



ISSN NO. 2320-5407

Journal homepage: <http://www.journalijar.com>Journal DOI: [10.21474/IJAR01](https://doi.org/10.21474/IJAR01)

**INTERNATIONAL JOURNAL
OF ADVANCED RESEARCH**

RESEARCH ARTICLE

An approach for reducing the redundancy of image using Discrete Wavelet Transform technique.**Manasa K Chigateri¹, Manjuvani K M¹, Manjunath K M¹ and Khaja Moinuddin².**

1. Asst. Professor ECE department, RYMEC, Bellary, Karnataka, India.
2. Assoc. Professor ECE department, RYMEC, Bellary, Karnataka, India.

Manuscript Info**Manuscript History:**

Received: 14 April 2016
 Final Accepted: 19 May 2016
 Published Online: June 2016

Key words:

SNR, wavelet coefficient, YUV
 Space, CST curve, 2D-wavelet.

Corresponding Author*Manasa K Chigateri.****Abstract**

Data compression is the process of converting data files into smaller files for efficiency of storage and transmission. As one of the enabling technologies of the multimedia revolution, data compression is a key to rapid progress being made in information technology. It would not be practical to put images, audio, and video alone on websites without compression. Here, we using image data compression using 2D-wavelet transform analysis. It carried out in two methods namely: Global Thresholding and Level-Dependent Threshold. In this analysis, we carried out the process by using level-dependent thresholding, the density of the wavelet decomposition was reduced by 3% while improving the L2-norm recovery by 3%. The EZW coding algorithm was implemented to achieve better compression results

*Copy Right, IJAR, 2016., All rights reserved.***Introduction:-**

Digital images are commonly used in computer applications. Uncompressed digital images prerequisite considerable storage capacity and transmission bandwidth. Efficient image compression solutions are becoming more critical with the recent growth of data intensive, multimedia based web applications. Data compression algorithms are used in those standards to reduce the number of bits required to represent an image or a video sequence. Compression is the process of representing information in a compact form. Data compression treats information in digital form as binary numbers represented by bytes of data with very large data sets. Compression is a necessary and essential method for creating image files with manageable and transmittable sizes. In order to be useful, compression algorithm has a corresponding decompression algorithm that, given the compressed file, reproduces the original file. There have been many types of compression algorithms developed. These algorithms fall into two broad types, lossless algorithms and lossy algorithms. A lossless algorithm reproduces the original exactly. A lossy algorithm, as its name implies, loses some data. Data loss may be unacceptable in many applications. Depending on the quality required of the reconstructed image, varying amounts of loss of information can be accepted.

Image compression:-

Reduce the redundancy of the image data in order to be able to store or transmit data in an efficient form.

**Ex: 3504X2336 (full color) image: $3504 \times 2336 \times 24/8 = 24,556,032$ Byte
 = 23.418 Mbyte**

For human eyes, the image will still seem to be the same even when the Compression ratio is equal 10. Human eyes are less sensitive to those high frequency signals. Our eyes will average fine details within the small area and record only the overall intensity of the area, which is regarded as a low pass filter. In addition, introduces the notion that the Contrast Sensitivity Threshold (CST) curve, which gives the relationship between contrast and frequency at the limits of human perception, should be used to choose the norm in which to measure the error in compressed images and to choose the quantization strategy.

Wavelets:-

Wavelet transforms encode more information than other techniques like Fourier transforms. Time and frequency information is saved. In practical terms, the transformation is applied to many scales and sizes within the signal, these results in vectors that encode approximation and detail information. By separating the signals, it is easier to threshold and removes information. Thus the data can be compressed. The wavelet transform (WT) has gained widespread acceptance in signal processing and image compression. Because of their inherent multi-resolution nature, wavelet-coding schemes are especially suitable for applications where *scalability* and *tolerable degradation* are important.

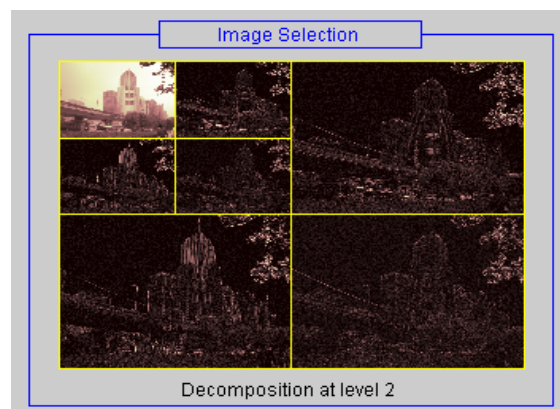


Figure (1): Image decomposed at level 2.

