International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 15 (2018) pp. 11715-11723
© Research India Publications. http://www.ripublication.com

Performance Analysis of Lossless and Lossy Image Compression Techniques for Human Object

[1]S.Gomathi, [2]T.Santhanam

*[1]Research Scholar, Manonmaniam Sundaranar University, Tirunelveli and Assistant Professor,
Department of Computer Science, S.D.N.B Vaishnav College for women, Chennai, Tamil Nadu, India.*

*[2] Associate Professor, PG&Research Department of Computer Science and Applications,
DG Vaishnav College, Chennai, India.*

Abstract

Image compression is an important step in image transmission and storage. In order to have efficient utilization of disk space and transmission rate, images need to be compressed. The two standard compression technique are lossy and lossless image compression. Though there are many compression technique already available, a technique which is faster, memory efficient and simple surely suits the requirements of the user. In this work, the lossless method of image compression and decompression using a simple coding technique called Run-length coding is compared with Huffman coding and lossy compression technique using DCT (Discrete Cosine Transform) and FFT(Fast Fourier Transform) are compared with Haar wavelet transform. The experimental results are analysed to compare the performance of various compression techniques.

Keywords: Lossless compression, Lossy compression, Run-length coding, Huffman coding, DCT, FFT, Haar wavelet transform.

INTRODUCTION

Multimedia data, especially images have been rapidly increasing every day. Because of their large size storage and transmission have become a difficult task as they need large storage devices and high bandwidth network systems. The purpose for image compression [1] is to reduce the amount of data required for representing sampled digital images and therefore reduce the cost for storage and transmission. Image compression plays a key role in many important applications, including image database, image communications, remote sensing. There are different techniques for compressing images. They are broadly classified into two classes called lossless and lossy compression techniques [7]. As the name suggests in Lossless compression techniques, no information regarding the image is lost. In other words, the reconstructed image from the compressed image is identical to the original image. There are four types of lossless compression. They are Run-length encoding, Entropy encoding, Huffman coding and Arithmetic encoding. Whereas in lossy compression, some image information is lost, i.e. the reconstructed image from the compressed image is similar to the original image but not identical to it.There are three types of lossy compression techniques. They are Chroma sampling, Transform coding and Fractal compression. In this work use a lossless compression and decompression through a technique called Run-length coding is compared with Huffman coding (i.e. Huffman encoding and decoding). And also lossy compression technique DCT (Discrete Cosine Transform) and FFT (Fast Fourier Transform) are compared with Haar wavelet transform.

Run-length encoding (RLE) [2] is a data compression algorithm that is supported by most bitmap file formats, such as TIFF, BMP, and PCX. RLE is suited for compressing any type of data regardless of its information content, but the content of the data will affect the compression ratio achieved by RLE. Huffman coding [3] is a lossless data compression technique. Huffman coding is based on the frequency of occurrence of a data item i.e. pixel in images. The technique is to use a lower number of bits to encode the data into binary codes that occurs more frequently.

The DCT is the preferred transform used for image compression as it is typically able to incorporate more information from the image in fewer coefficients in comparison to other transforms. This is because the DCT implicitly assumes an even extension of the signal outside the domain from which it is sampled, leading to a continuous extension at the boundaries [4]. The FFT is a basic algorithm underlying much of signal processing, image processing, and data compression. Wavelets [5, 6] are lossy compression technique essentially a type of multiresolution function approximation that allow for the hierarchical de-composition of a signal or image. Any given decomposition of a signal into wavelets involves just a pair of waveforms (mother wavelets). The two shapes are translated and scaled to produce wavelets at different locations and on different scales. The wavelet coefficients preserve all the information in the original image.

Based on the application and users requirements a compression technique that would effectively address the problem of data reduction in images is needed.

REVIEW OF LITERATURE

Compression is the process of reducing storage and transmission bandwidth required to store and transfer data. M.VidyaSagar et al [8] proposed run-length encoding (RLE) is a very simple form of data compression in which runs of data (that is, sequences in which the same data value occurs in many consecutive data elements) are stored as a single data value and count, rather than as the original run. Venkatasekhar et al [9] proposed huffman coding can offer better fast fractal compression than arithmetic coding when compare to arithmetic coding, huffman coding is best for compression. It increases the speed of compression and produces high PSNR. Pratishtha Gupta et al [10] proposed run length encoding compression algorithm which is capably well-matched for RGB images data. Here considered painted and natural images for the examination of implemented scheme. In [11] the author proposed huffman coding technique is basically based on frequency of occurrence of a data item. The principle behind this technique is to use lower number of bits to encode the data that occurs more frequently. Mei Jiansheng et al [12] proposed a discrete cosine transform (DCT) is a technique for converting a signal into elementary frequency components. It expresses a sequence of finitely numerous data points in terms of a sum of cosine functions oscillating at different frequencies which is widely used in image compression. Prabhakar.

