Huffman Code Based Improved Analysis of Lossy & Lossless Image Compression

BY

Shapna Rani Sutradhar

ID No: 181-31-255

A Thesis Submitted in partial fulfillment of the requirements for the degree of Master of Science in Electronics and Telecommunication Engineering

Supervised By

Prof. Dr. A. K. M. Fazlul Haque
Professor & Associate Dean
Department of ETE
Faculty of Engineering Daffodil International University

Daffodil International University
Dhaka, Bangladesh
DECEMBER 2019

APPROVAL

This project titled **"Huffman Code Based Improved Analysis of Lossy & Lossless Image Compression"** submitted by Shapna Rani Sutradhar, to the Department of Electronics and Telecommunication Engineering (ETE), Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of M.Sc. in Electronics and Telecommunication Engineering and approved as to its style and contents. The presentation was held on **December, 2019**.

BOARD OF EXAMINERS

Prof. Dr. A.K.M. Fazlul Haque
Professor & Associate Dean
Department of ETE
Daffodil International University

Chairman

Dr. Engr. M. Quamruzzaman
Professor
Department of ETE
Daffodil International University

Internal Examiner

Md. Zahirul Islam
Assistant Professor
Department of ETE
Daffodil International University

Internal Examiner

Dr. Saeed Mahmud Ullah
Professor
Department of EEE
University of Dhaka

External Examiner

DECLARATION

I hereby declare that this project is my own work and effort under the supervision of **Professor Dr. A.K.M. Fazlul Haque, Professor and Associate Dean, Department of Electronics and Telecommunication Engineering,** Daffodil International University, Dhaka. It has not been submitted anywhere for any award. Where other sources of information have been used, they have been acknowledged.

Supervised By:

Prof. Dr. A.K.M. Fazlul Haque
Professor and Associate Dean
Department of ETE
Faculty of Engineering
Daffodil International University

Submitted By:

Shapna Rani Sutradhar
ID: 181-31-255
Department of ETE
Daffodil International University

ACKNOWLEDGMENTS

First of all, I would like to convey my gratitude to the Almighty, for giving me the right direction while attempting the task.

The real sprit of achieving a goal is through the way of excellence and austere discipline. I would have never succeeded in completing my task without the cooperation, encouragement and help provided to me by various personalities.

This work would not have been possible without the support and guidance of Professor **Dr. A.K.M. Fazlul Haque, Professor and Associate Dean,** Department of Electronics and Telecommunication Engineering, Daffodil International University, Dhaka, under whose supervision I chose this topic and developed the project. Deep Knowledge and keen interest of my supervisor in the field of digital image processing influenced me to carry out this project.

I would like to express my heartiest gratitude to the other faculty members, the staffs of the ETE Department of Daffodil International University.

I must acknowledge with due respect the constant support and patience of my family members for completing this project.

Shapna Rani Sutradhar

ABSTRACT

Digital Image processing has discovered an unbelievable growth and created an important technological impact in different areas. The most significant application of digital image processing is image compression. It is the representation of data compression on digital images. The main objective of this paper is to analyze different parameters like, CR, MSE, PSNR, BPP, Retained signal energy etc. for lossless and lossy compression. One medical image is also being analyzed for different resolutions of the images and compared the results. An improved version of Huffman coding algorithm which is basically useful for lossless compression, is being used for both lossless and lossy compression in this paper. Since, Huffman coding algorithm is relatively easier for its simpler mathematical calculation due to determine the several parameters, that's why this algorithm is used for the compression. Histograms of the selected original and reconstructed images have been examined also.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

CHAPTER 1

INTRODUCTION

1.1 General Introduction:

Digital Image processing has revealed an incredible growth and created an important technological impact in different areas. The most vital application of digital image processing is image compression. A noble image will be examined in accordance to higher image quality parameters with memory space prerequisite that is very essential in image investigation [1]. Compression sates to reduce the amount of data that is used to characterize an image, a file or video content without extremely decreasing the excellence of the original information. Image compression is the representation of data compression on digital image [2]. Image compression is considered to be the art of effective coding of image data, whose goal is to reject the quantity of bits necessary to express an image. This is the result of significant storage in the memory compulsory for image storage or in the channel capacity necessary for image transmission [3]. Huffman coding algorithm is comparatively easier and very simple to implement. That's why, this coding is very useful in image compression. In this research, both lossless as well as lossy compression and decompression have been implemented using an improved version of Huffman coding algorithm. One medical image is also being analyzed. Performance of different parameters have been examined for both compressions for different image size like, 256×256 and 512×512. The histograms of the compressed and reconstructed images are also analyzed in this research. All the analysis and simulations have been analyzed and verified using MATLAB software and MATLAB wavelet toolbox.

1.2 Significance of this Research:

Data compression has become the prerequisite for many of the applications in diverse areas, for example, Information technology, communications, computer science, medicine etc. The storage of few images could pose a problem. That's why image compression is required. A lossless and lossy image compression methods were analyzed in this research in case of different parameters and an improved Huffman coding algorithm is used which is applicable for both compressions.

1.3 Aims and Objectives:

This research aims at analyzing and comparing the performance of various quality parameters, such as MSE, PSNR, CR, BPP, retained signal energy etc. of lossless and lossy compression and also the histograms of the images. Images of different resolutions have been used for this. To achieve this goal, an improved Huffman coding, an easier and simpler compression technique has been used for both compressions.

1.4 Methodology:

In this research, a lossless and lossy image compression has been studied using different parameters and also analyzed the histograms of the compressed and reconstructed images. MATLAB software and MATLAB wavelet toolbox is used for the overall analysis.

CHAPTER 2

IMAGE COMPRESSION

2.1 Preface:

At present, images are essential documents; images contain several information. In some applications such as data description and broadcasting system, compression of image is necessary. It is necessary to compress more or less according to the purpose of the application to work with them in a few applications. Many works have been done on image compression. In the paper [4], Rachit Patel, Virendra Kumar, Vaibhav Tyagi, Vishal Asthana proposed that compression of images using the coding technique of Huffman is simpler and easier to compress. The aim of this paper is to examine the Huffman coding technique that is used by analyzing different characteristics to eliminate the redundant bits in the information. In the paper [5], Nehal Markandeya, Prof. Dr. Sonali Patil analyzed image compression using block truncation coding where three algorithms were selected and comparison was performed among them. In the paper [6], Vikash Kumar, Sanjay Sharma worked on lossless image compression through Huffman coding and analyzed its application on image processing. Harpreet Kaur and Neelofar Sohi worked on applications of Histogram in Image Enhancement [7]. In the paper [8], Sanjay Kumar Gupta worked on an algorithm for image compression using Huffman coding techniques. Jyoti A. Patil et all. Worked on lossless image compression using Huffman coding algorithm [9].

2.2 Definition of Image:

An image, especially a digital image is called a distinct space which is the collection of minor surface elements entitled pixel. Intensity level of a set of value coding is enclosed in each of this elements at individual place. A digital image may be managed through a huge amount of dissimilar devices like, an MRI machine, a camera or different kind of element along with a sensor capable of capturing light intensity.

2.2.1 Gray Scale Image:

The digital image where every pixel only comprises one scalar value that is its intensity is called a gray scale image. The sum of possible amount of intensity values is focused on the mathematical form encoding the image.

For instance, an image is encoded with n=8 bits with probable intensity values of $L=2^8=256$ moving from 0 signifying black to L-1=255 signifying white.

2.2.2 Color Image:

Another type of image is a color image which is a virtual pixel display with color data. Depending on the three color stations encoded, each image could be decomposed into three layers in a color image which are: Red, Green and Blue.

For example, an 8-bits color image encodes the 3-bit red and green channel, and the 2-bit blue encodes 256 different colors.

2.3 Definition of Image Compression:

Compression implies decreasing the amount of information which is used to denote a folder, picture or video content without extremely decreasing the excellence of the original data. Accordingly, image compression denotes the reduction of the image file without decreasing the excellency the original image. In image compression, the dimension of the original image is reduced so that it requires less memory to store the data.

2.4 Why Image Compression:

An image without compression requires huge quantity of space in storage media, and additional time is needed to migrate from one device to another. Thus if we want to transfer or stock digital image then it is important to first compress it for rapid transmission speed and to restore in a less storage. For modern multimedia applications, compression is therefore very necessary [10].

2.5 Image Compression Principal:

Many images have the most important characteristics of interconnecting the adjacent pixels and thus covering redundant data. Then the principal responsibility is to discover fewer correlated image illustration. Two essential compression mechanisms are irrelevancy reduction and redundancy.

a. The purpose of redundancy reduction is to eliminate the signal's repetition forms.

b. Irrelevancy reduction ignores portions of the signal which are not be observed through the signal processor, specifically the Human visual system. Three forms of redundancies due to compress file size in an image are there. They are:

a) **Coding redundancy**: In coding redundancy, every pixel of the uncompressed image is coded by a stable length. Compression may be produced by using the variable length code patterns such as Huffman coding and arithmetic coding. It has smaller bits to signify repeatedly occurring symbols.

b) **Interpixel redundancy**: This redundancy relies on the nearby pixels that have nearly same value.

c) **Psyco visual redundancy**: This kind of redundancy is related to various sensitivities to all images signals and human visual system cannot instantly differentiate all colours.