The jpeg2000 standard gave up the discrete cosine transform in favor of a wavelet transform. The FBI uses wavelets to compress fingerprint scans by 15 – 20 times the original size. Recently the JPEG committee has released its new image coding standard, JPEG-2000, which has been based upon DWT.

- **Decompose:** Choose a wavelet, choose a level N. Compute the wavelet decomposition of the signal at level N.
- **Threshold detail coefficients:** For each level from 1 to N, a threshold is selected and hard thresholding is applied to the detail coefficients.
- **Reconstruct:** Compute wavelet reconstruction using the original approximation coefficients of level N and the modified detail coefficients of levels from 1 to N.

Discrete Wavelet Transform:-

The wavelet transform is computed separately for different segments of the time-domain signal at different frequencies.

- Multi-resolution analysis: analyzes the signal at different frequencies giving different resolutions
- MRA is designed to give good time resolution and poor frequency resolution at high frequencies and good frequency resolution and poor time resolution at low frequencies
- Good for signal having high frequency components for short durations and low frequency components for long duration. E.g. images and video frames.

Wavelet transform decomposes a signal into a set of basis functions. These basis functions are called *wavelets*. Wavelets are obtained from a single prototype wavelet $\psi(t)$ called mother *wavelet* by *dilations* and *shifting*:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right)$$

where a is the scaling parameter and b is the shifting parameter.

Existing methods:-

The key step in lossy data compression in which data cannot be recovered exactly is the quantization phase, which exploits a data reduction based on their low information content. This is not optimal for RGB images. Nevertheless, for three layers, this can lead to the elimination of some low coefficients in a channel in a certain spatial location, even though the corresponding coefficients in the other layers are not eliminated because they carry high information content. When reconstructing the image at that location, a high visual distortion is introduced. The

assumption of analyzing the three layers separately is valid only if they are not correlated with respect the visual appearance. [12].

RGB space:-

RGB is perhaps the simplest and most commonly used color space (see Figure 2). It uses proportions of red, green, and blue those are scaled to a minimum and maximum value for each component (for example, 0x00 through 0xFF, or 0.0 through 1.0). Most colors in the visible spectrum can be recreated, although not completely. This scheme is based on the additive properties of color. [12]

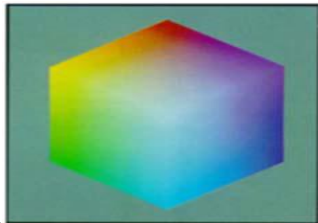


Figure (2): RGB Space [12].

YUV Space:-

YUV was originally used for PAL (European standard) analog video (see Figure3). To convert from RGB to YUV spaces, the following equations can be used:

$$Y = 0.299 R + 0.587 G + 0.114 B$$

$$U = 0.492 (B - Y)$$

$$V = 0.877 (R - Y)$$

Any errors in the resolution of the luminance (Y) are more important than the errors in the chrominance (U, V) values. The luminance information can be coded using higher bandwidth than the chrominance information [12].

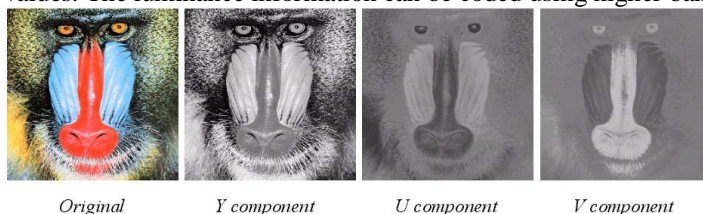


Figure (3).Example of YUV Space [12].

Compression Techniques:-

Compression takes an input X and generates a representation XC that hopefully requires fewer bits. There is a reconstruction algorithm that operates on the compressed representation XC to generate the reconstruction Y. Based on the requirements of reconstruction, data compression schemes can be divided into two broad classes. One is lossless compression, in which Y is identical to X. Examples of lossless methods are Run

1. Run Length coding,
2. Huffman coding,
3. Lempel/Ziv algorithms, and
4. Arithmetic coding.

The other is lossy compression, which generally provides much higher compression than lossless compression but allows Y to be different from X.

