An Improved Image Compression Technique Using Huffman Coding and FFT

Rachit Patel(✉), Sapna Katiyar, and Khushboo Arora

ABES-Institute of Technology, Ghaziabad, U.P., India
rachit0508lgece@gmail.com,
arora.khushi19@gmail.com,
sapna_katiyar@yahoo.com

Abstract. Huffman coding and Fourier Transform technique are the basic technique used in image compression. Fourier transform is very powerful technique compared to Huffman coding because Fourier transform has ability to use in multiorders. The purpose of this paper is to compressed digital images using Huffman coding and Fast Fourier Transform and compare the results of both techniques. In calculation of parameters Matlab tool required. These techniques are compared with respect to various parameters such as mean square error (MSE), Peak signal to noise ratio (PSNR), Compression ratio (CR) and Bits per pixel (BPP) for the various input image of different size and it involves the new method of splitting or dividing an input image into equal rows & columns and at final stage sum of all individual compressed images.

Keywords: Image compression · Matlab · Huffman coding · Fast Fourier Transform (FFT) · PSNR · MSE · CR · BPP

1 Introduction

Compression of images is a very useful application for saving the storage data. Main motive of compression is to diminish superfluity of the image data in order to make storage or transit of data more efficient. Compression basically means diminish or removing the unwanted information or data in the image which only lead to the enhancement of memory space requirement without affecting quality of image [1]. The only way to say that, particular image compression technique is better by calculating the time taken to complete the processing i.e. compression speed which will depend upon the speed of the processor. Image compression is used in minimization of bits from images [2]. Aim of compression is to minimize redundant bits from image so that it can be store on low space or share with high efficiency. There are two type of image compression. First one lossy and second one is lossless. Lossless compression is applied for artificial images like technical drawings, icons or comics. In lossy compression low bit rate uses. Lossless compression [1] method may be used for image scans made for archival purposes. Lossy methods are useful in case of natural images like photos in applications. Fourier Transform can also use for compression. Block diagram of compression process can be shown in Fig. 1 [3]. Quantization is the process reduction in noise which makes the digital images have superior noise performance.

A. Unal et al. (Eds.): SmartCom 2016, CCIS 628, pp. 54–61, 2016.
DOI: 10.1007/978-981-10-3433-6_7

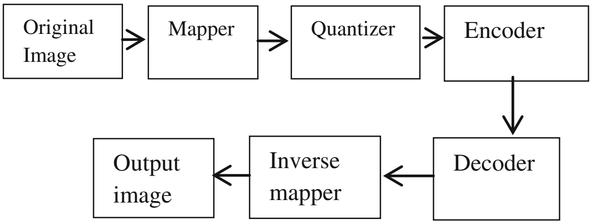

Fig. 1. Image compression process

2 Huffman Coding Algorithm

Huffman coding is a method for data compression and it is independent of data type. Image compression technique basically divided into two parts i.e. lossy technique and lossless technique [2]. Coding process begins by collecting all the probabilities for particular symbols in descending order [4]. The process starts from bottom and make a tree. At every leaf of tree, symbols are present. The process follows in steps. In each step, two probabilities are comparing and smallest one is selected. Selected probability is added to top of the tree. Selected probability deleted from the list. The process goes on until all the probabilities have been selected. In this paper the Huffman technique will be accomplished or applied after breaking the image into small parts (rows and columns) and apply this technique to each part [3].

3 Fast Fourier Transform

Fourier Transform is mathematical tool that use to convert one domain signal into other domain. Fourier Transform convert time domain signals in to frequency domain signals and Inverse Fourier Transform converts frequency domain signals to time domain signals [5]. FFT is an advance version of Discrete Fourier Transform (DFT) [5, 6]. It performs similar function in a better way within less time [7, 8]. DFT is highly recommended in frequency study because its convert a complex discrete time signal into simple discrete time frequency signal [7]. We can easily analyse the discrete nature of FFT with the help of Matlab or in real-time. With the help of FFT, N-point DFT is easily calculated with complexity of $O(N \log N)$ [9].

$$P^{(N)}Q = P_0^{\left(\frac{N}{2}\right)}(Q) + e^{-\frac{j2\pi k}{N}} P_1^{\frac{N}{2}}(Q)$$

where $P_0^{\left(\frac{N}{2}\right)}(Q)$ is $\left(\frac{N}{2}\right)$ point DFT for even and $P_1^{\frac{N}{2}}(Q)$ is point $\left(\frac{N}{2}\right)$ DFT of odd-numbered samples of $p(n)$. FFT is only applicable for the integer, means the value of N should be integer [8, 9] (Fig. 2).