2.6 Image Compression Techniques:

The following diagram shows the categories of image compression:

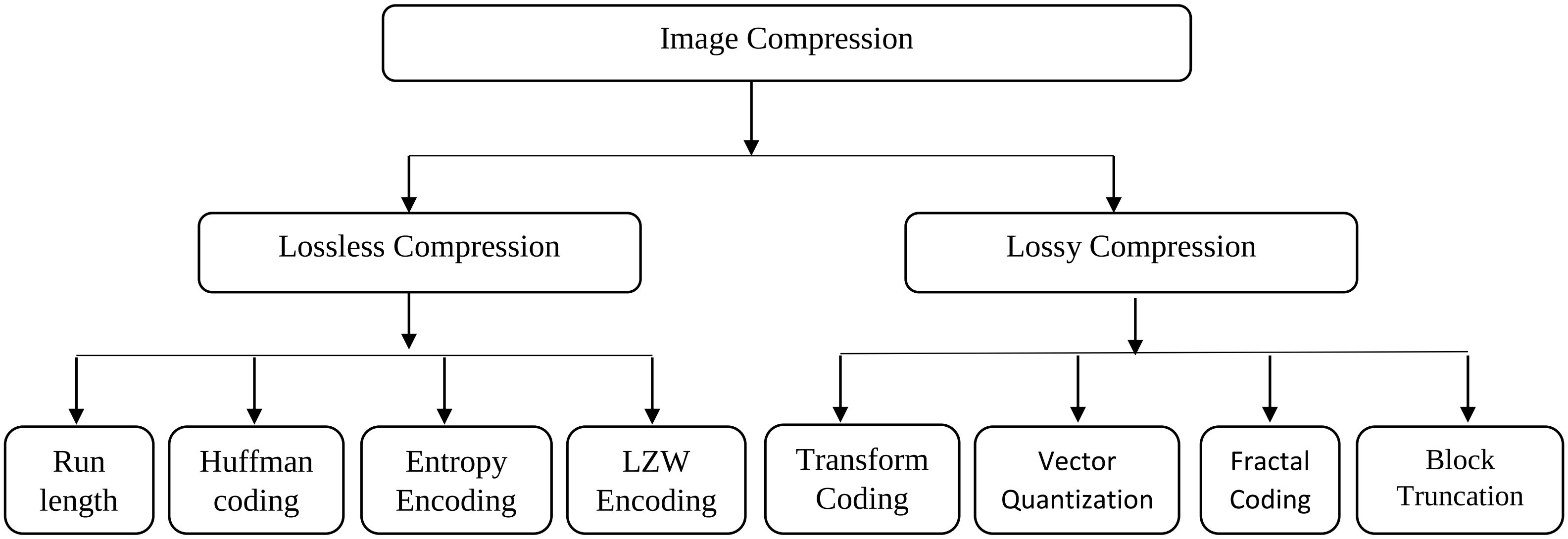


Figure 2.1: Classification of Image Compression

Figure 2.1 shows the details classification of image compression. Image compression techniques can be characterized into two classifications.

2.6.1 Lossless Compression:

Lossless image compression is a compression technique where image is compressed without losing any data. This compression ratio provides better quality of compressed images.

There are some methods of lossless compression:

a) Run Length Encoding:

One of the easiest methods of data compression is the run-length encoding system. In conjunction with repetitive data, this compression approach is advantageous. It contains shifting a structure (run) of equal symbols by a pair comprising the character and run length [11].

b) Huffman Coding:

Another encoding is Huffman encoding that uses the probability distribution of the symbols of the source to improve the code words for symbols. It is established on the arithmetical amount of frequencies or probabilities. This encoding can decrease the file size by 10% to 50% by eliminating the irrelevant information.

c) Entropy Encoding:

This encoding is a lossless data compression system which is not dependent on the definite features of the medium. One of the foremost sorts of entropy coding generates as well as allocates a distinctive prefix-free code to every distinctive symbol that happens in the input. After that these entropy encoders can compress data by substituting every fixed-length input symbol with the equivalent variable-length prefix-free output code word. The length of every code word is about relational to the negative logarithm of the probability. Consequently, the utmost common symbols use the shortest codes.

d) Lempel–Ziv–Welch Coding:

This is a common lossless algorithm that was generated by Abraham Lempel, Jacob Ziv, and Terry Welch for data compression. Welch circulated this coding in 1984 as an upgrade application of the LZ78 algorithm published in 1978 by Lempel and Ziv. LZW is a coding based on the dictionary.
It can be static or dynamic in this encoding. Dictionary is stable in the encoding and decoding processes in static dictionary coding. On the other hand, dictionaries are simplified on fly in dynamic dictionary coding. This algorithm is very relaxed to use, as it has the possibility in hardware implementations on behalf of very high throughput. This algorithm has been extensively used to

compress UNIX file compression tool and is very useful in the format of the GIF image. This compression was the first broadly used compression of images on computers worldwide. A big English text file can naturally be compressed to around half its original size through LZW [12].

e) Area Coding:

This coding is an improved length coding scheme that can reflect the image's two-dimensional character. This technique is one of the most substantial lossless compression techniques. For an image coding, understanding it as a consecutive stream does not create any sense, as it is, in fact, an array of systems, a two-dimensional entity construction. This algorithm for area coding has the effort to discover quadrilateral sections along with the same properties. These sections are coded in an expressive method of a component with two points as well as a particular arrangement. This kind of coding is able to be extremely effective, however it accepts the problem of a nonlinear process, that cannot be executed in hardware. The compression time act is therefore not competitive, although the compression ratio is [13].

2.6.2 Lossy Compression:

This is a compression technique where the compressed image is reconstructed with loss of data. Lossy compression usage source encoding methods that might include transform encoding, differential encoding or vector quantization [14].

The lossy compression methods are:

a) Transform Coding:

For specific data such as audio signals or photographic images, transform coding is one type of data compression. This conversion is usually lossy, which resulting in a minor quality copy of the original input. Reduced bandwidth is mandatory in this form of coding. Different transform like, DFT (discrete Fourier transform) and DCT (discrete coding transform) are very useful to convert the pixel of the original image into frequency domain coefficients [15].

b) Vector Quantization:

Vector quantization (VQ) method is the addition of Scalar quantization in various dimensions. This method progresses a dictionary of fixed-size vectors that are called code vectors. A specified image again subdivided into non-overlapping blocks that is called image vectors. After that for every image vector, the neighboring similar vector in the dictionary is identified and its index in the dictionary is used as the encoding of the original image vector.

c) Block Truncation Coding:

Block truncation coding is a very familiar method for image compression, it (BTC) classifies the original image into small sub blocks of size n x n pixels and after the division of image, it decreases the number of gray levels within each block. Reduction of gray level is accomplished by a quantizer. Threshold and restoration values are considered for each block and a bitmap of the block is attained for that values. We exchange the entire pixels whose values are larger than or equal (less than) to the threshold by a 1 (0), in this bit map. After that for every section (group of 1s and 0s) of the bitmap, regenerated value is determined.

d) Sub band coding:

This other coding type divided a signal's frequency bands. Every sub band is coded through encoder afterwards. On the other hand, the decoder decodes the sub band signal and it is experimented and delivered over the synthesis filter. In speech coding and image coding, this coding technique is usually very useful.

e) Fractal Coding:

By using standard image processing systems such as edge detection, texture analysis, color separation and spectrum, the image is decomposed into different divisions in this coding technique. After that segment is observed in a fractal library. In fact, the library includes codes called iterated function system (IFS) codes, that are compressed number sets. A set of codes for a specified image is calculated by means of a systematic process, so that when the IFS codes are applied to an appropriate set of image blocks an image is produced which is an actual adjacent calculation of the original.