Lossless Compression:-

If data have been losslessly compressed, the original data can be recovered exactly from the compressed data. It is generally used for applications that cannot allow any difference between the original and reconstructed data.

Run Length Encoding:-

Run length encoding, sometimes called recurrence coding, is one of the simplest data compression algorithms. It is effective for data sets that are comprised of long sequences of a single repeated character. For instance, text files with large runs of spaces or tabs may compress well with this algorithm.

RLE finds runs of repeated characters in the input stream and replaces them with a three-byte code. The code consists of a flag character, a count byte, and the repeated characters. For instance, the string "AAAAAABBBBCCCC" could be more efficiently represented as "A6*B4*C5". That saves us six bytes.

Huffman Coding:-

Huffman coding, developed by D.A. Huffman [3], is a classical data compression technique. It has been used in various compression applications, including image compression. It uses the statistical property of characters in the source stream and then produces respective codes for these characters. These codes are of variable code length using an integral number of bits. The codes for characters having a higher frequency of occurrence are shorter than those codes for characters having lower frequency. This simple idea causes a reduction in the average code length, and thus the overall size of compressed data is smaller than the original. Huffman coding is based on building a *binary tree* that holds all characters in the source at its *leaf nodes*, and with their corresponding characters' probabilities at the side.

Lempel-Ziv-Welch (LZW):-

Encoding. This original approach is given by J. Ziv and A. Lempel in 1977 [17]. T. Welch's refinements to the algorithm were published in 1984 [18]. LZW compression replaces strings of characters with single codes. It does not do any analysis of the incoming text. Instead, it just adds every new string of characters it sees to a table of strings. Compression occurs when a single code is output instead of a string of characters.

The code that the LZW algorithm outputs can be of any arbitrary length, but it must have more bits in it than a single character. The first 256 codes (when using eight bit characters) are by default assigned to the standard character set. The remaining codes are assigned to strings as the algorithm proceeds. There are three best-known applications of LZW: UNIX compress (file compression), GIF (image compression), and V.42 bis (compression over Modems).

Arithmetic Coding:-

Arithmetic coding is also a kind of statistical coding algorithm similar to Huffman coding. However, it uses a different approach to utilize symbol probabilities, and performs better than Huffman coding. In Huffman coding, optimal codeword length is when the symbol probabilities are of the form $(1/2)^x$, where x is an integer. This is because Huffman coding assigns code with an integral number of bits. This form of symbol probabilities is rare in practice. Arithmetic coding is a statistical coding method that solves this problem. The code form is not restricted to an integral number of bits. It can assign a code as a fraction of a bit.

Lossy Compression Methods:-

Lossy compression techniques involve some loss of information and data cannot be recovered or reconstructed exactly. In some applications, exact reconstruction is not necessary. For example, it is acceptable that a reconstructed video signal is different from the original as long as the differences do not result in annoying artifacts. However, generally obtain higher compression ratios than is possible with lossless compression.

System architecture:-

Transform Based Image Compression. The basic encoding method for transform based compression works as follows:

Image transform: Divide the source image into blocks and apply the transformations to the blocks.

Parameter quantization: The data generated by the transformation are quantized to reduce the amount of information. This step represents the information within the new domain by reducing the amount of data. Quantization is in most cases not a reversible operation because of its lossy property.

Encoding: Encode the results of the quantization. This last step can be error free by using Run Length encoding or Huffman coding. It can also be lossy if it optimizes the representation of the information to further reduce the bit rate.

Transform based compression is one of the most useful applications. Combined with other compression techniques, this technique allows the efficient transmission, storage, and display of images that otherwise would be impractical.

Wavelet Transform: Wavelets are functions defined over a finite interval. The basic idea of the wavelet transform is to represent an arbitrary function $f(x)$ as a linear combination of a set of such wavelets or basis functions. These basis functions are obtained from a single prototype wavelet called the mother wavelet by dilations (scaling) and translations (shifts). The purpose of wavelet transform is to change the data from time-space domain to time-frequency domain which makes better compression results.

