Image compression using Weighted Average and Least Significant Bit Elimination Approach

S.Subbulakshmi[1] Ezhilarasi Kanagasabai[2]
[1,2]Department of Information Technology
[1,2]KCG College of Technology Chennai, India

Abstract— Image compression plays a very vital role in the transmission and storage of image information. The main aim of image compression is to represent an image in the smallest number of bits without losing the essential information content contained by an original image. The image compression techniques are used to trim down the file size. The Proposed method for image compression uses Outline detection, wavelet encoding and bit plane encoding. The outline detection is accomplished by determining the maximum gradient value of a pixel from its three adjacent pixels. Wavelet encoding is a weighted averaging and differencing scheme. Bit Plane Encoding uses the data as sets of significant bits: the most significant bit of each piece of data forms the first "bit plane", the second most significant bit of each item forms the second bit plane, and so on. It allows the user to determine the quality of the compressed image during the compression process.

Key words: Image Compression, Wavelet Encoding, Arithmetic Bit Plane Encoding and Outline Detection

I. INTRODUCTION

In recent years the usage of digital images is increased in different areas like medical, satellite and web applications. Since the digital image files are so large, maintaining image in its raw, uncompressed form requires immense physical storage resources, and accessing raw data requires high-bandwidth networks and large memory workstations. So the reduction in file size allows more images to be stored in a given amount of disk or memory space. It also reduces the time required for images to be sent over the Internet or downloaded from Web pages.

The objective of image compression is to reduce unimportance and redundancy of the image information in order to be able to store or convey data in an efficient form. The image compression techniques are used to reduce the file size. Three kinds of compression and data loss techniques are: uncompressed and lossless, compressed and lossless and compressed and lossy.

The performance of compression algorithms are measured in terms of memory usage, CPU usage, and I/O bandwidth. Quality of an image yielded by the compression algorithms may fall in any one of the criteria

Numerically lossless: This level of compression typically yields a 2:1 compression ratio, for a 50% reduction in storage space. Lossless compression should be used when it is critical that all bits of the original image be preserved. This is the case for archival storage, as well as for uncommon workflows where no possible loss of precision is ever acceptable.

Visually lossless: This level of compression is typically 20:1 for RGB and 10:1 for grayscale imagery. This is the most common level of compression quality used, as it preserves the appearance of the imagery for most workflows, including use of your imagery as a background layer and for many forms of visual analysis and exploitation.

Lossy: Beyond 20:1, image degradation and artifacts can appear, although often not too significantly until ratios of 40:1 or 50:1. Such lossy quality may be acceptable when the imagery is used only as a background layer for appearance or when the image quality is less important than the storage size or speed, such as for informal visual inspections

II. IMAGE COMPRESSION TECHNIQUES

A. Types of Image Compression

On the basis of our requirements image compression techniques are broadly split in to the following two major categories.

1. *Lossless image compression*
2. *Lossy image compression*

1) Lossless Compression Techniques:

Lossless compression compresses the image by encoding all the information from the original file, so when the image is decompressed, it will be exactly identical to the original image. Examples of lossless [14] image compression are PNG and GIF. When to use a certain image compression format really depends on what is being compressed.

a) Run Length Encoding:

Run-length encoding (RLE) is a very simple form of image compression in which replace adjacent message of equal value with their Counts. It is used for sequential [13] data and it is helpful for repetitive data. In this technique replaces sequences of identical symbol (pixel), called runs. The Run length code for a grayscale image is represented by a sequence {Vi , Ri } where Vi is the intensity of pixel and Ri refers to the number of consecutive pixels with the intensity Vi as shown in the figure. This is most useful on data that contains many such runs for example, simple graphic images such as icons, line drawings, and animations. It is not useful with files that don't have many runs as it could greatly increase the file size. Run-length encoding performs lossless image compression [15]. Run length encoding is used in fax machines.