Telagarapu et al [13] proposed the Discrete Cosine Transform (DCT) attempts to de correlate the image data. After de correlation each transform coefficient can be encoded independently without losing compression efficiency. Anamitra Bardhan Roy et al [14] proposed a Fast Fourier Transform (FFT) is a faster, computational, efficient algorithm to compute the discrete Fourier Transform (DFT). Discrete Fourier Transform (DFT) is the transformation of the discrete signal taking in time domain into its discrete frequency domain representation. Sindhu M S et al [15] proposed image compression is to scale back the size and form of image without degrading the quality retrieving the back the original image. The best Haar moving ridge transform, it is the foremost simplest method among separate wavelet remodel [DWT] is mistreatment. Together with Haar moving ridge remodel, laborious thresholding has been applied to a picture.

LOSSLESS IMAGE COMPRESSION TECHNIQUE

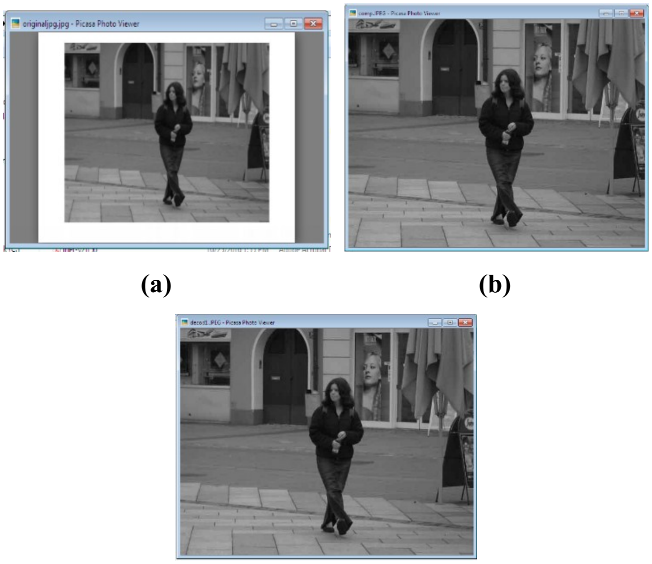
HUFFMAN ENCODING

Huffman code [16] is a lossless technique for compressing data. Huffman's greedy algorithm looks at the occurrence of each character and store it as a binary string in an optimal way. Huffman coding is a form of statistical coding which attempts to reduce the amount of bits required to represent a string of symbols. The term refers to the use of a variable-length code table for encoding a source symbol where the variable-length code table has been derived in a particular way based on the estimated probability of occurrence for each possible value of the source symbol. Huffman coding is based on frequency of occurrence of a data item. The principle is to use a lower number of bits to encode the data that occurs more frequently [17]. A Huffman code dictionary [18], which associates each data symbol with a codeword, has the property that no code-word in the dictionary is a prefix of any other codeword in the dictionary [19].Code word lengths are no longer fixed. Code word lengths vary and will be shorter for the most frequently used characters.

HUFFMAN DECODING

Huffman decoding is the reverse process of encoding, which is used to decompress the image. Before starting the compression [11] of a data file, the compressor has to determine the codes. It does that based on the probabilities of the symbols. The probabilities or frequencies have to be written, as side information, to the output, so that any huffman decoder will be able to decompress the data. This is easy, because the frequencies are integers and the probabilities can be written as scaled integers. It normally adds just a few hundred bytes to the output. It is also possible to write the variable-length codes themselves on the output. Decoding a Huffman-compressed file by sliding down the code tree for each symbol is conceptually simple, but slow. The compressed file has to be read bit by bit and the decoder has to advance a node in the code tree for each bit.

The following figure shows (a) original image, (b) compressed image and (c) reconstructed image using Huffman coding and decoding technique.



(a) **(b)**

(c)