The simplest form of wavelets, the Haar wavelet function (see Figure 5) is defined as:

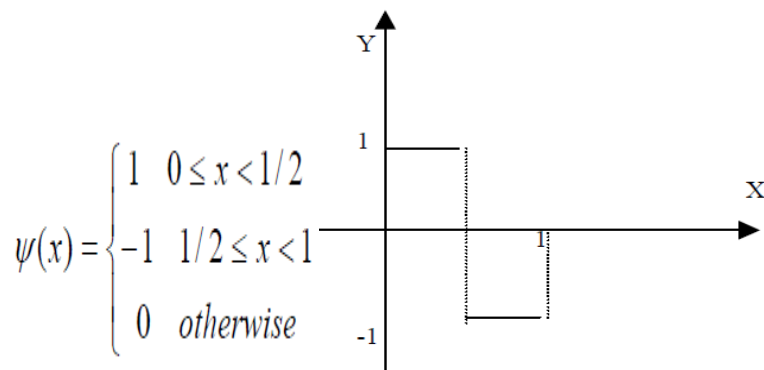


Figure (5): Haar wavelet.

Discrete Wavelet Transform: Advantage over CWT: reduce the computational complexity (separate into H & L freq.). Inner product of $f(t)$ and discrete parameters a & b

$$a = a_0^{-m}, \quad b = nb_0 a_0^{-m} \quad m, n \in \mathbb{Z}$$

If $a_0=2, b_0=1$, the set of the wavelet

$$\psi_{m,n}(t) = a_0^{m/2} \psi(a_0^m t - nb_0) \quad m, n \in \mathbb{Z}$$

The DWT

$$\psi_{m,n}(t) = 2^{m/2} \psi(2^m t - n)$$

$$w_{m,n} = \langle f(t), \psi_{m,n}(t) \rangle = a_0^{m/2} \int f(t) \psi(a_0^m t - nb_0) dt$$

$$f(t) = \sum_m \sum_n w_{m,n} \psi_{m,n}(t)$$

Discrete Wavelet Transform (DWT) which transforms a discrete time signal to a discrete wavelet representation. It converts an input series x_0, x_1, \dots, x_m into one high-pass wavelet coefficient series and one low-pass wavelet coefficient series (of length $n/2$ each).

Method 1: Global Thresholding:-

The compression features of a given wavelet basis are primarily linked to the relative scarceness of the wavelet domain representation for the signal. The notion behind compression is based on the concept that the regular signal component can be accurately approximated using the following elements: a small number of approximation coefficients (at a suitably chosen level) and some of the detail coefficients. Global thresholding is performed to convert the entropy image into binary image. Global thresholding, using an appropriate threshold T :

$$g(x, y) = 1, \text{ if } f(x, y) > T$$

$$0, \text{ if } f(x, y) \leq T$$

Method 2: Level-Dependent Thresholding:-

The WDENCMP function also allows level and orientation-dependent thresholds. In this case the approximation is kept. The level-dependent thresholds in the three orientations horizontal, diagonal and vertical are as follows:

opt = 'lvd'; % Level dependent thresholds

thr_h = [17 18]; % Horizontal thresholds.

thr_d = [19 20]; % Diagonal thresholds.

thr_v = [21 22]; % Vertical thresholds.

$\text{thr} = [\text{thr_h} ; \text{thr_d} ; \text{thr_v}] ;$

In this second example, notice that the WDENCMP function performs a compression process from the image x.

Block diagram description:-

It consists of three steps as shown below.

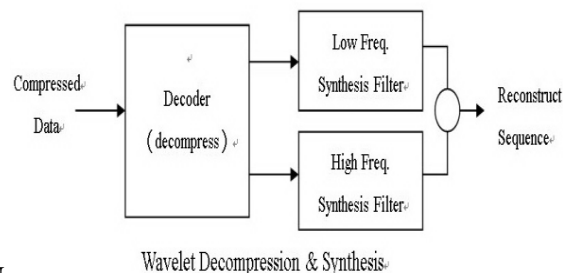
Decompose: Choose a wavelet, choose a level N. Compute the wavelet decomposition of the signal at level N.

Threshold detail coefficients:-

For each level from 1 to N, a threshold is selected and hard thresholding is applied to the detail coefficients.