2.7 Different Types of Images:

In an original position, not only in RAM but also in storage, images may need a huge quantity of memory. Compression of images decreases the storing space which is necessary for image as well as the desired bandwidth while running the image through a network. There are various kinds of images which have different characteristics.

a) JPG: For portraits and related constant tone images, the JPG image is adjusted to include numerous color numbers. It can stock 24-bit color data. The amount of JPG compression is variable [16].

b) PNG (Portable Network Graphics): PNG images are incredible for cooperative documents such as web pages, but are not suitable for print. PNG images are lossless image. So we can compress the image and rebuild the original image without losing the quality of the original image.

c) GIF (Graphics Interchange Formats): GIFs are another type of image. This image is suitable for images which have less than 256-(2^8) colors, grayscale images and black and white text. The initial drawback of a GIF is that it merely works on images with 8-bits per pixel or less, which means 256 or fewer colors. Maximum color images are 24 bits per pixel. To store these in GIF format that must first alter the image from 24 bits to 8 bits. GIF is a lossless image file format. Therefore, GIF is "lossless" only for images with 256 colors or less. The true color image, GIF may "lose" 99.998% of the colors. It is not appropriate for photographic images, as it can comprise only 256 colors per image [17].

d) TIFF (Tagged Image File Format): This is a versatile layout that might be lossless or lossy compression. It generally keeps 8 bits or 16 bits per color (red, green, blue) for 24-bit and 48-bit totals, correspondingly. The specifics of the image storage algorithm are comprised as amount of the file. In exercise, TIFF is used absolutely as a lossless image storage format which uses no compression at all. TIFF files are not suitable for web images. They only produce big files, and more prominently, most web browsers will not display TIFFs.

e) BMP: Within the Microsoft Windows OS, the Bitmap (BMP) image layout uses visuals files. Naturally, BMP image files are not compressed, therefore they are enormous. The importance is that in Windows software their simplicity, broad acceptance, and use.

f) RAW: This kind of image format basically used for lossless compression. It generates files that are much lesser than TIFF formats. The disadvantage of this image is that there is a diverse format for every manufactures and so it has to use the manufacture's software to observe the images [18].

CHAPTER 3
HUFFMAN CODING

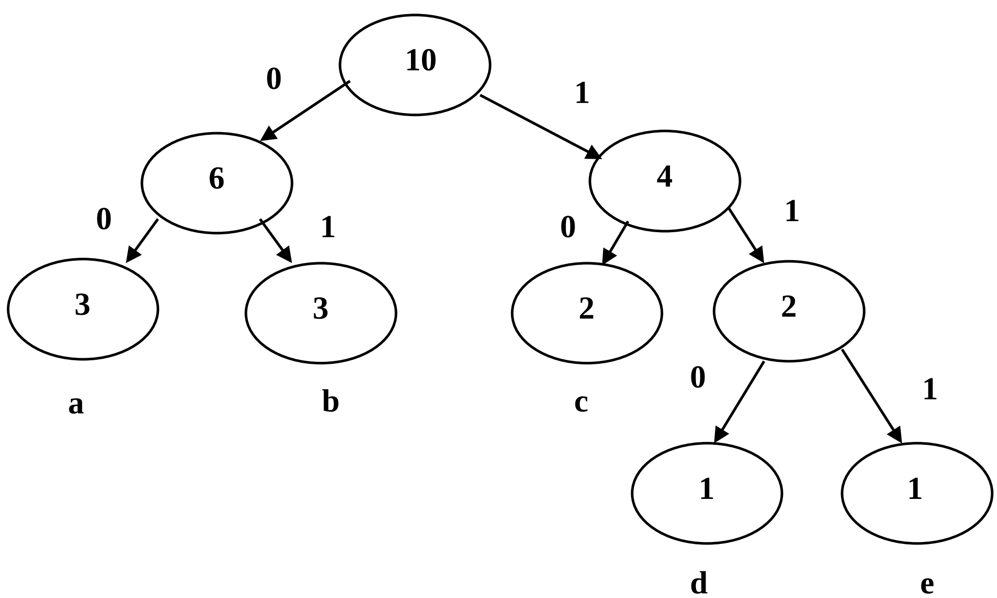
3.1 What is Huffman Coding?

Huffman coding is David Huffman's traditional data compression schemes. It is prefix code which is in optimum state that is produced from a collection of probabilities and used in several compression presentations. Such protocols are of variable length of code which uses an important amount of bits. This method creates a drop in the average length of the code and hence the entire dimension of compressed data is lesser compared to the original data. The system of Huffman code is established on the two interpretations.

a) Most commonly happened symbols can consume smaller words of code compared to less frequently occurring symbols.

b) The two least frequently occurring symbols will have the same size.

The Huffman code is considered by integrating the smallest possible symbols, besides this procedure is repetitive till merely two prospects of two composite symbols are left and therefore a code tree is created and Huffman codes are achieved from the code tree labeling [19].



In this coding, it is expected that each pixel intensity has related with it a particular probability of occurrence. Besides, this probability is especially invariant. A binary code is allocated to every intensity value through Huffman coding by shorter intensity code [20].

The following parameters are involved in Huffman coding:

- Entropy
- Average length
- Efficiency
- Variance

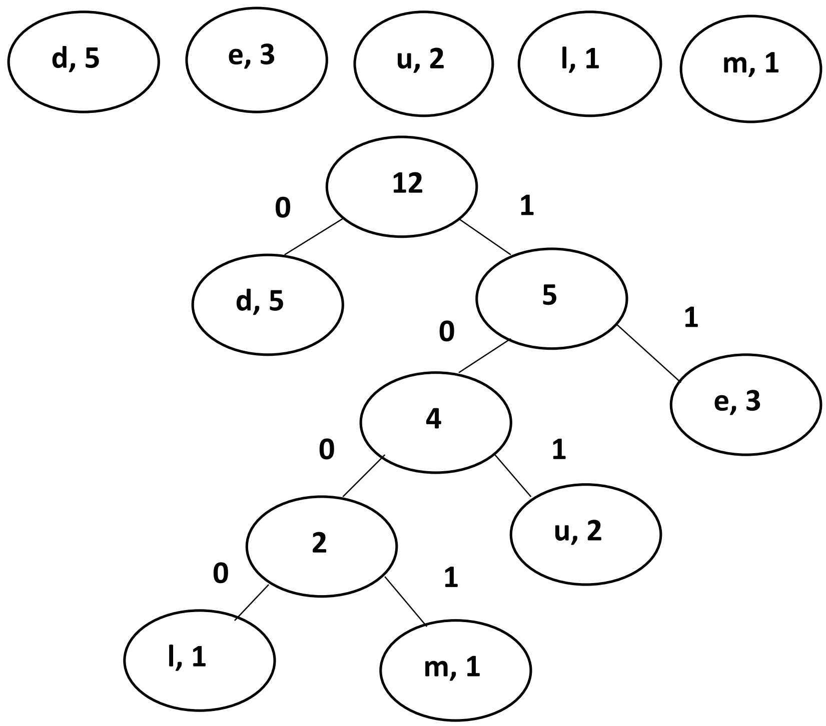
3.2 Huffman Algorithm:

An entropy encoding algorithm is Huffman coding, which in information theory and computer science is simply very useful for lossless data compression. It denotes the usage of a table of a variable-length code to represent a source symbol (e.g. a character in a file) wherever this code table was constructed in a specific manner established on the predictable probability of incidence for any probable significance of the source symbol.

Huffman coding usages a particular technique to pick the prototype for every symbol, which results in a prefix-free code (i.e., the bit string signifying any precise symbol is never a prefix of the bit series representing any other symbol) specifying the greatest common characters by utilizing smaller strings of bits compared to those used for less common source symbols. Huffman was capable to scheme the most effective compression technique of that form: no other mapping of distinct source symbols to distinctive strings of bits generates a smaller typical output size while the authentic symbol frequencies do not disagree with those used to generate the code. A system was established later to organize this in linear time if input probabilities are preserved [21].

3.3 Basic Technique of Huffman Coding:

Suppose, there is a source which generates five different symbols (e, d, u, l, m) with frequency (3, 5, 2, 1, 1). Generating a binary tree from left to right by selecting the two smallest likely symbols, placing them together to create another equal symbol with a likelihood of adding the two symbols. Until it has one symbol, it's a continuous process. Then the tree has been read backwards, from right to left, conveying various bits to various branches.



This system performs by forming a binary tree of nodes. In the tree, bit '0' signifies the left node and bit '1' signifies the right node.

The ultimate Huffman code:

SYMBOL	CODE
e	11
d	0
u	101
l	1000
m	1001

3.4 Huffman Decoding:

When code is generated, either coding, decoding or both is done in an easy look-up table way [22]. Itself the code is an instant exceptionally decodable code block. A block code is entitled, as every source symbol is drawn into a fixed code symbol structure. This one is rapid since every code word can be decoded devoid of referencing successive symbols in a sequence of code symbols. This is distinctively decodable, as any sequence of code symbols might be only one-way decoding. Any series of Huffman encoded symbols can therefore be decoded by observing the separate symbols of the sequence in a left to accurate way [23].

3.5 Advantages of Huffman Coding:

- This algorithm is very relaxed to implement
- This is extensively used in lossless image compression
- Though each character is coded with a diverse number of bits, the receiver will spontaneously calculate the character whatever their direction.
- It offers the more effective results than other methods of compression.

3.6 Disadvantages of Huffman Coding:

- This algorithm differs with various formats.
- The effectiveness relies on the accurateness of the arithmetical model used and form of image.
- This one is mandatory to deliver Huffman table when the compressed file is opened, otherwise it will not be decoded by the decompression which creates overhead.

3.7 Why Huffman Coding in Image Processing?

Image compression is relatively simpler and easier for implementation by using the Huffman coding algorithm. Huffman coding is basically under the lossless compression technique. As we get better quality image after lossless image compression, that's why Huffman coding is used in image compression.

