# Image Compression Using Discrete Wavelet Transform

[I]Niti, [II]Abhishek Gupta
[I]B.Tech (HCE. Sonepat), M. Tech. (GIET, Sonepat), Haryana, India
[II]Assistant Professor, (GIET Sonepat), Haryana, India

**Abstract**

*Image compression is a method through which we can reduce the storage space of images, videos which will helpful to increase storage and transmission process's performance. In image compression, we do not only concentrate on reducing size but also concentrate on doing it without losing quality and information of image. In this paper, we present the comparison of the performance of discrete cosine transform, discrete wavelet transform and wavelets like Haar Wavelet and Daubechies Wavelet for implementation in a still image compression system and to highlight the benefit of these transforms relating to today's methods. The performance of these transforms are compared in terms of Signal to noise ratio SNR, Mean squared error (MSE) and Energy Retained (ER) etc.*

**Keywords**

*DWT, SNR, MSE, ER, Image Compression*

## I. Introduction to Image Compression

Image compression is an application of data compression that encodes the original image with few bits. The objective of image compression is to reduce the redundancy of the image and to store or transmit data in an efficient form. The main goal of such system is to reduce the storage quantity as much as possible, and the decoded image displayed in the monitor can be similar to the original image as much as can be. For example,

Someone with a web page or online catalog that uses dozens or perhaps hundreds of images will more than likely need to use some form of image compression to store those images. There are several methods of image compression available today. This fall into two general categories: lossless and lossy image compression with lossless compression every single bit of data that was originally in the file remains after the file is uncompressed. All of the information is completely restored. The Graphics Interchange File (GIF) is an image format used on the Web that provides lossless compression. On the other hand, lossy compression reduces a file by permanently

eliminating certain information, especially redundant information. When the file is uncompressed, only a part of the original information is still there (although the user may not notice it). The JPEG image file, commonly used for photographs and other complex still images on the Web, is an image that has lossy compression.

## II. The Flow of Image Compression Coding

Image compression coding is to store the image into bit-stream as compact as possible and to display the decoded image in the monitor as exact as possible. When the encoder receives the original image file, the image file will be converted into a series of binary data, which is called the bit-stream. The decoder then receives the encoded bit-stream and decodes it to form the decoded image. If the total data quantity of the bit-stream is less than the total data quantity of the original image, then this is called image compression. The full compression flow is as shown

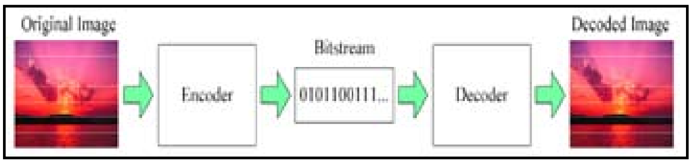



Fig. 2.1 : The basic flow of image compression coding.

The compression ratio is defined as follows:

$$Cr = n1/n2, \quad (1.1)$$

Where n1 is the data rate of original image and n2 is that of the encoded bit-stream. In order to evaluate the performance of the image compression coding, it is necessary to define a measurement that can estimate the difference between the original image and the decoded image. Two common used measurements are the Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR),f(x, y) is the pixel value of the original image, and f'(x, y)is the pixel value of the decoded image Most image compression systems are designed to minimize the MSE and maximize the PSNR.

$$MSE = \sqrt{\frac{\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}\left[f(x,y)-f'(x,y)\right]^2}{WH}}$$

$$PSNR = 20\log_{10}\frac{255}{MSE}$$

## III. Discrete Wavelet Transform

In wavelet analysis, the Discrete Wavelet Transform (DWT) decomposes a signal into a set of mutually orthogonal wavelet basis functions. These functions differ from sinusoidal basis functions in that they are spatially localized – that is, nonzero over only part of the total signal length. Furthermore, wavelet functions are dilated, Translated and scaled versions of a common function φ, known as the mother wavelet. As is the case in Fourier analysis, the DWT is invertible, so that the original signal can be completely recovered from its DWT representation. Unlike the DFT, the DWT, in fact, refers not just to a single transform, but rather a set of transforms, each with a different set of wavelet basis functions. Two of the most common are the Haar wavelets and the Daubechies set of wavelets. Following are the important properties of DWT:

1. Wavelet functions are spatially localized;
2. Wavelet functions are dilated, translated and scaled versions of a common mother wavelet; and
3. Each set of wavelet functions forms an orthogonal set of basic functions.

Wavelet Transform of an image is a representation across multiple scales wherein the sharp variations are enhanced at finer scales wherein the sharp variations are enhanced at finer scales and slow

ISSN : 2394-2975 (Online)
ISSN : 2394-6814 (Print)
Vol. 2, Issue 4 (Oct. - Dec. 2015)
International Journal of Advanced Research in Education & Technology (IJARET)

variations and background are distinctly visible at coarser scales. The DTWT splits or decomposes the given signal into components of different scales. Thus, the grounding of wavelet transforms in multiscale decomposition would seem to provide a justification for its use in image compression and analysis in particular, our perception of edges, a crucial image feature, appears to be based on their detectability at multiple scales. Most of the wavelet coefficients in the details have very low values, consequently most of these can be quantized to zero without affecting the perceived quality of the reconstructed image significantly. All wavelet based image compression techniques take advantage of this phenomenon.