Figure 1: (a) Original image (b) Compressed image (c) Reconstructed image

PROPOSED METHOD OF LOSSLESS IMAGE COMPRESSION TECHNIQUE

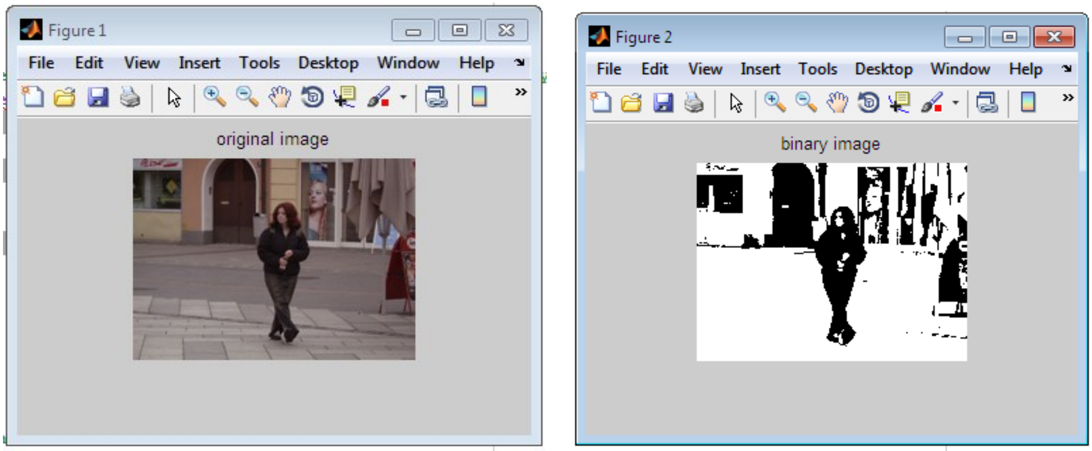
RUN LENGTH ENCODING

Run-length encoding is a data compression algorithm that is supported by most bitmap file formats, such as TIFF, BMP, and PCX. RLE schemes are simple and fast, but their compression efficiency depends on the type of image data being encoded. In this research work proposing a scheme which is compressing an image with the help of RLE. Run-length encoding (RLE) is a very simple form of lossless data compression in which runs of data (that is, sequences in which the same data value occurs in many consecutive data elements) are stored as a data value and count, rather than as original run. In Run length encoded, producing a sequence of integer run values which are then stored as an integer array. Each run value represents the length of a run of either black or white pixels and alternate run values represent alternate colors. For run-length encoding on an image, transmission of a quantity count of each of successive run of black or white scanned picture element. A black-and-white image that is mostly white, such as the page of a book, will encode very well, due to the large amount of contiguous data that is all the same color. An image with many colors that is very busy in appearance, however, such as a photograph, will not encode very well. This is because the complexity of the image is expressed as a large number of different colors [2].

RUN LENGTH DECODING

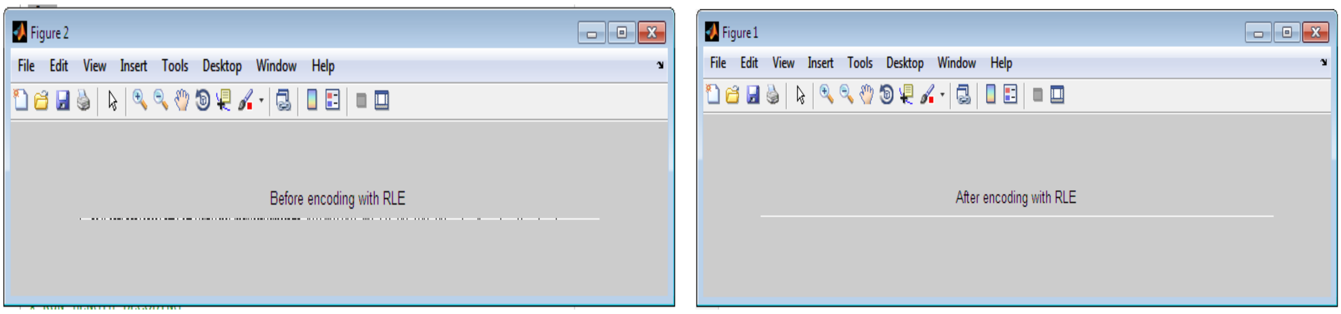
Run-length encoding converts a stream of pixels from an original image into an array of run values. Run-length decoding produces the opposite effect, restoring an original image from an array of run values. Based on the value of the first element in a compressed image array, selects the proper routine to decode and restore the original image.

The following figure shows (a) Original image, (b) Binary image, (c) Before Encoding, (d) After Encoding and (e) After Decoding image using Run-length encoding and decoding technique.



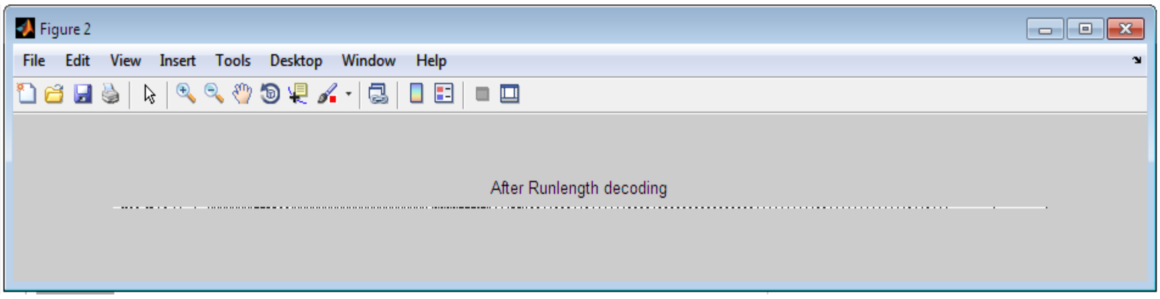
(a) Original image

(b) Binary image



(c) Before Encoding

(d) After Encoding



(e) After Decoding

Figure 2: Images of before and after RLE encoding and decoding technique

ALGORITHM

The algorithm for Run length encoding and decoding technique is given below.