CHAPTER 4

PARAMETERES FOR IMAGE COMPRESSION

4.1 Parameters for Image Quality Measurement:

Some significant parameters are involved in image compression which measure the performance of the compressed and uncompressed images.

4.1.1 Compression Ratio (CR):

The most important term of image compression algorithm quality calculation is compression ratio. This one is used to measure the compression efficiency. It is the compressed image dimension proportion with the uncompressed image width. If the compression ration rises, the image quality should improve as well.

$$\text{Compression Ratio} = \frac{\text{Size of uncomressed image}}{\text{Size of compressed image}}$$

4.1.2 Peak Signal to Noise Ratio (PSNR):

Peak signal to noise ratio is another essential parameter. For better quality image, PSNR must be high as much as possible. PSNR refers to the ratio of peak to average noise signal energy. While there is less variation between the original image and the compressed image, the PSNR is high, and therefore the image quality is also high. PSNR is based on image's mean square error (MSE).

$$\text{PSNR}= 10 \log \frac{MAX^{2}}{MSE}$$

4.1.3 The Mean Squared Error (MSE):

This parameter is very useful for examining image's quality. For improved quality image, the MSE should be less as possible as. The dissimilarity between original data and compressed data of image is known as the mean square error (MSE). It is inversely proportional to PSNR, since the increase in PSNR decreases with MSE. Compression of the image is lossless while zero is MSE.

4.1.4 Bits Per Pixel (BPP):

It is data (bits) kept per image pixel. It is a relation between the number of bits in the compressed image in the original image and the total amount of pixels.

The compressed image requires a number of bits per pixel.

$$BR = (b/CR)$$

Here, b= No. of bits per pixel.

Although bit rate is high, storing an image requires a huge memory. High bit rate specifies the extra colors of the image so that the bit rate must be lower.

4.1.5 Retained Signal Energy:

The retained energy refers to the quantity of information reserved by an image after compression and decompression and this is equal to the pixel number of squares. If the preserved energy is 100%, then the compression is called the image can be accurately reconstructed.

Such value specifies the quantity preserved by a proportion of the initial signal energy in the compressed signal.

$$RSE\ (\%) = \frac{||x(n)||^2}{||r(n)||^2} \times 100$$

4.1.6 Percentage of Zero Coefficients:

Percentage of zeroes of an image can be calculated as,

$$100 \times (\% \text{ of current decomposition}) / \% \text{ of coefficient}$$

CHAPTER 5

COMPRESSION AND DECOMPRESSION OF LOSSLESS, LOSSY AND MEDICAL IMAGES

5.1 MATLAB Wavelet Toolbox:

The **Wavelet Toolbox** is a function group constructed on the **MATLAB** Environment of Technical Computing. It offers tools for investigating and combining signals and images, and tools for arithmetic applications, using **MATLAB** wavelets and wavelet packets.

Numerous MATLAB products, for example, Curve Fitting Toolbox™, Signal Processing Toolbox™, and Control System Toolbox™ comprise custom user interface applications. We can also make our own custom applications for other people to use, containing their respective UIs.

The 2-D continuous wavelet transform (CWT) app allows us to evaluate the image data and store the results of that investigation to the MATLAB workspace. The 2-D CWT is a space-scale representation of an image.

5.2 Working Block Diagram of Compression and Decompression:

The following figure 5.1 shows the research's compression and decompression work block diagram.

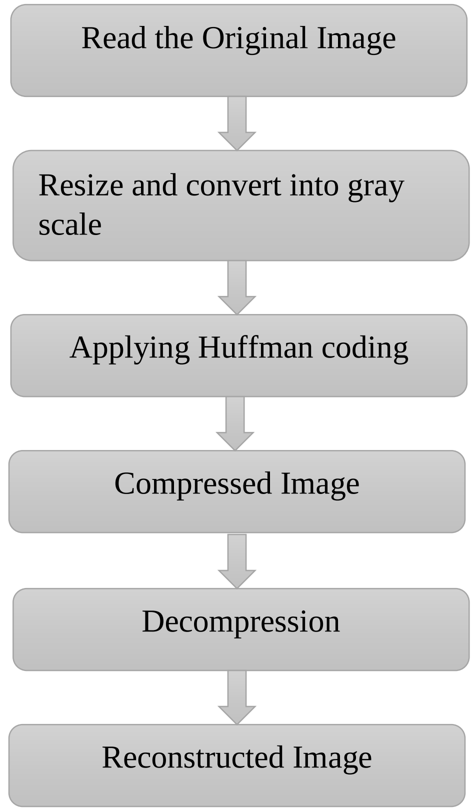


Figure 5.1: Working Flow Chart of Compression and Decompression.

5.3 Working Process:

The block diagram exposed in the Figure 5.1 shows a standard image compression process. At first, an image is read. Then the image is being resized and converted into gray scale. After that, Huffman coding is applied to the resized image and final output image is generated.

5.4 Algorithm:

The overall algorithm that is used for the compression and decompression of lossless and lossy image is given below:

Step 1: By using “Read” function an image has been selected for compression

Step 2: Resize the image into 256×256

Step 3: Transform the original color image into gray scale image

Step 4: Call a function which will compute the probability of each symbol by counting the zeroes value.

Step 5: A function is called that can find the symbols

Step 6: Derive the symbols and remove the ones with sufficiently low probability (For lossy compression).

Step 7: After that, a Huffman directory is created and Huffman encoding is applied that results in a compressed image.

Step 8: After the compression, the decompression is performed by using Huffman decoding and original image is reconstructed.

The details description of these step by step process is given below:

5.4.1 Read Images for Compression and Decompression:

For this analysis, both lossless and lossy images have been selected and analyzed the performance of different parameters. One medical image is also being examined here.

We know that, the PNG (Portable Network Graphics) images are lossless image. So we can compress the image and reconstruct the original image without losing the quality of the original image. That's why, one ".png" image has been selected for lossless compression and decompression.

On the other hand, as JPG compression is lossy, for this reason one ".jpg" image has been selected for lossy compression and decompression.

One medical (X-ray) image is also selected for the analyzing purpose and compare the result. Images of different size, such as 256×256 and 512×512 have been analyzed separately and compare the result of the performance. For the compression, Huffman coding is used.

(a) (b) (c)

Figure 5.2: Original Image (a) PNG (b) JPG and (c) Medical Image

5.4.2 Resize Image and Convert into Gray Scale:

At first, the original .png and .jpg and medical images are resized by 256×256. These original images are RGB (R, G, B) images.

(a) (b) (c)

Figure 5.3: Original and resized (256 × 256) RGB images. (a) PNG (b) JPG and (c) Medical

Then these images are resized by 512×512 so that the resolution of the images is being increased.

(a) (b) (c)

Figure 5.4: Original and resized (512 × 512) RGB images. (a) PNG, (b) JPG and (c) Medical

Then these color images (RGB images) are converted into gray level image. Gray scale images can simply reduce the complexity from a 3D pixel value (R, G, B) to a 1D value. The better output can be produced by gray scale images.

(a) (b) (c)

Figure 5.5: Original Gray scale images (256×256) (a) PNG, (b) JPG and (c) Medical

(a) (b) (c)

Figure 5.6: Original Gray scale images (512×512): (a) PNG, (b) JPG and (c) Medical

5.4.3 Lossless Image Compression:

A function is called to determine the symbols (for example, pixel value). Then another function is called to determine the probability of symbol for every pixel. After that, Huffman coding is performed to the respective symbols which resulting in a compressed data. Fig. 5.7 shows the lossless compressed images for different resolutions.

(a) (b)

Figure 5.7: Compressed image (Lossless: PNG): (a) 256×256 (b) 512×512

5.4.4 Lossy image compression:

As Huffman coding technique is used for lossless compression, in this report, this useful method is also used for lossy compression. But this is possible when we derive our symbols and remove the ones with sufficiently low probability or merge them with the nearest higher probability symbol. Then Huffman coding can be done on what is left. Fig. 5.8 shows the lossy compressed image of 256×256 and 512×512.

(a) (b)

Figure 5.8: Compressed image (Lossy: JPG): (a) 256×256 (b) 512×512

If we compare the compressed lossless and lossy image, it might be shown that the lossless PNG image's quality remains unchanged and it is better for the higher resolution image. There is no degradation of the quality from the original image. But for lossy image, 100% quality image cannot be generated after compression.

But if this improved Huffman algorithm technique is applied, then better output of the lossy images can be gained after compression.

5.4.5 Lossless Image Decompression:

After the compression, the decompression is done using Huffman decoding for lossless (.png) images and reconstructed images are generated after the decompression.

(a) (b)

Figure 5.9: Reconstructed PNG Gray Scale Images: (a) 256×256 (b) 512×512

Fig 5.9 shows the reconstructed images of lossless images after decompression. As the original gray scale images are used for compression, that's why, reconstructed images are also gray scale images. After generating the reconstructed images, no quality degradation is found although the size of the images is different.

5.4.6 Lossy Image Decompression:

After the lossless decompression, the lossy image decompression is performed for different size images and reconstructed images are generated.