65 65 65 70 70 70 70 72 72 72

{65,3} {70,4} {72,3}

b) Entropy Encoding:

In information theory an entropy encoding is a lossless data compression scheme that is independent of the specific characteristics of the medium. One of the main types of entropy coding creates and assigns a unique prefix-free code for each unique symbol that occurs in the input. These entropy encoders then compress the image by replacing each

All rights reserved by www.ijsrd.com

fixed-length input symbol with the corresponding variable-length prefix free output codeword.

c) Huffman Encoding:

In computer science and information theory, Huffman coding is an entropy encoding algorithm used for lossless data compression. It was developed by Huffman. Huffman coding [2] today is often used as a "back-end" to some other compression methods. 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. The pixels in the image are treated as symbols. The symbols which occur more frequently are assigned a smaller number of bits, while the symbols that occur less frequently are assigned a relatively larger number of bits. Huffman code is a prefix code. This means that the (binary) code of any symbol is not the prefix of the code of any other symbol.

d) Arithmetic Coding :

Arithmetic coding is a form of entropy encoding used in lossless data compression. Normally, a string of characters such as the words "hello there" is represented using a fixed number of bits per character, as in the ASCII code. When a string is converted to arithmetic encoding, frequently used characters will be stored with little bits and not-so-frequently occurring characters will be stored with more bits, resulting in fewer bits used in total. Arithmetic coding differs from other forms of entropy encoding such as Huffman coding [4] in that rather than separating the input into component symbols and replacing each with a code, arithmetic coding encodes the entire message into a single number.

e) Lempel–Ziv–Welch Coding:

Lempel–Ziv–Welch (LZW) is a universal lossless data compression algorithm created by Abraham Lempel, Jacob Ziv, and Terry Welch. It was published by Welch in 1984 as an improved implementation of the LZ78 algorithm published by Lempel and Ziv in 1978. LZW is a dictionary based coding. Dictionary based coding can be static or dynamic. In static dictionary coding, dictionary is fixed when the encoding and decoding processes. In dynamic dictionary coding, dictionary is updated on fly. The algorithm is simple to implement, and has the potential for very high throughput in hardware implementations. It was the algorithm of the widely used UNIX file compression utility compress, and is used in the GIF image format. LZW compression became the first widely used universal image compression method on computers. A large English text file can typically be compressed via LZW to about half its original size.

2) Lossy Compression Techniques:

Lossy compression as the name implies leads to loss of some information. The compressed image is similar to the original uncompressed image but not just like the previous as in the process of compression [3] some information concerning the image has been lost. They are typically suited to images. The most common example of lossy compression is JPEG. An algorithm that restores the presentation to be the same as the original image is known as lossy techniques. Reconstruction of the image is an approximation of the original image, therefore the need of measuring of the quality of the image for lossy compression technique. Lossy compression technique provides a higher compression ratio than lossless compression.

Major performance considerations of a lossy compression scheme include:

- Compression ratio
- Signal to noise ratio
- Speed of encoding & decoding

Lossy image compression techniques include following schemes:

a) Scalar Quantization:

The most common type of quantization is known as scalar quantization. Scalar quantization, typically denoted as Y=Q (x), is the process of using a quantization function Q to map a scalar (one-dimensional) input value x to a scalar output value Y. Scalar quantization can be as simple and intuitive as rounding high precision numbers to the nearest integer, or to the nearest multiple of some other unit of precision.

b) Vector Quantization:

Vector quantization (VQ) is a classical quantization technique from signal processing which allows the modeling of probability density functions by the distribution of prototype vectors. It was originally used for image compression. It works by dividing a large set of points (vectors) into groups having approximately the same number of points closest to them. The density matching property of vector quantization is powerful, especially for identifying the density of large and high-dimensioned data. Since data points are represented by the index of their closest centroid, commonly occurring data have low error, and rare data high error. This is why VQ is suitable for lossy data compression. It can also be used for lossy data correction and density estimation.

III. RELATED WORK

Techniques used for compression and the quality of the image are analyzed based on the noise factor and compression ratio from the below mentioned papers [18].

In this FMM (Five Module Method) Model convert each pixel value in 8x8 blocks [1] into a multiple of 5 for each of RGB array. After that the value could be divided by 5 to get new values which are bit length for each pixel and it is less in storage space than the original values which is 8 bits. The advantage of their method is it provided high PSNR (peak signal to noise ratio) although it is low CR (compression ratio). This method is appropriate for bi-level like black and white medical images where the pixel in such images is presented by one byte (8 bit).