Step 1: Read the image.

Step 2: Compute the size of the image.

Step 3: Convert the image to binary.

Step 4: Perform RLE on binary image.

Step 5: Perform dynamic allocation and initialization of next element of RLE.

Step 6: Output the image before and after RLE

Step 7: Apply decoding for decompression.

Step 8: Output decoding image.

LOSSY IMAGE COMPRESSION TECHNIQUE

HAAR WAVELET TRANSFORM

Wavelets [5, 6] are lossy compression technique essentially a type of multiresolution function approximation that allow for the hierarchical de-composition of a signal or image. Wavelets are functions generated from a single function by its dilations and translation. Haar transforms the simplest compression process of this kind. The Haar transform provides a multiresolution representation of an image with wavelet features at different scales capturing different levels of detail; the scale wavelets encode large regions, while the fine scale wavelets describe smaller, local regions. The wavelet coefficients preserve all the information in the original image. Haar transform decomposes each signal into two components namely the average and the difference [5, 6]. It is used to reduce the memory requirements and the amount of inefficient movement of Haar coefficients.

The following figure shows (a) The original image and the transformed image of 1d wavelet and (b) The original and vertical transformed image for 2d wavelet.


1-Dimension Haar Transform

Figure 1(a). 1-D Haar WaveletTransform

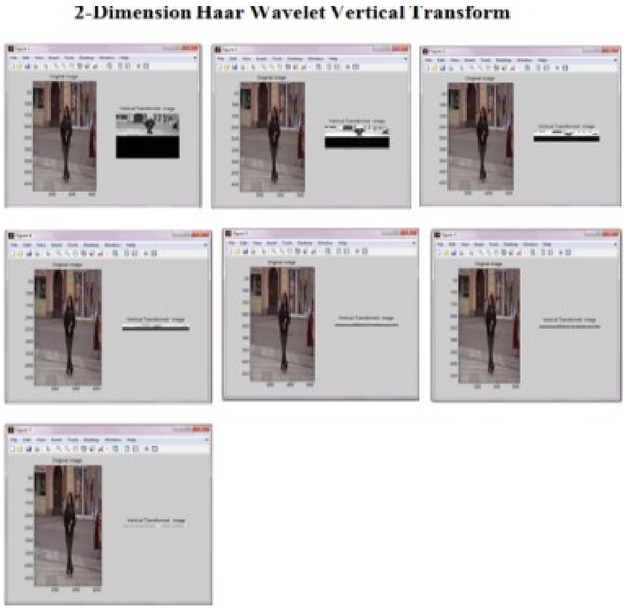

2-Dimension Haar Wavelet Vertical Transform

Figure 1(b). 2-D Haar Wavelet Vertical Transform

PROPOSED METHOD OF LOSSY IMAGE COMPRESSION TECHNIQUE

FAST FOURIER TRANSFORM (FFT)

The FFT is a basic algorithm used in much of signal processing, image processing, and data compression. A Fast Fourier Transform (FFT) is a faster, computational, efficient algorithm to compute the Discrete Fourier Transform (DFT) [20] and it's inverse.

Let x0, ..., xN-1be complex numbers. The DFT is defined by the formula

$$X_k = \sum_{n=0}^{N-1} x_n \, e \frac{-2\pi i}{N} nk \tag{1}$$

Discrete Fourier Transform (DFT) is the transformation of the discrete signal taking in time domain into its discrete frequency domain representation [21]. The functions Y = fft(x) and y = ifft(X) implement the transform and inverse transform pair given for vectors of length N by:

$$X(k) = \sum_{j=1}^{N} x(j) \omega_N^{(j-1)(k-1)}$$

$$x(j) = (1/N) \sum_{k=1}^{N} X(k) \omega_N^{-(j-1)(k-1)} \tag{2}$$

The FFT algorithm reduces the computational burden to O(nlog n) arithmetic operation. FFT requires the number of data points to be a power of 2.FFT requires evenly spaced time series.

DISCRETE COSINE TRANSFORM (DCT)

A discrete cosine transform (DCT) is a technique for converting a signal into elementary frequency components [15, 22]. It expresses a sequence of finitely numerous data points in terms of a sum of cosine functions oscillating at different frequencies which is widely used in image compression.