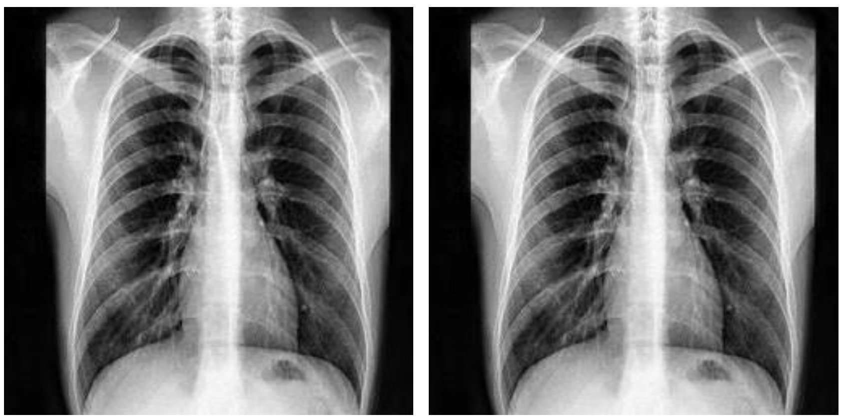
(a) (b)

Figure 5.10: Reconstructed JPG Gray Scale Images: (a) 256×256 (b) 512×512

Although Huffman coding algorithm is applicable for lossless image compression, but this improved Huffman coding is used for lossy image compression also and the better quality image can be reconstructed.

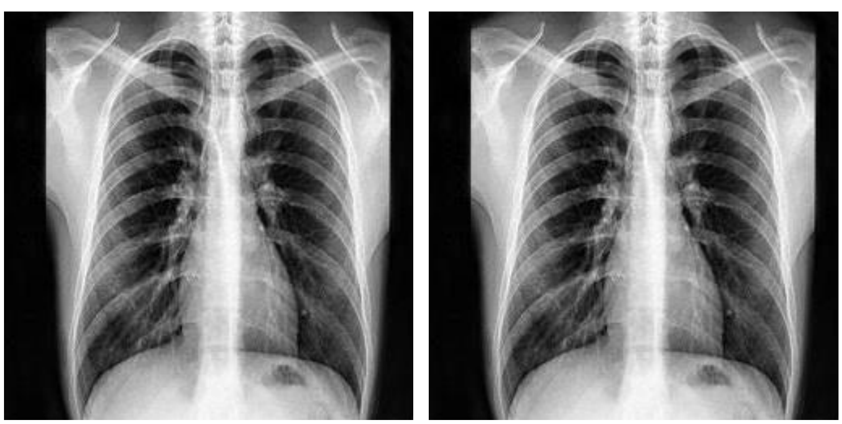
5.4.7 Medical Image Compression and Decompression:

Medical image can also be compressed and decompressed using the same Huffman coding algorithm and comparatively better output is gained after the reconstruction.



(a) (b)

Figure 5.11: Compressed & Reconstructed Medical Images of 256×256



(a) (b)

Figure 5.12: Compressed & Reconstructed Medical Images of 512×512

Figure 5.11 and 5.12 show the compressed and reconstructed medical images for two different resolutions and the quality of the images is unchanged.

5.5 Parameter Calculation:

MATLAB wavelet toolbox is used for calculating different parameters for lossless and lossy images.

Table 1: Calculated Parameters for Lossless Image

Parameters	256×256	512×512
CR	3.13	3.16
MSE	12.54	9.96
PSNR	22.56	31.46
BPP	0.7501	0.5678

Table 2: Calculated Parameters for Lossy Image

Parameters	256×256	512×512
CR	2.39	2.98
MSE	16.37	15.25
PSNR	17.07	19.34
BPP	0.9890	0.8054

Table 3: Calculated Parameters for Medical Image

Parameters	256×256	512×512
CR	2.83	2.98
MSE	11.32	11.10
PSNR	18.70	20.47
BPP	0.5924	0.3792

Table 1, table 2 and table 3 show different parameters that is calculated for lossless, lossy and medical images respectively. The first parameter is compression ratio (CR) which must be higher for better quality images. After analyzing the result, this might be said that the lossless image's quality is better than lossy image as it gives higher compression ratio. This compression ratio is quite better for high resolution image.

The second parameter is mean squared error (MSE) which should be less as possible as. If MSE for both images is compared, then it is found that the lossless image gives less error than lossy image. This error is also less for medical image also.

Another important parameter that is calculated is PSNR. For better quality image, PSNR must be high which is produced from lossless image.

BPP is another important parameter. Lossy image compression gives more BPP as compare to lossless image compression which means that lossy compression requires large memory space to store an image.

Although, a better performance is found for lossless image compression but for this Huffman coding algorithm technique, comparatively better performance can be measured for lossy image compression and medical image also. Comparatively better output is found for high resolution image than low resolution image.

5.6 Retained Energy and Number of Zeroes:

Another two important parameters for image compression, retained signal energy and number of zeroes have also been analyzed for lossless, lossy and medical image compression. MATLAB Wavelet toolbox is used for calculating these parameters.

The tool repeatedly picks thresholding levels to deliver a good initial balance between retaining the image's energy while reducing the number of coefficients that is needed to denote the image.

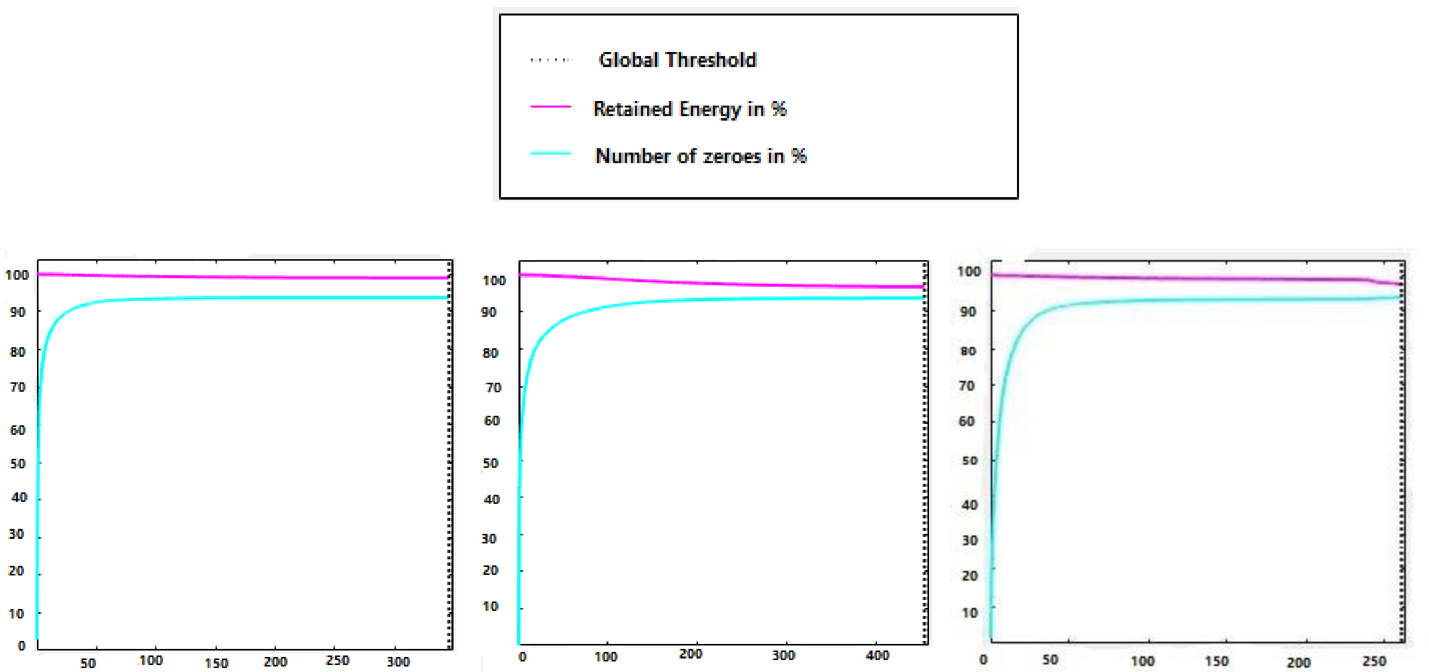
Table 4 and table 5 show the retained signal energy and number of zeroes for lossless, lossy as well as medical images. If the global thresholding is selected from the menu, then some of the retained energy has lost. It is not possible to gain 100% energy after compression.

Table 4: Retained Signal Energy & % of Zeros (256×256)

Parameters (%)	Lossless (.png image)	Lossy (.jpg image)	Medical Image
Retained Signal Energy	99.01	96.8	97.52
Number of zeroes	93.90	93.75	93.71

Table 5: Retained Signal Energy & % of Zeros (512×512)

Parameters (%)	Lossless (.png image)	Lossy (.jpg image)	Medical Image
Retained Signal Energy	99.12	99.07	97.47
Number of zeroes	93.75	93.79	93.73



(a) (b) (c)

Figure 5.13: Retained Signal Energy & Number of zeroes, (a) Lossless (b) Lossy (c) Medical

Retained energy should be more which means that the image is reconstructed more accurately after compression. Fig 5.8 also shows the graph of the retained energy and number of zeroes in %. Retained energy and number of zeroes for lossless image is better than lossy image which is almost 100% for greater resolution image.