Reconstruct:-

Compute wavelet reconstruction using the original approximation coefficients of level N and the modified detail



coefficients of levels from 1 to N.

Figure (7): Block diagram of wavelet compression.

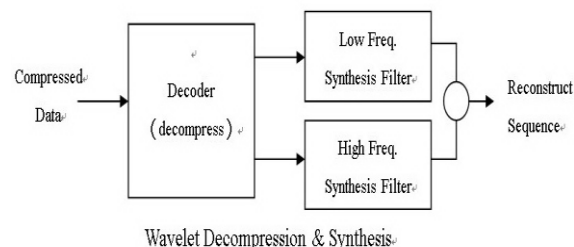


Figure (8): Block diagram of wavelet decomposition.

Dimension (analysis):-

2-D wavelets are analyzed by below figure(9), it consists of 2 rows and 4 columns with the Haar transform to perform 2-d image compression using wavelets. Input of $x(m, n)$ is passed where m represents rows and n represents columns to Haar transformation to obtain compressed image, later it can be implemented.

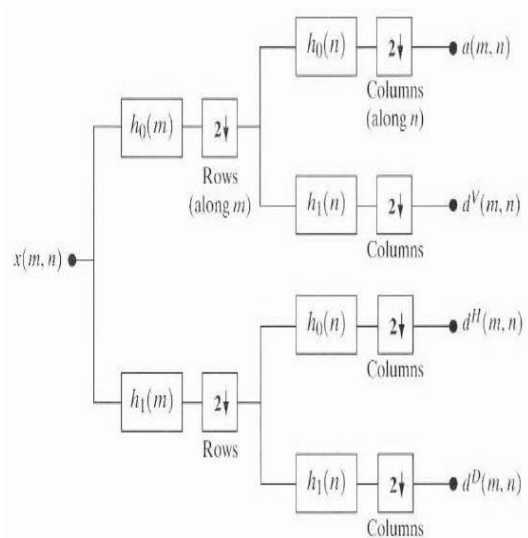
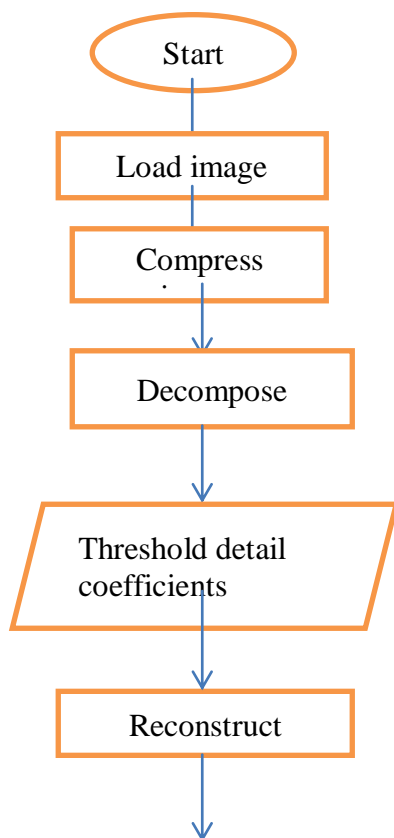
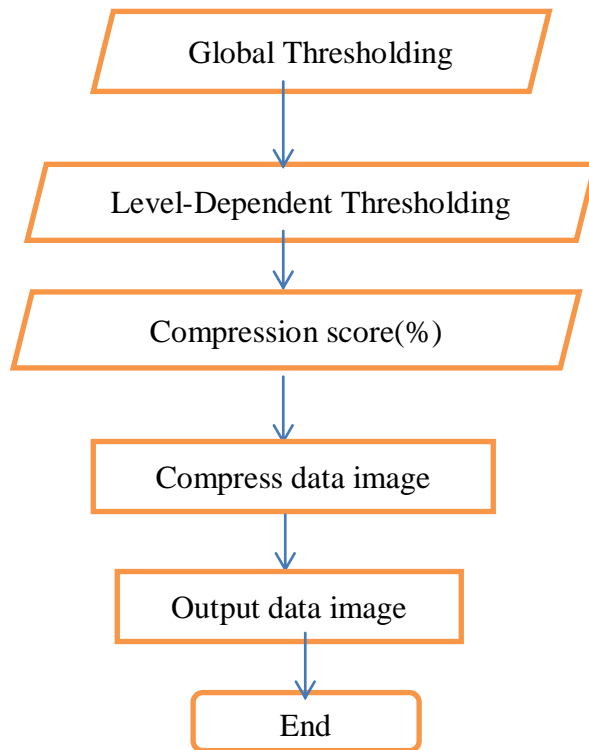


