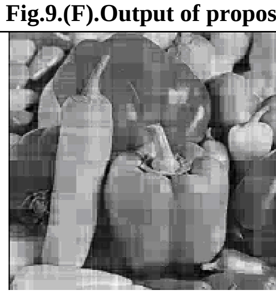


Fig.10.(B).Output of HWT

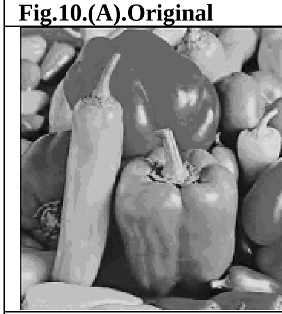


Fig.10.(C).Output of SPIHT

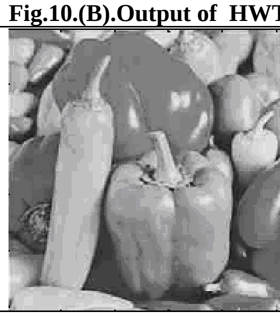


Fig.10.(D).Output of JPEG


Fig.10.(E).Output of DWT

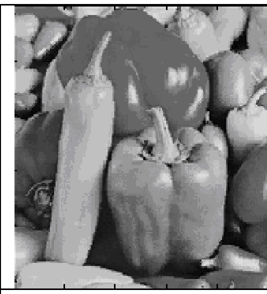


Fig.10.(F).Output of proposed

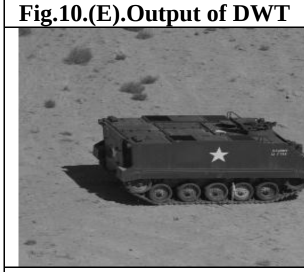


Fig.11.(A).Original

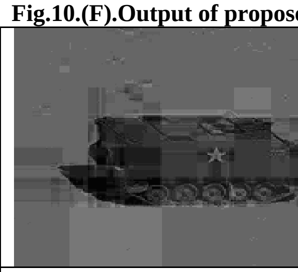


Fig.11.(B).Output of HWT


Fig.11.(C).Output of SPIHT


Fig.11.(D).Output of JPEG


Fig.11.(E).Output of DWT

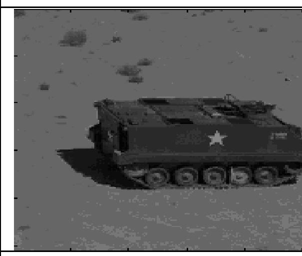


Fig.11.(F).Output of proposed

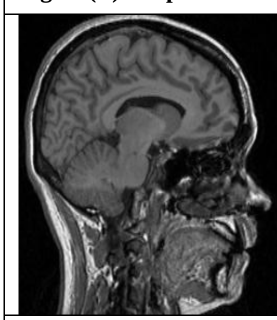


Fig.12.(A).Original

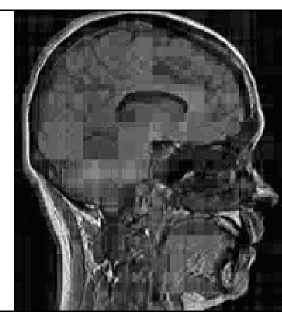


Fig.12.(B).Output of HWT

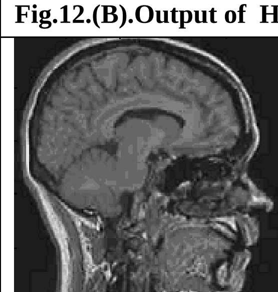


Fig.12.(C).Output of SPIHT	**Fig.12.(D).Output of JPEG**

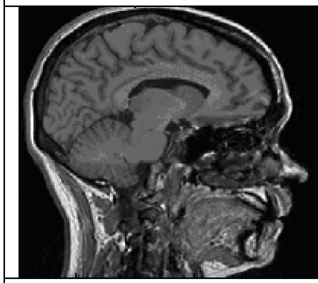
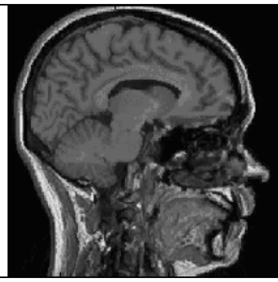

	