The DCT helps to separate the image into parts (or spectral sub-bands) of differing importance (with respect to the image's visual quality). DCT converts an image from spatial domain into a frequency domain. The DCT is closely related to the Discrete Fourier Transform (DFT) with some dissimilarity. Coefficients produced by a DCT operation on a block of pixels are similar to the frequency domain coefficients produced by a DFT operation. For processing one-dimensional signals such as speech waveforms one-dimensional DCT is used.

For analysis of two dimensional (2D) signals such as images, a 2D version of the DCT is required. As the 2-Dimensional DCT can be computed by applying 1D transforms separately to the rows and columns, it can be said that the 2D DCT is separable in the two dimensions.

For an M × N digital image f (x, y), its two-dimensional discrete cosine transform and its inverse transformation is defined by the following equations [15].

$$C(u,v)=\alpha(u)\,\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{\Pi(2x+1)u}{2N}\right]\cos\left[\frac{\Pi(2y+1)v}{2N}\right] \tag{3}$$

$$f(x,y)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\alpha(u)\,\alpha(v)\,c(u,v)\cos\left[\frac{\Pi(2x+1)u}{2N}\right]\cos\left[\frac{\Pi(2y+1)v}{2N}\right] \tag{4}$$

There, C (u, v) is the result of discrete transform, and also known as DCT coefficient.

Where, u, v = 0, 1, 2 ... N-1

x, y = 0, 1, 2, ... N-1

α (u) is defined as follows**:**

$$\begin{aligned} \alpha(u) &= \sqrt{\frac{1}{N}} && u=0; \\ \alpha(u) &= \sqrt{\frac{2}{N}} && u=1,2\ldots\ldots N-1 \end{aligned} \tag{5}$$

The order of values obtained by applying the DCT is coincidentally from lowest to highest frequency. When reconstructing the data and transforming it back to the spatial domain, the results are remarkably similar to the original signal. The DCT method can be used to compress both color and gray scale images.

The following figure show the compression images using DCT and FFT.

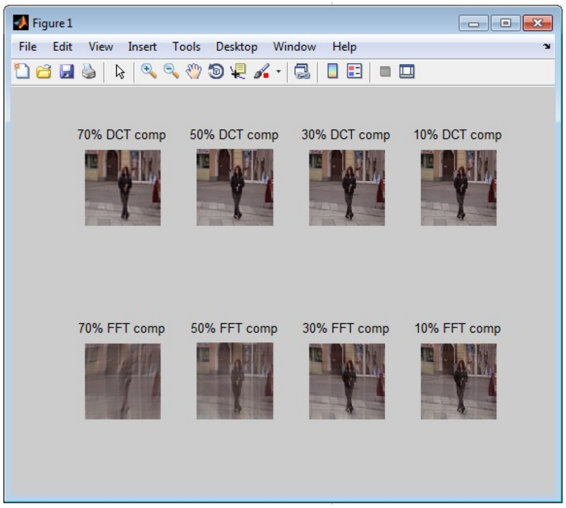


Figure 4: Compression images using DCT & FFT techniques.

ALGORITHM

The algorithm for DCT & FFT techniques is given below.

Step 1: Read the image.

Step 2: Compute the size of the image.

Step 3: Apply a function to computes the two-dimensional DCT of an image.

Step 4: Perform the two-dimensional inverse DCT of an image.

Step 5: Output the compression image after DCT.

Step 6: Apply a function computes the two-dimensional FFT of an image.

Step 7: Perform the two-dimensional inverse FFT of an image.

Step 8: Output the compression image after FFT.

PERFORMANCE METRICS

To compare the uncompressed with compressed image, the following performance measures are used.

Compression ratio

One success measurement in image data compression is the Compression Ratio. The Compression ratio [23] is to measure the ability of data compression by comparing the size of the image being compared with the original size. The Greater the compression ratio means better performance.

The Compression ratio is defined as the ratio of an original image and compressed image [23, 24].

Compression Ratio = Original Image / Compressed Image.

Elapsed Time

Quality analysis has been presented by measuring overall elapsed time [5] for the wavelet transform and for the Huffman coding with help of start watch timer and stopwatch timer functions. They are used to determine how long it takes for a program to run.

Size

It specifies the memory size [5] of objects before and after the compression.

Quality Evaluation

There are two error metrics which is used to compare the quality of image compression, that are known as **MSE** and **PSNR** [11, 23].

Quality is evaluated by using **PSNR** (peak signal-to-noise ratio) value. PSNR is measured between two images. Here, **PSNR** is measured between original image and decompressed image.