The hybrid image compression technique inherits the properties of localizing the global spatial and frequency correlation from wavelets and classification and function approximation tasks from modified forward-only counter propagation neural network (MFOCPN) for image compression. In this scheme several tests are used to investigate the usefulness of the proposed scheme. In this

All rights reserved by www.ijsrd.com

paper, they explore the use of MFO-CPN [5] networks to predict wavelet coefficients for image compression. In this the analysis results in higher compression ratio but the quality of the reconstructed image is not good.

The paper [6] provides an image compression method which is capable to perform both lossy and lossless compression. A threshold value is associated in the compression process, different compression ratios can be achieved by varying the threshold values and lossless compression is performed if the threshold value is set to zero.

The bi-level image compression technique using neural networks is the lossy image compression technique. In this method, the locations of pixels of the image are applied to the input of a multilayer perceptron neural network [7]. The output the network denotes the pixel intensity 0 or 1. The final weights of the trained neural network are quantized, represented by few bites, Huffman encoded and then stored as the compressed image. Huffman encoded and then stored as the compressed image. In the decompression phase, by applying the pixel locations to the trained network, the output determines the intensity. The results of this technique provide High compression ratios as well as high PSNRs were obtained using the proposed method.

In encryption and compression of an image technique, stream cipher is used for encryption of an image after that SPIHT [8] is used for image compression. In this paper stream cipher encryption is carried out to provide better encryption used. SPIHT compression provides better compression as the size of the larger images can be chosen and can be decompressed with the minimal or no loss in the original image. Thus high and confidential encryption and the best compression rate has been energized to provide better security the main scope or aspiration of this paper is achieved.

The paper [9] provides SPIHT and EZW algorithms with Huffman encoding using different wavelet families and after that compare the PSNRs and bit rates of these families. These algorithms were tested on different images, and it is seen that the results obtained by these algorithms have good quality and it provides high compression ratio as compared to the previous exist lossless image compression techniques.

The spatial domain of lossless image compression algorithm [10] use reduction of size of pixels for the compression of an image. In this the size of pixels is reduced by representing pixel using the only required number of bits instead of 8 bits per color. This proposed algorithm has been applied on asset of test images and the result obtained after applying algorithm is encouraging. In this paper they also compared to Huffman, TIFF, PPM-tree, and GPPM. In this paper, they introduce the principles of PSR (Pixel Size Reduction) lossless image compression algorithm. They also had shows the procedures of compression and decompression of their proposed algorithm.

The implementation of multiwavelet transform coding for lossless image compression describes the performance of the IMWT (Integer Multiwavelet Transform) [11] for lossless studied. The IMWT provides good result with the image reconstructed. In this paper the performance of the IMWT for lossless compression of images with magnitude set coding have been obtained. In this proposed technique the transform coefficient is coded with a magnitude set of coding & run length encoding technique. The performance of the integer multiwavelet transform for the lossless compression of images was analyzed. It was found that the IMWT can be used for the lossless image compression. The bit rate obtained using the MS-VLI (Magnitude Set-Variable Length Integer Representation) with RLE scheme is about 2.1 bpp (bits per pixel) to 3.1 bpp less then that obtain using MS-VLI without RLE scheme.

The new modified international data encryption algorithm to encrypt the full image [12] in an efficient secure manner and encryption after the original file will be segmented and converted to other image file. By using Huffman algorithm the segmented image files are merged and they merge the entire segmented image to compress into a single image. Finally they retrieve a fully decrypted image. Next they find an efficient way to transfer the encrypted images to multipath routing techniques. The above compressed image has been sent to the single pathway and now they enhanced with the multipath routing algorithm, finally they get an efficient transmission and reliable, efficient image.

IV. PROPOSED WORK

The compression technique as shown in Figure 1 consists of following modules:

A. Outline Detection
B. Wavelet Encoding
C. Bit Plane Encoding

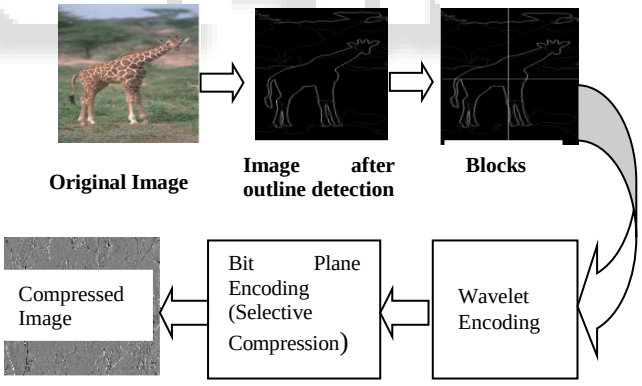

Fig. 1: Process of Image Compression

A. Outline Detection

A matrix 'T' of order m x n is used to represent the image of width n and height m. The possible values for $T_{i,j}$ are in the range [0,255] , for any i=1..m, any j=1..n. Let S be a matrix of the equal order as of 'T' for storing the gradient value, with a range of [0,255]. Let B be a binary matrix of the equal order as of 'T' and storing the brand of the pixel if it is in edge or not, if it is in edge the value 1 is assigned otherwise 0 is assigned.

The outline detection is accomplished by determining the maximum gradient value of a pixel from its three adjacent pixels as shown in Figure 2. If the maximum value of the gradients fulfills certain prerequisite then the corresponding pixel is considered to be in edge. The

All rights reserved by *www.ijsrd.com*

computation to detect whether a pixel $T_{i,j}$ at (i,j) is in edge or not is given in equations (1) and (2).

$T_{i,j}$	$T_{i,j+1}$
$T_{i+1,j}$	$T_{i+1,j+1}$

Table. 2: Neighborhood Pixels

$$S_{i,j}=\max\{|T_{i,j}-T_{i+1,j}|,|T_{i,j}-T_{i+1,j+1}|,|T_{i,j}-T_{i,j+1}\}$$

$$E_{i,j}=\begin{cases}1 \text{ if } S_{i,j} > \tau \\ 0 \text{ otherwise}\end{cases}$$

Where τ is a predefined threshold value. Here a threshold value is carefully taken empirically, and the precondition to a pixel $T_{i,j}$ at (i,j) to be in edge is that the value of $S_{i,j}$ must be greater than the threshold, otherwise the pixel is believed to be in non-edge area. This novel method [16] considers only 3-neighbours of a pixel to resolve if the pixel is on edge or not which decreases the computation time.

B. Wavelet encoding

Wavelet encoding is a weighted averaging and differencing scheme. Consider a list of 4 numbers: {6, 8, 14, 10}. By pair wise averaging, we can reduce this to a "second level" list of two numbers, {7, 12}, and a list of second level "differences", {±1, ±2}. Performing the same operation again on the {7, 12} list, we get the third-level list {8.5} and the third level difference list {±1.5}. Observe that only the final singleton list {8.5} and two difference lists of {±1.5} and {±1, ±2} are required to perform the inverse operation and restore the original input list. Using a1D wavelet across both dimensions of the 2D array of pixel data, we reduce the image to a pyramid of images, each level being half the width and height of the previous level and including the corresponding difference list. This is a mathematically lossless process. No significant compression has been achieved, but we have a better representation of the data because it inherently reflects the data at multiple resolution levels. This is similar to the well-known manual image pyramid scheme [17], which has power-of-two sequential reductions in resolution. This reduced resolution decodes fall directly out of the wavelet decomposition, making them easily accessible.

C. Arithmetic Encoding and Bit Planes

The points produced by the wavelet decomposition are grouped into spatially adjacent blocks that can now be encoded independently. Within each block, we consider the data as sets of significant bits: the most significant bit of each piece of data forms the first "bit plane", the second most significant bit of each item forms the second bit plane, and so on. It uses an encoding technique known as arithmetic encoding to efficiently (and lossless) compress these strings of bits. It can be thought of as using WinZip or gzip on a small binary array, although it is more efficient in general. If the user wishes, as part of this process we may choose to remove a number of least significant bits from each point. For example, a block of numbers such as {315, 672, 429, 865} nominally requires twelve (four times three) digits representation. However, if stored as {31_, 67_, 42_, 86_}, where _ represents an omitted digit, only eight (four times two) digits are required, and if stored as {3__, 6__, 4__, 8__}, only four digits are required. Clearly 310 is only an approximation of 315; such is the nature of lossy compression - space is traded off for precision. Again, this description is only to provide some intuition into the actual process but the key properties are the same. Lossless means not throwing away any bits, lossy means we throw away least significant bits. For lossy compression one cannot simply remove arbitrary bits from arbitrary blocks, however, statistics are kept to determine the relative "importance" of each block (based on resolution level and potential error), and thus less important blocks are more likely to lose bits. Data have been processed in blocks, allowing for spatially adjacent points to be stored independently. This gives us our selective decompression property.