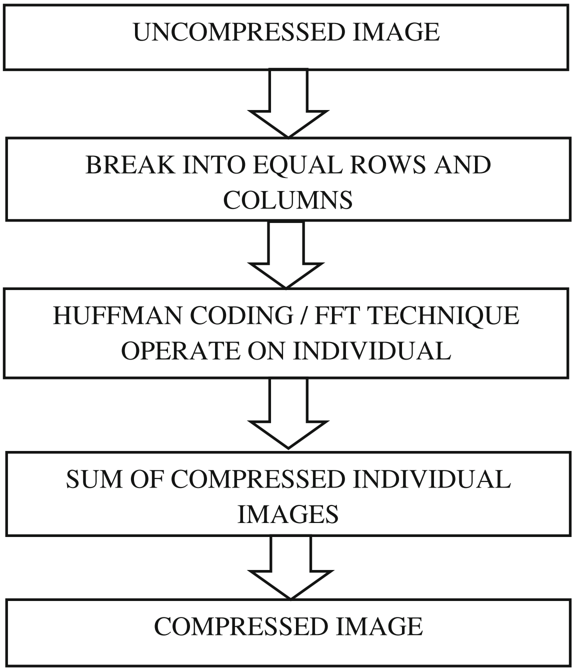

Fig. 2. Flow chart of the technique used

4 Image Quality Parameters

There are four major important parameters measure between uncompressed image and compressed image, these are following-

A. **Compression Ratio (CR)**

It is basically defined as ratio of size of uncompressed image and compressed image [1]. It is given as-

$$\text{Compression Ratio} = \frac{\text{Size of uncompressed image}}{\text{Size of compressed image}} = \frac{D1}{D2}$$

B. **Mean Square Error (MSE)**

It is defined as squared difference between an uncompressed and compressed image. It is given as-

$$MSE = \frac{1}{RC}\sum_{a=0}^{R-1}\sum_{b=0}^{C-1}[h'(a,b) - h(a,b)]^2$$

Where R is row and C is column $h'(a,b)$ the function of is compressed image and $h(a, b)$ is function of uncompressed image.

C. **Peak Signal to Noise Ratio (PSNR)**

PSNR means how much noise contains an output signal. It is mathematically expressed in logarithmic form. There exist a direct relationship between image size and PSNR.

$$\mathrm{PSNR} = 10\log_{10}\left[\frac{\mathrm{R} \times \mathrm{C}}{\mathrm{MSE}^2}\right]\mathrm{dB}$$

D. **Bits per Pixel (BPP)**

It is given as-

$$\mathrm{BPP} = \frac{\text{bits in compressed image}}{\text{pixel in the image}}$$

5 Mathematical Analysis

The complete M × M digital image can be given in the following compact matrix form where t(r, s) is the digital image, each of this matrix arrays is called an image pixel.

$$\mathrm{t(r,s)} = \begin{bmatrix} \mathrm{k}_{0,0} & \mathrm{k}_{0,1} & \cdots & \mathrm{k}_{0,\mathrm{M}-1} \\ \mathrm{k}_{1,0} & \mathrm{k}_{1,1} & \cdots & \mathrm{k}_{1,\mathrm{M}-1} \\ \vdots & \vdots & & \vdots \\ \mathrm{k}_{\mathrm{M}-1,0} & \mathrm{k}_{\mathrm{M}-1,1} & \cdots & \mathrm{k}_{\mathrm{M}-1,\mathrm{M}-1} \end{bmatrix}$$

this digital image $\mathrm{t(r,s)}$ is break into a set of non-overlapping four sub images [1], i.e. two row and two column this can be represented as –

$$\mathrm{t(r,s)} = \begin{bmatrix} \mathrm{t_1(r,s)} & \mathrm{t_2(r,s)} \\ \mathrm{t_3(r,s)} & \mathrm{t_4(r,s)} \end{bmatrix}$$

$$\mathrm{t(r,s)} = \mathrm{t_1(r,s)} + \mathrm{t_2(r,s)} + \mathrm{t_3(r,s)} + \mathrm{t_4(r,s)}$$

these t_1(r, s), t_2(r, s), t_3(r, s), t_4(r, s) are the sub-matrix of original image after applying Huffman coding/FFT technique on these sub-matrix they gives $t'_1(r,s), t'_2(r,s), t'_3(r,s)$, $t'_4(r,s)$ respectively. The compressed image can be obtain by adding these $t'_1(r,s)$, $t'_2(r,s), t'_3(r,s), t'_4(r,s)$ matrixes.

6 Results Analysis

In this paper the author calculate different image quality parameters i.e. CR, PSNR, MSE and BPP in order to make a conceptual difference between Huffman and FFT and determine which will provide better result by analysing the above listed parameters. These parameters are calculated for 256 × 256, 512 × 512 of three images and the results are shows in the tables to the corresponding image.