PSNR formula is as:

$$PSNR = 10log10 \frac{(2n-1)^2}{\mathrm{MSE}} \tag{6}$$

MSE: MSE is the Mean square error among original and decompressed image.

MSE formula is as:

$$\mathrm{MSE}= \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=1}^{n-1} [I(i,j) - K(i,j)]^2 \tag{7}$$

Where m, n is the size of the original image.

For a good quality image after decompression, following conditions must hold:

- PSNR should be high
- MSE should be low
- Compression ratio should be high

RESULTS

In this research work, the proposed compression technique is tested on INRIA PERSON data set. The experimental results of the various compression technique implemented in MATLAB. Lossy Haar wavelet, DCT, FFT and Lossless Huffman and Run-length coding for compression ratio and elapsed time are depicted in Table 1. It is evident from the table that, the compression ratio of the DCT & FFT for different types of images are higher than the Haar wavelet transform in lossy compression and compression ratio of Huffman coding is less than Run-length coding in Lossless compression. On the other hand, the elapsed time of Haar wavelet transform is less than DCT&FFT in lossy compression and the elapsed time of Huffman coding is more than Run-length coding in Lossless compression.

Table 1: Compression Ratio and Elapsed time of Lossy and Lossless compression

Test Images	Compression Ratio in Bytes					Elapsed Time(sec)				
	Lossy Compression			Lossless Compression		Lossy Compression			Lossless Compression	
	Haar Wave-let	DCT	FFT	Huffman Coding	Run-length Coding	Haar Wave-let	DCT	FFT	Huffman Coding	Run-length Coding
BMP	3.91	18.36	18.36	2.99	8.41	0.0381	0.3701	0.3552	0.5741	0.3635
JPEG	5.00	7.72	8.88	1.36	7.61	0.0464	0.3837	0.3636	0.2950	0.3704
TIFF	7.86	3.22	3.22	1.18	15.87	0.0392	0.3725	0.3708	0.3992	0.3855
PNG	5.14	25.73	29.10	3.27	39.43	0.0381	0.3620	0.3461	4.0051	0.3811

Table 2: Memory size of Lossy and Lossless compression

Test Image	Memory size(KB)										
	Before Compr-ession	Lossy Compression						Lossless Compression			
		Haar Wavelet		DCT		FFT		Huffman Coding		Run-length Coding	
		After	Percent-age (%)	After	Percent-age (%)	After	Percent-age (%)	After	Percent-age (%)	After	Percent-age (%)
BMP	900	230	25.55	49	5.44	49	5.44	301	33.44	107	11.88
JPEG	27.1	5.41	19.96	3.51	12.95	3.05	11.25	19.8	73.06	3.56	13.13
TIFF	103	13.1	12.71	31.9	30.97	31.9	30.97	87	84.46	6.49	6.30
PNG	422	97	22.98	16.4	3.88	14.5	3.43	129	30.56	1.07	0.25

Memory size before and after the lossy and lossless compression is shown in Table 2.

From this table DCT&FFT result in memory size compared to Haar wavelet in Lossy compression and the resultant memory size using Run-length coding is less compared to Huffman coding in lossless compression.

Table 3 and Table 4 describes about the Haar Wavelet and DCT & FFT in lossy compression and Huffman coding and Run-length coding in lossless compression output with different performance evaluation parameters such as mean square error, peak signal to noise ratio (db), These different parameters are used to compare the output or quality of the different types of the image. Different type of images will be taken to compare the quality e.g. .bmp, .jpg, .tif and .png etc.

The MSE represents the Means Squared Error between the compressed and the original image whereas PSNR represents a measure of the peak error. Lower the value of MSE and higher the value of PSNR indicates a low error and good image quality respectively.

Table 3: Mean Square error of Lossy and Lossless compression Techniques.

Test Images	MSE				
	Lossy Compression			Lossless Compression	
	Haar Wavelet	DCT	FFT	Huffman Coding	Run-length Coding
BMP	0.0025	0.0013	9.5530e-004	2.9634e-004	0.0017
JPEG	4.1905e-004	6.7064e-004	6.6433e-004	2.3601e-004	0.0018
TIFF	2.2855e-004	7.1997e-004	7.2146e-004	2.9700e-004	0.0020
PNG	4.2202e-004	7.2013e-004	7.2151e-004	2.9634e-004	0.0020

Table 4: Peak Signal to Noise Ratio of Lossy and Lossless compression Techniques.