Algorithm:

1. Input the image to be compressed.
2. Segment the input image into background and foreground based on outline detection.
3. Perform Wavelet Encoding on foreground image
4. Perform Arithmetic selective bit plane compression on background image
5. Compressed Image is generated

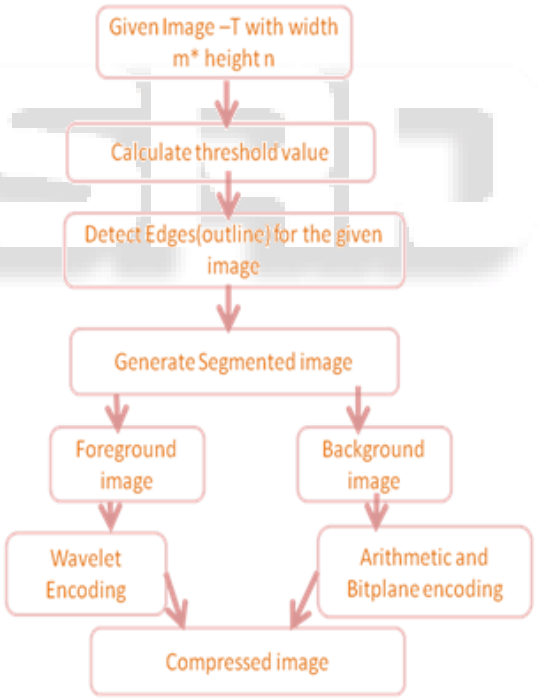

Fig. 2: Work Flow

V. RESULTS

The lena image (Fig 3.a) was taken as an input. The outline detected image (Fig 3.b) was segmented into foreground (Fig 3.d) and background (Fig 3.c) images.

Wavelet encoding was performed on the foreground image (Fig 3.b). Arithmetic selective bit plane compression (fig 4 a to h) was performed on background image based on user preference. The comparison of MSE and PSNR of the images shown below in the table (Table 1) and graph (Fig 5).

All rights reserved by *www.ijsrd.com*

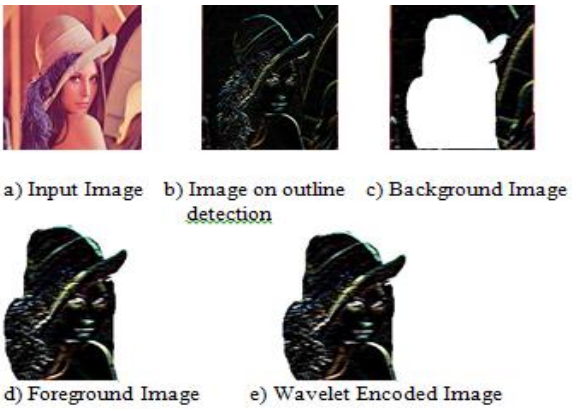

Fig. 3: Processed Images

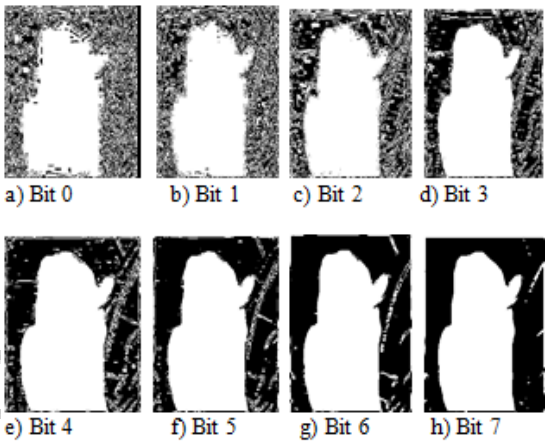

Fig. 4: Arithmetic selective bit plane compressed images

The MSE and PSNR of the images were listed in the Table 2.