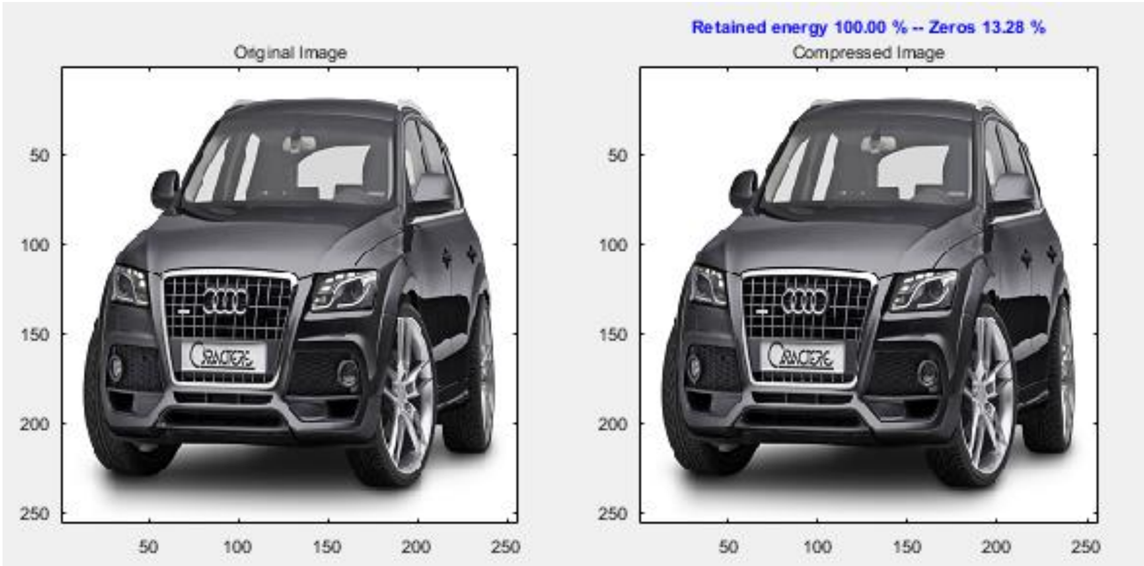


Figure 5.14: 100% retained energy for lossless image compression

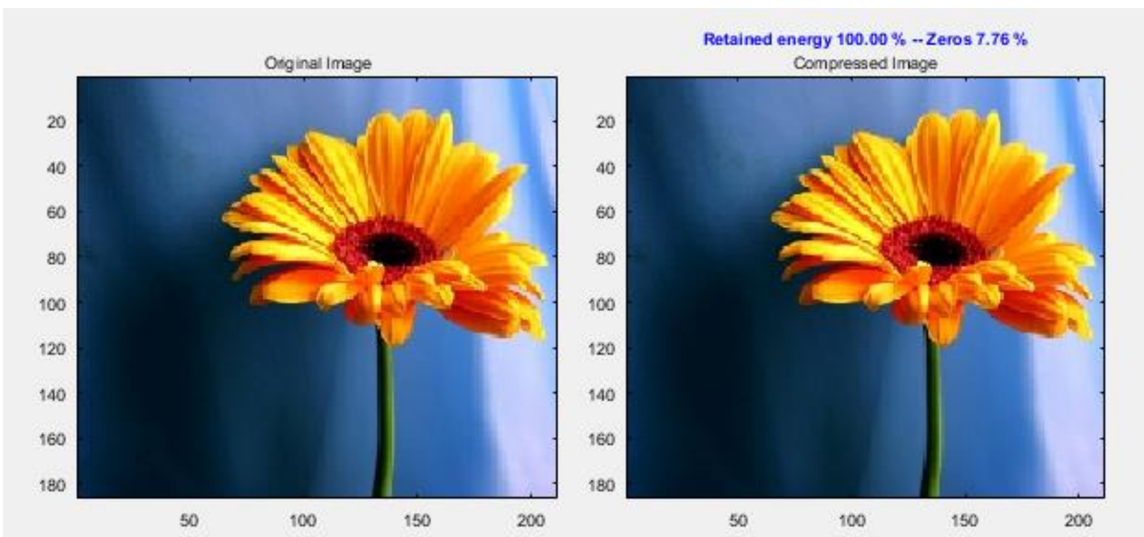


Figure 5.15: 100% retained energy for lossy image compression

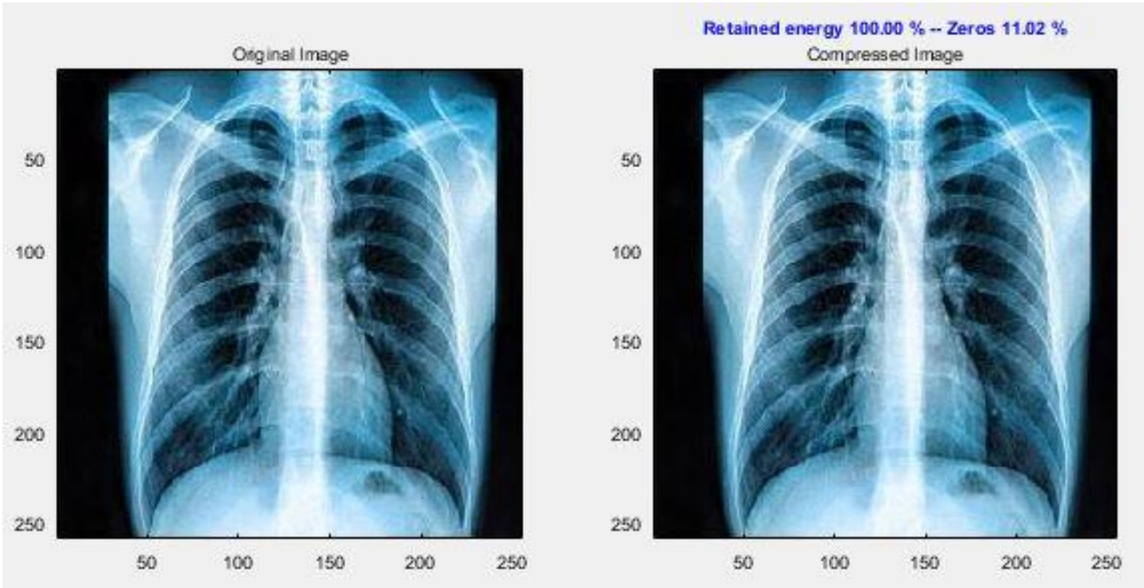


Figure 5.16: 100% retained energy for medical image

If the by ‘Level thresholding option’ and the ‘Remove near 0’ method is selected from the Select thresholding method menu, then 100% energy can be retained.

From fig 5.14, fig 5.15 and fig 5.16, it can be shown that the compression removes nearly half the coefficients, however no noticeable decline of the image appears.

CHAPTER 6

HISTOGRAM ANALYSIS FOR LOSSLESS, LOSSY AND MEDICAL IMAGES

6.1 What is Histogram?

Histograms are supposed to be the foundation for a quantity of spatial domain systems. In a simplified visual format, histograms are useful for setting the image statistics. An image histogram defines the frequency of an image's intensity values.

The histogram's x-axis denotes the collection of intensity (pixel) values while; y-axis denotes the number (frequency) of these intensity values in an image.

A digital image's histogram in the range [0, L-1] by intensity level is a discrete function defined as:

$$h\ (r_k) = n_k,$$

where r_k is the gray level of kth and n_k is the number of gray level r_k pixels in the image.

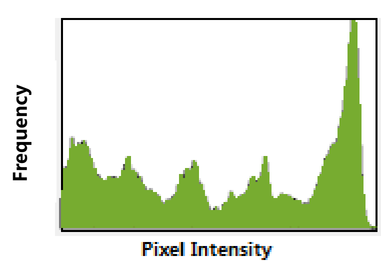


Figure 6.1: Histogram of image

6.2 Application of Histogram in Image Enhancement:

Histogram plays a key role in enhancing the digital images. The applications of histogram are given below:

- Histograms are very necessary to examine the image. The image's property can be predicted only by observing the histogram data.
- For increasing the brightness of an image histograms are used. From the histogram of an image, brightness might be adjusted.
- To modify the contrast of an image, histograms are very necessary.
- For image equalization, histogram is used also. To generate a high contrast image, the gray level intensities are extended along the x-axis.
- In thresholding, histograms are very useful as well.
- For increasing the visual presence of an image.
- An image histogram represents the difficulties that create for example, dynamic pixel size, contrast etc. in image acquisition.
- Histograms represent an extensive collection of susceptibilities like the effect of image compression, saturation, spikes, and gaps.
- The histogram shape expects information on the possibility of enriching the contrast [24].