A. **Original Images**
See Fig. 3

Fig. 3. Original image

B. **Compressed Images**

The compressed image can be obtain by adding these $t'_1(r,s), t'_2(r,s), t'_3(r,s)$, $h'_4(x,y)$ matrixes, it can be given as

$$\mathrm{t}'(\mathrm{r},\mathrm{s}) = \mathrm{t}'_1(\mathrm{r},\mathrm{s}) + \mathrm{t}'_2(\mathrm{r},\mathrm{s}) + \mathrm{t}'_3(\mathrm{r},\mathrm{s}) + \mathrm{t}'_4(\mathrm{r},\mathrm{s})$$

By Huffman Coding
See Figs. 4 and 5

Fig. 4. Output image of 256 × 256

Fig. 5. Output image of 512 × 512

By FFT

See Figs. 6, 7, 8 and Tables 1, 2

Fig. 6. Output image of 256 × 256

Fig. 7. Output image of 512 × 512

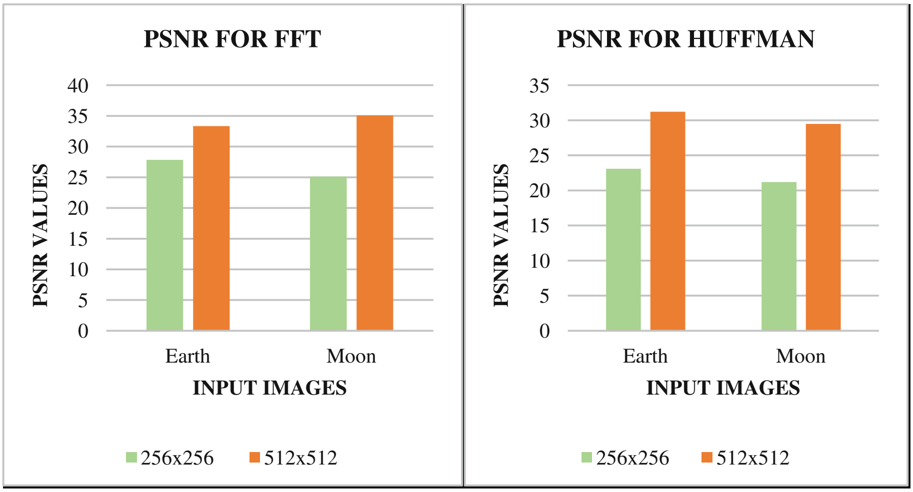

Fig. 8. PSNR for FFT and HUFFMAN

Table 1. Parameters

For earth image			For moon image	
Algorithm	Huffman		Huffman	
Resolution	256×256	512×512	256×256	512×512
CR	6.98	9.02	7.78	8.59
PSNR	23.09	31.22	21.19	29.48
MSE	18.34	14.17	20.34	17.32
BPP	2.44	1.98	2.35	2

Table 2. Parameters

For earth image			For moon image	
Algorithm	FFT		FFT	
Resolution	256×256	512×512	256×256	512×512
CR	8.75	9.53	9	9.07
PSNR	27.84	35.07	25.124	33.35
MSE	18.70	14.44	22.2052	17.45
BPP	2	1.62	2	1.75

7 Conclusion

Although the new method of splitting an image into different rows and columns will not only provide better results in image compression but also helpful for security purpose of the image transmission. Analysing the various image parameters as described in the tabular format after performing on the Matlab, it has been concluded that both compression technique will give good result, but FFT technique will provide better result than Huffman technique as seen from the image quality parameters tables. Hence it is clear that FFT technique for compression is much better than Huffman technique for image compression.

References

1. Gonzalea, R.C., Woods, R.E.: Digital Image Processing, 2nd edn. Prentice Hall, Upper Saddle River (2004)
2. Jain, A. k.: Image data compression: a review. In: Proceedings of IEEE, vol. 69, pp. 349–389 (1981)
3. Patel, R.: A fast and improved image compression technique using Huffman coding. In: IEEE WiSPNET 2016, Chennai
4. Chen, C.C.: On the selection of image compression algorithms. In: Pattern Recognition, Brisbane, Australia, August 1998
5. Sahnoun, K., Benabadji, N.: On-board satellite image compression using the Fourier transform and Huffman coding. In: WCCIT, pp. 1–5, June 2013

6. Kaushik, C.S.H., Elamaran, V.: A tutorial review on discrete Fourier transform with data compression application. In: ICGCCEE, pp. 1–6, March 2014
7. Zayed, A.I.: On the relationship between the Fourier and fractional Fourier transforms. IEEE Signal Process. Lett. **3**, 310–311 (1996)
8. Yusong, Y., Guangda, S., Chunmei, W., Qingyum, S.: Invertible integer FFT applied on lossless image compression. IEEE Robot. Intell. Syst. Signal Process. **2**, 1219–1223 (2003)
9. Van loan, C.: Transform computational framework for fast Fourier. In: SIAM (1992)