Test Images	PSNR				
	Lossy Compression			Lossless Compression	
	Haar Wavelet	DCT	FFT	Huffman Coding	Run-length Coding
BMP	26.0975	28.8941	30.1986	35.2821	27.7276
JPEG	33.7773	31.7351	31.7762	36.2708	27.3432
TIFF	36.4102	31.4269	31.4179	35.2724	27.0796
PNG	33.7467	31.4259	31.4176	35.2821	27.0796

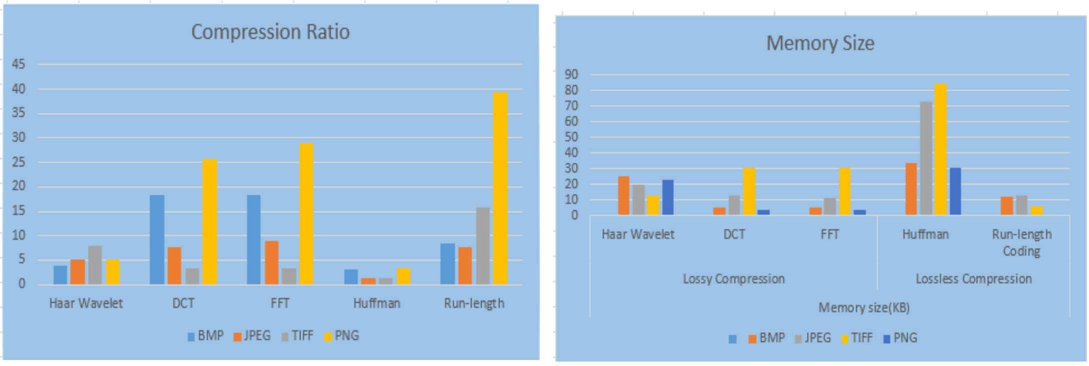


Figure 5: (a) Compression Ratio of Lossy and Lossless compression techniques

(b) Memory size of Lossy and Lossless compression techniques

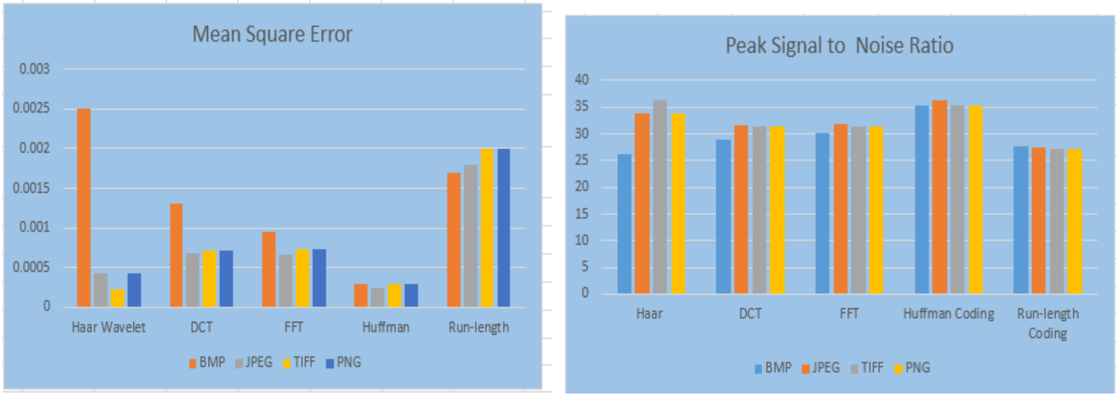


Figure 6: (a) MSE of Lossy and Lossless compression techniques

(b) PSNR of Lossy and Lossless compression techniques

The graphical representation of compression ratio and memory size of both lossy compression and Lossless compression techniques is shown in Figure 5.And also Figure 6 shows the graphical representation of MSE and PSNR value of lossy and lossless compression techniques.

CONCLUSION

In this article, a comparative analysis shows a comparison between the various types of images using a lossless Huffman coding and Run-length coding technique and lossy Haar wavelet technique, Discrete Cosine Transform (DCT) and Fast Fourier Transform (FFT) for human object. Some important performance measures as compression ratio, elapsed time, memory size before and after the compression, MSE and PSNR have been used. After observing the results it can be concluded that the objective of the research is achieved with following observation. It is concluded that Image compression is an important technique in digital image processing. There are different types of compression techniques but DCT& FFT in lossy compression and Run-length coding in lossless compression have high compression ratio and results in less memory size.

Run-length coding has been used in the proposed algorithm and better quality of compressed image with less MSE and Huffman coding contains high PSNR value in lossless compression when compared to lossy compression techniques. In future other compression techniques can be explored for images of human objects.

REFERENCES

[1] Jagadish H. Pujar,Lohit M. Kadlaskar,2010,” A New Lossless method of Image Compression and Decompression using Huffman coding techniques”,Journal of Theoretical and Applied Information Technology, pp. 18-22.

[2] Amin Mubarak Alamin Ibrahim,Mustafa Elgili Mustafa,2015,”Comparison Between (RLE And Huffman) Algorithms for Lossless Data Compression”, (IJITR) International Journal of Innovative Technology and Research, Vol. 3(1), pp.1808 – 1812.