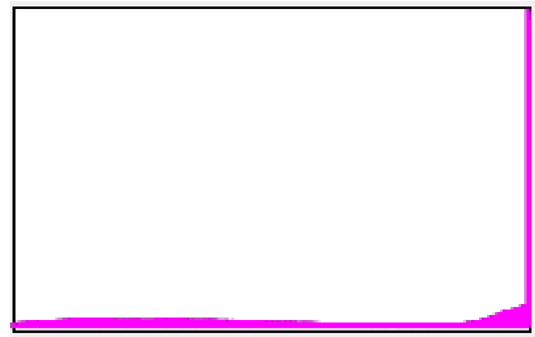
6.3 Normalized Histogram:

A system involving into converting the discrete intensity distribution into a discrete probability distribution is called a normalized histogram. For this, every histogram value has to be divided by the pixel number. Since a digital image is a discrete form of values which can be realized like a matrix, dividing every n_k by the array dimension that is the invention of the width by the image's length.

- Normalized histogram, $p(r_k) = n_k / M*N$, $k = 0, 1, 2 \ldots\ldots\ldots L-1$

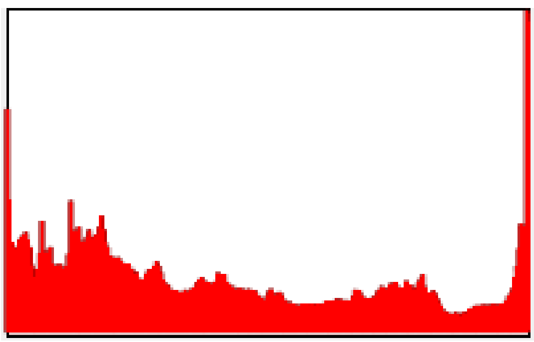
Where, M is the no of rows of the image and N is the no of column of the image [25].

(a)

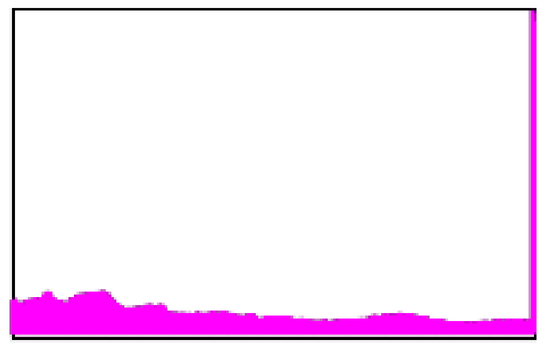


(b)

Figure 6.2: Normalized Histogram (a) Original Lossless (PNG) image (b) Compressed Lossless (PNG) image

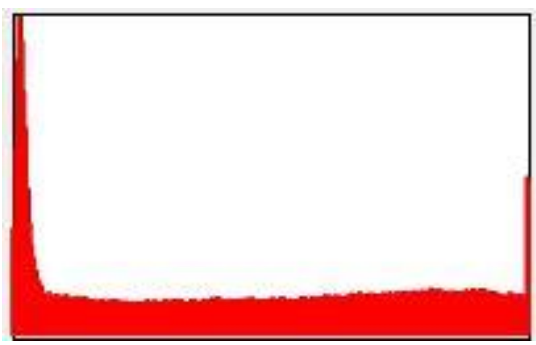


(a)

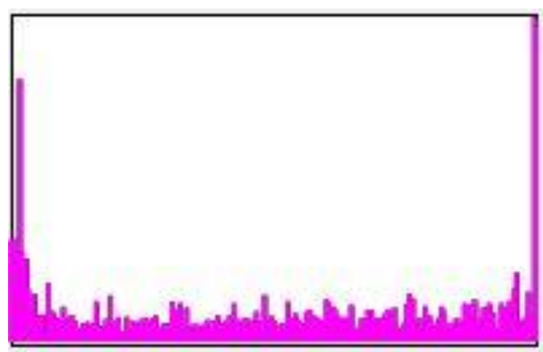


(b)

Figure 6.3: Normalized Histogram (a) Original Lossy (JPG) image (b) Compressed Lossy (JPG) image



(a)



(b)

Figure 6.4: Normalized Histogram (a) Original medical image (b) Compressed medical image

Figure 6.2, 6.3 and 6.4 show the normalized histogram of our selected and analyzed lossless, lossy images and medical image as well as their compressed images respectively.

6.4 Histogram Equalization:

A perfect image is one where all the gray levels have equal number of pixels. Histogram equalization is a method where our goal is not only to spread the dynamic range but also to have equal pixels at all gray levels.

Histogram equalization is a technique for analyzing images by modifying an image's contrast by transforming the histogram's intensity distribution. The aim of this system is to provide the cumulative probability function related to the image with a linear development.

This process of histogram equalization depends on the cumulative probability function (cdf) practice [26]. This cdf is a cumulative quantity of all the probabilities in its domain which is expressed through:

$$\text{cdf}(x) = \sum_{k=-\infty}^{x} P(k)$$

6.5 Histogram Analysis for Lossless Image:

The histogram for lossless (car image) original and reconstructed image has also been generated by this algorithm in MATLAB.

Figure 6.5: Original PNG image

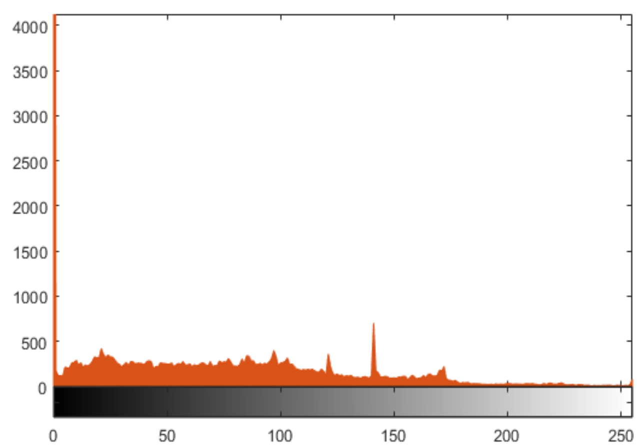


Figure 6.6: Original Histogram: PNG

Fig 6.5 and 6.6 shows the original gray scale image and original histogram respectively for lossless image.

Figure 6.7: Reconstructed PNG image

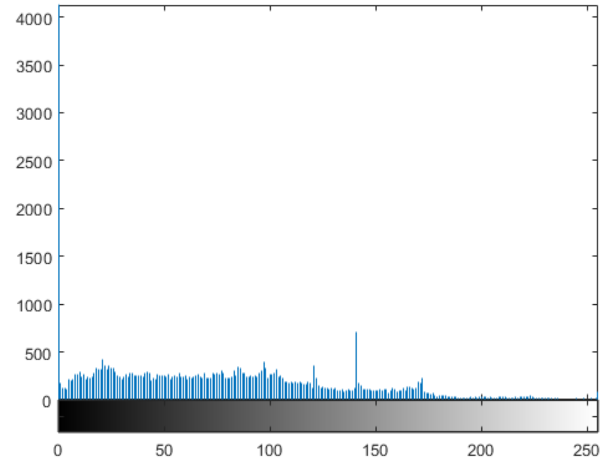


Figure 6.8: Reconstructed PNG image Histogram

Fig 6.7 and 6.8 shows the reconstructed image and reconstructed image histogram respectively for lossless image.

From the original and reconstructed image histogram, it can be clearly observed that there is no difference between them and the quality of the images remained unchanged.

6.6 Histogram Analysis for Lossy Image:

The histogram for lossy (flower image) original and reconstructed image has also been analyzed here.

Figure 6.9: Original JPG image

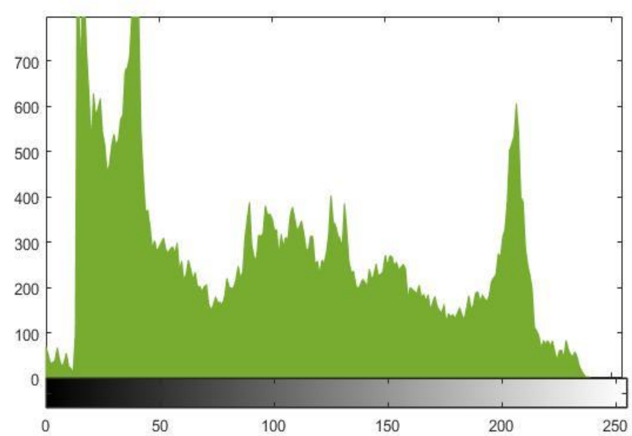


Figure 6.10: Original JPG Histogram

Figure 6.11: Reconstructed JPG image

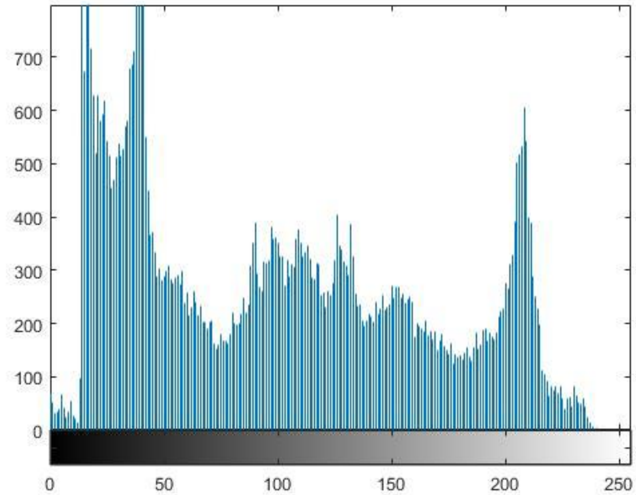


Figure 6.12: Reconstructed JPG image Histogram

Fig 6.10 and Fig 6.12 shows the original and reconstructed image histogram. Only the appearance of the figure is different but there is no difference between the quality of the images.