Figure (9): 2-D wavelet analysis

One area where wavelets have incontestably proven their applicability is image processing. As you know high resolution images claim a lot of disk space. In the age of information highway, the amount of data that needs to be stored or transmitted is huge. Therefore, compression greatly increases the capacity of the storage device, on one hand, and on the other, it also reduces costs.

To illustrate the use of compression take the simplest example: an image of 256×256 , which takes approximately 0.2 MB. On a simple floppy disk one can therefore store 7 such images. But think if this image can be compressed at a 25:1 ratio. The result is 175 images stored on the same floppy disk. Using the "Lenna" image for illustrations. Finally we will present the MATLAB code for output.





Figure(10): Flow Chart Of program procedure.

Implementation:-

A natural way to determine the fidelity of a recovered image is to find the difference between the original and reconstructed values. Two popular measures of distortion are the squared error measure and the absolute measure, which are called *difference distortion measures*. If $\{X_n\}$ is the source output and $\{Y_n\}$ is the reconstructed sequence, the squared error measure is given by

$$d(x, y) = (x - y)^2$$

and the absolute difference measure is given by

$$d(x, y) = |x - y|.$$

Practically, it is difficult to examine the difference on a term-by-term basis. Some average measures are used to summarize the information. The most often used average measure is the average of squared error measure.

This is called the *mean squared error* (MSE) and is often denoted by the symbol

$$\delta_d^2 = \frac{1}{N} \sum_{n=1}^N (x_n - y_n)^2.$$

If we are interested in the size of the error relative to the signal, we can find the ratio of the average squared value of the source output and the MSE. This is called the *signal-to-noise-ratio* (SNR).

$$SNR = \frac{\delta_x^2}{\delta_d^2}$$

Sometimes we are more interested in the size of the error relative to the peak value of the signal *peak x* than the size of the error relative to the average squared value of the signal. This ratio is called the *peak-signal-to-noise ratio* (PSNR) and is calculated by the following equation:

$$PSNR(dB) = 10 \log_{10} \frac{x_{peak}^2}{\delta_d^2}.$$

PSNR is the most commonly used value to evaluate the objective image compression quality.

Results and discussion:-

Output of data image compression using wavelet transform is shown below figure (14). It consist of original image, compressed image using global thresholding and level-dependent thresholding techniques.

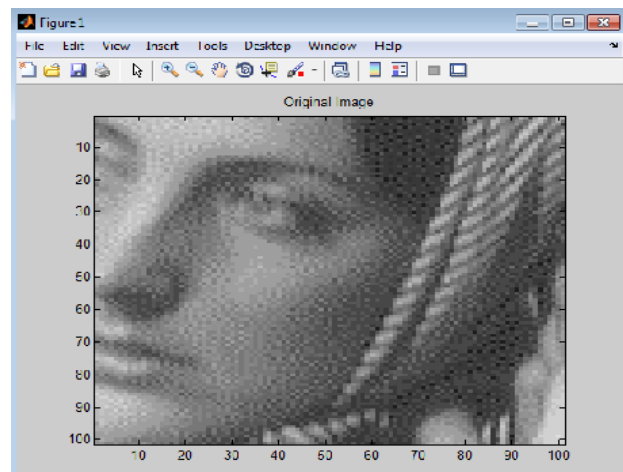
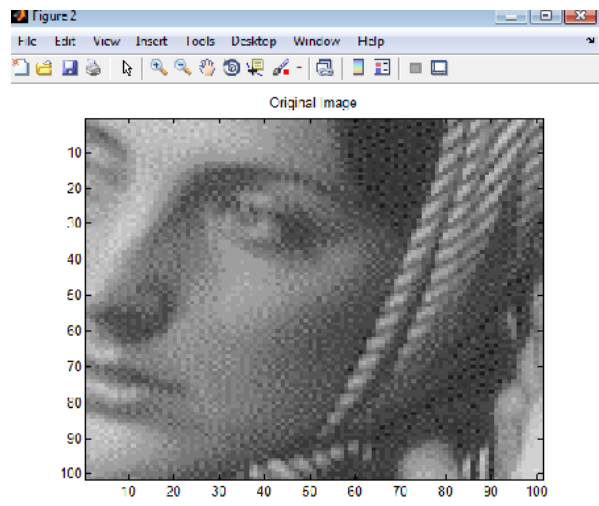