[3] Mamta Sharma,2010 **“**Compression Using Huffman Coding”, IJCSNS International Journal of Computer Science and Network Security, Vol.10(5), pp.133-141.

[4] A.K. Jain, 1989,Fundamentals of Digital Image Processing, Prentice Hall..

[5] T.Santhanam, S.Gomathi,” Representing a image using a Haar-wavelet Transformation for Human Parts Detection”,2014, International Conference on Information Communication and Embedded Systems (ICICES).

[6] Navjot Kaur, Pree.ti Singh,2012 “A New Method of Image Compression Using Improved SPIHT and MFHWT,” International Journal of Latest Research in Science and Technology,Vol.1(2), pp.124-126.

[7] S.Vijayarani,, Ms. Nivetha,2016,” Image Compression Techniques – An Overview”, International Journal of Modern Computer Science (IJMCS), Vol.4(5),pp.96-101.

[8] M.VidyaSagar, J.S. Rose Victor,2013,” Modified Run Length Encoding Scheme for High Data Compression Rate”, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol. 2(12),pp.3238-3242.

[9] D. Venkatasekhar, P. Aruna,2012,” A fast fractal image compression using Huffman coding,” Asian Journal of Computer Science and Information Technology”, Vol.2(9), pp 272– 275.

[10] Pratishtha Gupta , Varsha Bansal,2015,” The Run Length Encoding for RGB Images”, International Journal of Converging Technologies and Management (IJCTM), Vol.1(1),pp.1-4.

[11] Rahul Shukla, Narender Kumar Gupta,2015,” Image Compression through DCT and Huffman Coding

Technique",International Journal of Current Engineering and Technology,Vol.5(3), pp.1942-1946.

[12] Mei Jiansheng, Li Sukang. Tan Xiaomei,2009," A Digital Watermarking Algorithm Based On DCT and DWT", Proceedings of the 2009 International Symposium on Web Information Systems and Applications (WISA'09) Nanchang, P. R. China, pp. 104-107.

[13] Prabhakar.Telagarapu, V.Jagan Naveen, A.Lakshmi..Prasanthi, G.Vijaya Santhi,2011, " Image Compression Using DCT and Wavelet Transformations",International Journal of Signal Processing, Image Processing and Pattern Recognition,Vol. 4(3),pp.61-74.

[14] Anamitra Bardhan Roy,Debasmita Dey,DevmalyaBanerjee,BidishaMohanty,2012,"Comparison of FFT, DCT, DWT, WHT Compression Techniques on Electrocardiogram & Photoplethysmography Signals", International Conference on Computing, Communication and Sensor Network (CCSN),pp.6-11.

[15] Sindhu M S, Dr. Bharathi.S.H,2016," Image Compression using Haar Wavelet Transform and Huffman Coding", Vol.3(5), pp.182-186.

[16] A. A. Shaikh, P. P. Gadekar,2015," Huffman Coding Technique for Image compression", Internaional Journal of Advanced Computer Technology,Vol.4(4),pp.1585-1587.

[17] Sharma .M,2010,"Compression Using Huffman Coding",International Journal of Computer Science and Network Security, Vol.10(5).

[18] Li, Z., and Drew, M. S.2004, "Fundamental of Multimedia", School of Computing Science Fraser University,Pearson Education International, Second Edition.

[19] Mitzenmacher, M.2004, "On the Hardness of Finding Optimal Multiple Preset Dictionaries", IEEE Transaction on Information Theory, Vol.50(7), pp.1536-1539.

[20] Frigo, M. and S. G. Johnson,1998, "FFTW: An Adaptive Software Architecture for the FFT",Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, Vol. 3, pp. 1381-1384.

[21] Nilanjan Dey, Tanmay Bhattacharya, S.R. Bhadra Chowdhury, 2012,"A Session based Multiple Image Hiding Technique using DWT and DCT", International Journal of Computer Applications Vol. 38(5).

[22] Mei Jiansheng, Li Sukang. Tan Xiaomei,2009," A Digital Watermarking Algorithm Based On DCT and DWT", Proceedings of the 2009 International Symposium on Web Information Systems and Applications (WISA'09) Nanchang, pp. 104-107.

[23] Gaurav Kumar, Er. Sukhreet Singh Brar, Rajeev Kumar, Ashok Kumar,2015," A Review: DWT-DCT Technique and Arithmetic-Huffman Coding based Image Compression", IJEM, Vol.5(3), pp.20-33.

[24] Navjot Kaur, Preeti Singh,2012, " Enhancement of Compression Ratio and Image Quality using ISPIHT with MFHWT", International Journal of Computer Applications, Vol.54(8), pp.20-23.