6.7 Histogram Analysis for Medical Image:

The following figure shows the histograms of original medical image and reconstructed image.

Figure 6.13: Original medical image

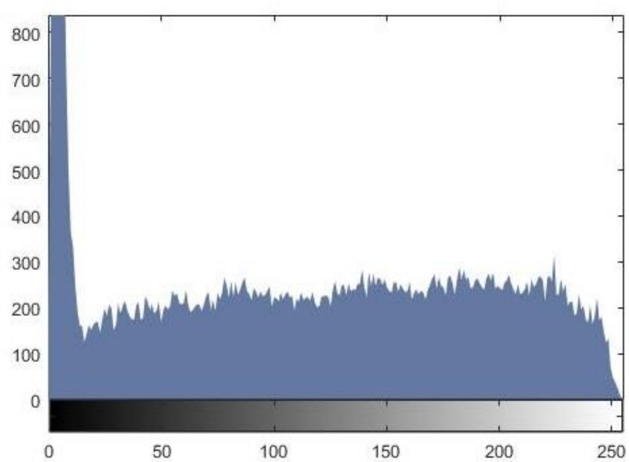


Figure 6.14: Original medical image Histogram

Figure 6.15: Reconstructed medical image

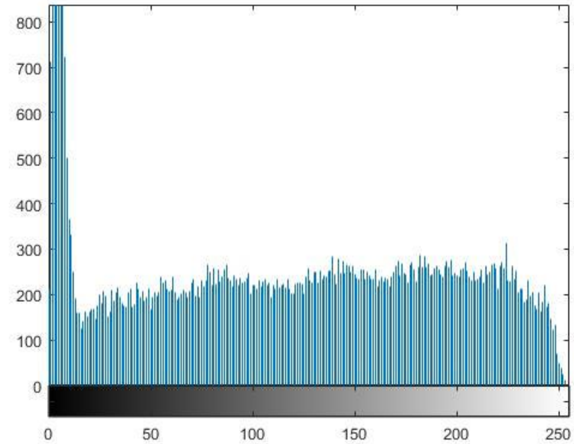


Figure 6.16: Reconstructed medical image Histogram

After the histogram analysis, it is found that the reconstructed image's histogram is same as the original image's histogram.

6.8 Comparative Analysis of this research:

It can be observed after the literature review that most of the authors worked with Huffman coding technique on lossless image compression, whereas, the main focus of this research is to compare the performance of different parameters and also the histograms not only for lossless compression but also for lossy image compression as well as for medical image. Also a Huffman coding algorithm is developed which is applicable for both compressions. If this algorithm is applied, then comparatively better output can be gained specially for higher resolution image.

CONCLUSION

In this report, different parameters have been used to analyze the performance of lossless, lossy and medical image compression. An improved Huffman coding algorithm is used for this analyzing purposes. This algorithm is applicable not only for lossless but also for lossy image compression as well as for medical image. Different parameters, such as, CR, MSE, PSNR, BPP, retained energy etc. are used for compression and decompression and compare the output. MATLAB wavelet toolbox is also used for this analysis. Histograms of the selected lossless, lossy and medical images have also been analyzed.

From this experiment it is found that this improved algorithm is applicable for both compression techniques and also for medical image. Due to its simpler mathematical calculation, the Huffman algorithm is relatively easier to determine different parameters. Moreover, lossless compression provides better performance than lossy compression for high resolution images than lower. Lossy compression also gives comparatively better result than other compression method.

References

[1] A.K. Jain. Image data compression: A review. Proc.IEEE. Vol.69. pp.349-389,1981.

[2] Mamta Sharma. Compression Using Huffman Coding. IJCSNS International Journal of Computer Science and Network Security. VOL.10 No.5. pp.133-141, May 2010.

[3] Majid Rabbani and Paul W. Jone. Digital Image Compression Techniques. Vol.TT7. Chapter-1. pp.3-4.

[4] Rachit Patel, Virendra Kumar, Vaibhav Tyagi, Vishal Asthana. A Fast and Improved Image Compression Technique Using Huffman Coding. IEEE. pp. 2283-2286, 2016.

[5] Nehal Markandey and Prof.Dr.Sonali Patil. Image Compression Using Huffman Coding. International Journal of Engineering and Computer Science ISSN: 2319-7242. Vol.6. pp.19999-20002, 2017.

[6] Vikash Kumar and Sanjay Sharma. Lossless Image Compression through Huffman Coding Technique and its application in Image processing using MATLAB. International Journal of Soft Computing and Engineering. Vol.7. 2017.

[7] Harpreet Kaur and Neelofar Sohi worked on applications of Histogram in Image Enhancement

[8] Sanjay Kumar Gupta worked on an algorithm for image compression using Huffman coding techniques.

[9] Jyoti A. Patil et all. Worked on lossless image compression using Huffman coding algorithm

[10] Akhand Pratap Singh, Dr. Anjali Potnis and Abhineet Kumar. A review on latest techniques of image compression. International Journal of Engineering and Technology.Vol.3, 2016

[11] Vikash Kumar and Sanjay Sharma. Lossless Image Compression through Huffman Coding Technique and its application in Image processing using

MATLAB. International Journal of Soft Computing and Engineering. Vol.7. 2017.

[12] Gaurav Vijayvargiya, Dr. Sanjay Silakari and Dr. Rajeev Pandey. A Survey: Various Techniques of Image Compression. International Journal of Computer Science and Information Security. Vol 11, 2013

[13] Trupti S Bobade and Anushri S.sastikar. Lossless method of image compression using Huffman coding techniques. International journal of engineering science & research technology, 2015

[14] G.M.Padmaja and P.Nirupama. Analysis of various image compression techniques. ARPN Journal of Science and Technology. Vol.2, 2012.

[15] Akhand Pratap Singh, Dr. Anjali Potnis and Abhineet Kumar. A review on latest techniques of image compression. International Journal of Engineering and Technology.Vol.3, 2016

[16] Asha Lata and Permender Singh. Review of image compression techniques. International Journal of Engineering Technology and Advanced Engineering. Vol.3, 2013.

[17] Athira B. Kaimal, S. Manimurugan and C.S.C. Devadass. Image Compression Techniques: A Survey. International Journal of Engineering Inventions. Vol.2. pp.26-28, 2013.

[18] Sonali Chawla, Meenakshi Beri and Ritu Mudgil. Image Compression Techniques: A Review. International Journal of Computer Science and Mobile Computing. Vol.3. pp.291-296, 2014

[19] Jagadish H. Pujar. A new lossless method of image compression and decompression using Huffman coding techniques. Journal of Theoretical and Applied Information Technology.

[20] S Jayaraman, S Esakkirajan and T Veerakumar. Digital Image Processing. pp.452

[21] Mamta Sharma. Compression Using Huffman Coding. IJCSNS International Journal of Computer Science and Network Security. VOL.10 No.5. pp.133-141, May 2010.

[22] Efficient Huffman decoding by MANOJ AGGRAWAL and AJAI NARAYAN.

[23] Jagadish H.Pujar and Lohit M. Kadlaskar. A new lossless method of image compression and decompression using Huffman Coding Techniques. Journal of Theoretical and Applled Information Technology.

[24] Harpreet Kaur and Neelofar Sohi. A Study for Applications of Histogram in Image Enhancement. The International Journal of Engineering and Science (IJES). Vol.6. pp.59-63

[25] Rafael C. Gonzalez and Richard Eugene Woods. Digital Image Processing, 3rd edition.

[26] Harpreet Kaur and Neelofar Sohi. A Study for Applications of Histogram in Image Enhancement. The International Journal of Engineering and Science (IJES). Vol.6. pp.59-63

PSEUDOCODE

READ an image in MATLAB environment

CONVERT the image into a gray scale image with values from 0 to 255 for each pixel

GET the dimension of the image

CREATE a frequency array of size 256

COUNT the number of zeroes

FIND the probability of occurrence

COUNT total number of symbols

COMPUTE the cumulative probability

CREATE Huffman directory with probability and symbol

APPLY Huffman coding

PERFORM Huffman decoding

RECONSTRUCT the original image.