Fig.12.(E).Output of DWT	**Fig.12.(F).Output of proposed**
	
Fig.13.(A).Original	**Fig.13.(B).Output of HWT**
	
Fig.13.(C).Output of SPIHT	**Fig.13.(D).Output of JPEG**

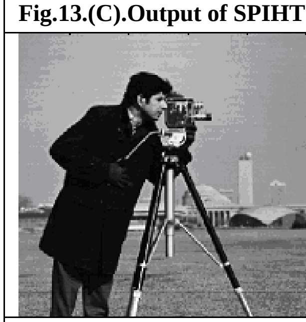



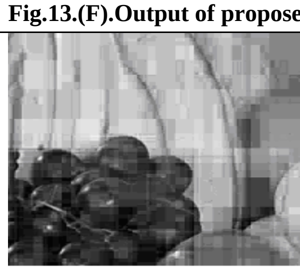
Fig.13.(E).Output of DWT	**Fig.13.(F).Output of proposed**
	
Fig.14.(A).Original	**Fig.14.(B).Output of HWT**

	
Fig.14.(C).Output of SPIHT	**Fig.14.(D).Output of JPEG**

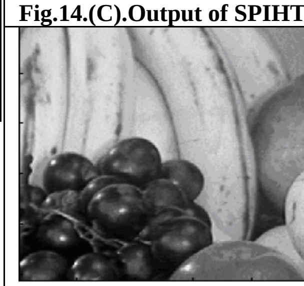
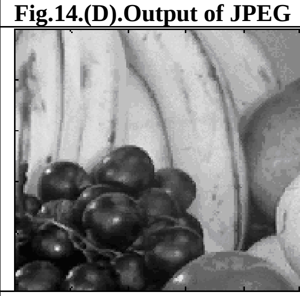

	

Fig.14.(E).Output of DWT	**Fig.14.(F).Output of proposed**

4. Conclusion

This paper proposed a novel image compression algorithm based on block truncation coding (BTC) which provide higher compression ratio as compared to existing lossy compression techniques. This algorithm was compared with existing lossy image compression techniques such as HWT, JPEG, SPIHT, and DWT. This algorithm is simple and efficient. It can be used for high quality image compression in applications such as cartoons where picture quality is very poor. But definitely it is not suitable at all in applications such as medical imaging and other related low quality application. Simulation & experimental results demonstrated that proposed algorithm outperform as compared to existing lossy image compression techniques. In future this work can be extended to video compression.

Reference

[1]. T. Acharya, A. K. Ray, "Image Processing: Principles and Applications", John Wiley & Sons, pp.351-368. 2005,

[2]. R. C. Gonzolez, R. E. Woods, S. L. Eddins, "Digital Image Processing Using Matlab", Prentice Hall, 2004.

[3]. Algorithms for Image Processing and Computer Vision, J. R. Parker, John Wiley & Sons, Inc., 1997.

[4]. T.Sreenivasulu reddy, K.Ramani, S.Varadarajan and B.C.Jinaga, Image Compression Using Transform Coding Methods, IJCSNS International Journal of Computer Science and Network Security, VOL. 7 No. 7. July,2007

[5]. Ahmed, N., Natarajan, T., and Rao, K.R. Discrete Cosine Transform, *IEEE Trans. Computers*, vol. C-23, Jan. pp. 90-93,1974,

[6]. Ken Cabeen and Peter Gent, Image Compression and the Cosine Transform, Math 45, College of the Redwoods,http://online.redwoods.cc.ca.us/instruct/darnold/laproj/fall98/pken/dct.pd