Figure (15): original images

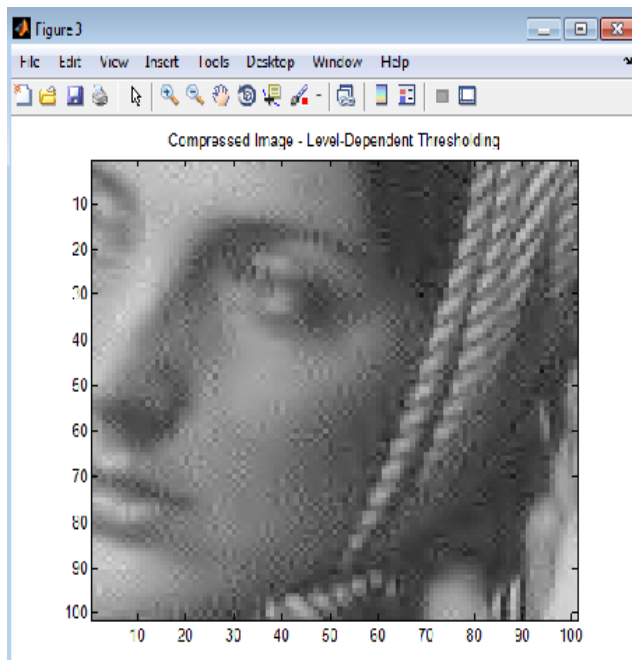


Figure (16): compressed output images

Conclusions:-

The data compression schemes can be divided into two classes. One is lossless compression and the other is lossy compression. Lossy compression generally provides much higher compression than lossless compression. Wavelets are a class of functions used to localize a given signal in both space and scaling domains. In order to compare wavelet methods, a MinImage was originally created to test one type of wavelet and the additional functionality was added to Image to support other wavelet types.

Scope of enhancement:-

Like images compression, we can implement video, audio, speech recognition using 1-D, 2-D wavelet transform. Wavelet transforms encode more information than other techniques like Fourier transforms. DWT has been implemented in hardware such as ASIC and FPGA. Improved low bit-rate compression performance, lossless and lossy compression, continuous-tone and bi-level compression. Be able to compress large images use single decompression architecture.

References:-

1. C.Gargour, M. Gabrea, V. Ramachandran, J.M. Lina, "A short introduction to wavelets and their applications," *Circuits and Systems Magazine, IEEE*, Vol. 9, No. 2, (05 June 2009), pp. 57-68.
2. Nancy A. Breau and Chee-Hung Henry Chu, "Wavelet methods for compression, rendering, and descreening in digital halftoning," *SPIE proceedings series*, vol. 3078, pp. 656-667, 1997.
3. M. Barlaud et al., "Image Coding Using Wavelet Transform" *IEEE Trans. on Image Processing* 1, No. 2, 205-220 (April, 1992).
4. [http://www.mathworks.com/access/helpdesk/help/toolbox/wavelet/wavelet.html:matlab:wavelet toolbox](http://www.mathworks.com/access/helpdesk/help/toolbox/wavelet/wavelet.html:matlab:wavelet%20toolbox).
5. K. Sayood, *Introduction to Data Compression*, Second edition, Academic Press, 2000.
6. Fournier, "Wavelets and their Applications in Computer Graphics", University of British Columbia, SIGGRAPH'95 Course Notes, 1995.
7. C.S. Burrus, R.A. Gopinath, and H. Guo, *Introduction to Wavelets and Wavelet Transforms*, Englewood Cliffs, NJ: Prentice Hall, 1998.
8. S. Saha, "Image Compression - from DCT to Wavelets"
9. <http://www.acm.org/crossroads/xrds6-3/sahaimgcoding.html>