Image	MSE	PSNR
Lena-Input image		
Lena after outline detection	6564	9.9591
Foreground image	11204	58.0373
Background image	9000	22.6653
Lena after wavelet encoding	726	19.5214
Arithmetic bit plane encoding		
Bit 0 - 6	2176	14.7542
Bit 7	9644	8.2

Table 2: MSE and PSNR of the images

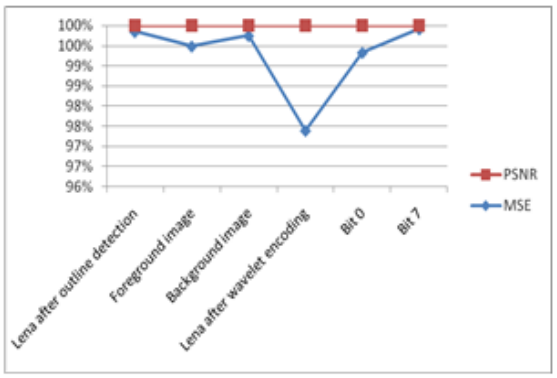

Fig. 5: Comparison of MSE and PSNR of the images

VI. Conclusion

In this paper, an image is compressed based on outline detection, wavelet and bit plane encoding. This paper proposed the user defined lossy compression based on the storage requirements of data, and can adjust the amount of loss the user is willing to accept.

References

[1] Firas A. Jassim and Hind E. Qassim," Five Modulus Method for Image Compression", SIPIJ Vol.3, No.5, pp. 19-28, 2012.

[2] Mridul Kumar Mathur, Seema Loonker and Dr. Dheeraj Saxena "Lossless Huffman Coding Technique For Image Compression And Reconstruction Using Binary Trees",IJCTA, pp. 76-79, 2012.

[3] V.K Padmaja and Dr. B. Chandrasekhar"Literature Review of Image Compression Algorithm", IJSER, Volume 3, pp. 1-6, 2012.

[4] Jagadish H. Pujar and Lohit M. Kadlaskar" A New Lossless Method Of Image Compression and Decompression Using Huffman Coding Techniques", JATIT, pp. 18-22, 2012.

[5] Ashutosh Dwivedi, N Subhash Chandra Bose, Ashiwani Kumar"A Novel Hybrid Image Compression Technique: Wavelet-MFOCPN‖pp. 492-495, 2012.

[6] Yi-Fei Tan and Wooi-Nee Tan" Image Compression Technique Utilizing Reference Points Coding with Threshold Values",IEEE, pp. 74-77, 2012.

[7] S. Sahami and M.G. Shayesteh" Bi-level image compression technique using neural networks", IET Image Process, Vol. 6, Iss. 5, pp. 496–506, 2012.

[8] C. Rengarajaswamy and S. Imaculate Rosaline"SPIHT Compression of Encrypted Images",IEEE, pp. 336-341,2013.

[9] S.Srikanth and Sukadev Meher" Compression Efficiency for Combining Different Embedded Image Compression Techniques with Huffman Encoding",IEEE, pp. 816-820, 2013.

[10] Pralhadrao V Shantagiri and K.N.Saravanan"Pixel Size Reduction Lossless Image Compression Algorithm",IJCSIT, Vol 5, 2013.

[11] K. Rajakumar and T. Arivoli"Implementation of Multiwavelet Transform coding for lossless image compression",IEEE, pp. 634-637, 2013.

[12] S. Dharanidharan , S. B. Manoojkumaar and D. Senthilkumar"Modified International Data Encryption Algorithm using in Image Compression Techniques",IJESIT , pp. 186-191,2013.

[13] Sonal, Dinesh Kumar" A Study of Various Image Compression Techniques",pp. 1-5.

[14] Ming Yang and Nikolaos Bourbakis" An Overview of Lossless Digital Image Compression Techniques",IEEE, pp. 1099-1102,2005.

[15] Tzong Jer Chen and Keh-Shih Chuang" A Pseudo Lossless Image Compression Method",IEEE, pp. 610-615, 2010.

[16] M.Mohamed Sathik , K.Senthamarai Kannan and Y.Jacob Vetha Raj "Hybrid Jpeg Compression Using Edge Based Segmentation" Signal & Image Processing : An International Journal(SIPIJ) Vol.2, No.1, March 2011

[17] A white paper, LizardTech introduces you to the concept of compression, MrSID technology, and the features that the MrSID format brings to your applications and workflows. Copyright © 2010 LizardTech, a Celartem company.

[18] Gaurav Vijayvargiya Dr. Sanjay Silakari Dr.Rajeev Pandey "A Survey: Various Techniques of Image Compression", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 11, No. 10, October 2013.

All rights reserved by *www.ijsrd.com*