[7]. Subhasis Saha, *"Image Compression - from DCT to Wavelets* : A Review",www.acm.org/crossroads/xrds6-3/sahaimgcoding.htm
[8]. Peggy MortonHP Authorized, Image Compression Using the Haar Wavelet transform, http://online.redwoods.cc.ca.us/instruct/darnold/laproj/Fall97/PMorton/imageComp3/
[9]. G. K. Wallace, 'The JPEG Still Picture Compression Standard', Communications of the ACM, Vol. 34, Issue 4, pp.30-44,1991.
[10]. C. Cuturicu, 'A note about the JPEG decoding algorithm',http://www.opennet.ru/docs/formats/jpeg.txt, 1999.
[11]. Said, A. and Pearlman, W. A. A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees, *IEEE Trans. CSVT*, vol. 6, no. 3,pp.243-250, June 1996, http://ipl.rpi.edu/pub/EW_Code/SPIHT.ps .gz
[12]. B. E. Usevitch, 'A Tutorial on Modern Lossy Wavelet Image Compression: Foundations of JPEG 2000', IEEE Signal Processing Magazine, vol. 18, pp. 22-35, Sept. 2001.
[13]. Kotsas P, Piraino DW, Recht MP, Richmond BJ: "Comparison of adaptive wavelet-based and discrete cosine transform algorithms in image compression," *Radiology pp.*193-331, 1994.
[14]. Lawson S. and Zhu J Image Compression Using Wavelets and JPEG2000", Comm, Electronics and Commn. Engg. Journal, June, 2002.
[15]. Lewis, A. S. and Knowles, G. Image Compression Using the 2-D Wavelet Transform, *IEEE Trans. IP*, vol. 1, no. 2, pp. 244-250, April 1992.
[16]. Seungjong Kim and Jechang Jeong, Image Compression Using the Wavelet Transform and Context-Based Arithmetic Coding, Proc. SPIE: Second International Conference on Image and Graphics Vol. 4875, pp. 133-140, 2002.
[17]. Image Data Compression: Block Truncation Coding, Los Alamitos, CA: IEEE Computer Society Press, 1995.
[18]. Image Compression Using Block Truncation Coding, Delp, E.J., and Mitchell, O.R., IEEE Transaction on Communication, Vol.27, No. 9, pp.1335-1342, Sept. 1979.

Shyam Lal received B.Tech. (with Hons.) in Electronics & Communication Engineering from Bundelkhand Institute of Engineering & Technology(B.I.E.T) Jhansi (U.P.)-India and M.Tech.(with Hons.) in Electronics & Communication Engineering from National Institute of Technology, Kurukshetra (Haryana)-India in year 2001 & 2007, respectively . He is pursuing Ph.D. degree in the area of Digital Image Processing from Department of Electronics & Communication Engineering, Birla Institute of Technology, Mesra, Ranchi (Jharkhand)-India

He is Associate Professor in the department of Electronics & Communication Engineering, Moradabad Institute of Technology, Moradabad (U.P.)-India. He has published more than 33 papers in the area of Digital Image Processing and Wireless Communication & Computing at National/International Level. He is life member of Indian Society of Technical Education (ISTE), New Delhi-India (LM-39989), International Association of Engineers (IAENG), Hong Kong (M.No.-103480) and Life member International Association of Computer Science and Information Technology (IACSIT), Singapore (LM- 80333445). He has more than eight years teaching experience. His area of interest includes Digital Image Processing, Digital Signal Processing and Wireless Communication.

Mahesh Chandra received B.Sc. from Agra University, Agra (U.P.)-India in 1990 and A.M.I.E. from I.E.I., Kolkata(W.B.)-India in winter 1994. He received M.Tech. from J.N.T.U., Hyderabad-India in 2000 and Ph.D. from AMU, Aligarh (U.P.)-India in 2008. He has worked as Reader & HOD in the Department of Electronics & Communication Engineering at S.R.M.S. College of Engineering and Technology, Bareilly (U.P.)-India from Jan 2000 to June 2005. Since July 2005, he is working as Reader in the Department of Electronics & Communication Engineering, B.I.T., Mesra, Ranchi (Jharkhand)-India. He is a Life Member of ISTE, New Delhi-India and Member of IEI Kolkata (W.B.)-India. He has published more than 24 research papers in the area of Speech, Signal and Image Processing at National/International level. He is currently guiding 04 Ph.D. students in the area of Speech, Signal and Image Processing. His areas of interest are Speech, Signal and Image Processing.

Gopal Krishna Upadhyay received B.Sc. (Maths) from Kanpur University, Kanpur (U.P.)-India in 1992,, M.Sc. (Electronics) from V.B.S. Purvanchal university Jaunpur (U.P.)-India in 1994 and Ph.D.(Solid State Physics) from V.B.S. Purvanchal University Jaunpur (U.P.) in 2001.He worked as Lecturer in the Department of Physics, K.A.P.G.A. Allahabad(U.P.)-India from July 1994 to 2003 and Asstt. Professor in Department of Physics of United College of Engg., Allahabad(U.P.)-India from July 2000 to July 2003. He had also worked as Professor & Dy. Director in T.M.I.M.T, Teerthankar Mahaveer University, Moradabad (U.P.)-India. He is currently working as Director, SSITM Kasganj-Moradabad(U.P.)-India. He has more than 15 years of teaching experience. He has published more than 30 papers in the National/International level. He has also published 4 books. He is currently guiding 05 Ph.D. students. His area of interest is solid state devices, computing & image processing.