### A. Haar Wavelet

A Haar wavelet is the simplest type of wavelet. In discrete form, Haar wavelets are related to a mathematical operation called the Haar transform. The Haar transform serves as a prototype for all other wavelet transforms. Like all wavelet transforms, the Haar transform decomposes a discrete signal into two sub-signals of half its length. One sub-signal is a running average or trend; the other sub-signal is a running difference or fluctuation.

The Haar wavelet's mother wavelet function $\psi(t)$can be described as

$$\psi(t) = \begin{cases} 1 & 0 \le t < \frac{1}{2}, \\ -1 & \frac{1}{2} \le t < 1, \\ 0 & \text{otherwise.} \end{cases}$$

Its scalling function $\phi(t)$can be described as

$$\phi(t) = \begin{cases} 1 & 0 \le t < 1, \\ 0 & \text{otherwise.} \end{cases}$$

The Haar wavelet transform has a number of advantages:

- It is conceptually simple.
- It is fast.
- It is memory efficient, since it can be calculated in place without a temporary Array.
- It is exactly reversible without the edge effects that are a problem with other Wavelet transforms.

The Haar transform also has limitations which can be a problem with for some applications. In generating each of averages for the next level and each set of coefficients, the Haar transform performs an average and difference on a pair of values. Then the algorithm shifts over by two values and calculates another average and difference on the next pair. The high frequency coefficient spectrum should reflect all high frequency changes. The Haar window is only two elements wide. If a big change takes place from an even value to an odd value, the change will not be reflected in the high frequency coefficients. The audio de-noising by Haar wavelet is not always so effective, because the transform can't compress the energy of the original signal into a few high-energy values lying above the noise threshold. Also if we try to use the Haar wavelet for threshold compression of audio signal, we get poor results. So Haar wavelet transform is not useful in compression and noise removal of audio signal processing. Haar 2-tap wavelet can be used to perform the Haar wavelet transforms.

### B. Daubechies wavelet transform

The Daubechies wavelet transforms are defined in the same way as the Haar wavelet transform by computing the running averages and differences via scalar products with scaling signals and wavelets the only difference between them consists in how these scaling signals and wavelets are defined. The Daubechies wavelet is more complicated than the Haar wavelet. Daubechies wavelets are continuous; thus, they are more computationally expensive to use than the Haar wavelet, this wavelet type has balanced frequency responses but non-linear phase responses. Daubechies wavelets use overlapping windows, so the high frequency coefficient spectrum reflects all high frequency changes. *Audio de-noising and compression is more sonically pleasing with the Daubechies wavelet than with the Haar wavelet.* The Daubechies 4 filter can be used to perform the Daubechies wavelet transforms. The level 2 Daubechies Daub4 wavelet transform is $x \rightarrow (b_1 \mid b_2 \mid a_2)$, where $a_2$ and $b_2$ are the level 1 Daubechies Daub4 wavelet transform of $a_1$, computed with vectors A and B of length N / 2. The level N Daubechies Daub4 wavelet transform is computed by successively taking the transforms of the "a" portion of the previous level of the transform, with vectors that are of length equal to half of the length of the previous transform.

Note that x(k) can be reconstructed from $a_1$ and $b_1$ with the computation

$$x(k) = \sum_{i=0}^{N-1} A_i(k)\, a_1(i) + \sum_{i=0}^{N-1} B_i(k)\, b_1(i)$$

In the level 2 transform, a1 can be reconstructed from $a_2$ and $b_2$ and thus x(k) can be reconstructed from $b_1$, $b_2$, and $a_2$ (hence, $a_1$ is omitted from the definition of the transform $x \rightarrow (b_1 \mid b_2 \mid a_2)$).

In each successive iteration, the "a" and "b" vectors are of length equal to half of the length of the previous level transform. Thus, if $N = 2^n$ for some integer n, we can write $x \rightarrow (b_1 \mid b_2 \mid \ldots \mid b_{N-1} \mid a_{N-1})$, where $a_{N-1}$ and $b_{N-1}$ are vectors of length 2 (unlike when using the Haar Wavelet Transform we cannot get to level N and vectors $a_N$ and $b_N$ of length 1, as the A and B vectors have four nonzero values rather than just two). Note that $b_1$ is of length N / 2, $b_2$ is of length N / 4 and so on. The combined length of $b_1$, $b_2$, … , $b_{N-1}$, $a_{N-1}$ is N and so these vectors carry the same amount of information as x(k). Since x(k) can be reconstructed from these vectors, they carry the same information as x(k).

As in the Haar wavelet transform, $a_1$ averages the signal x(k) and is the "trend" of the signal. $b_1$ contains the fluctuations that $a_1$ removes. $a_1$ is computed as the convolution of the signal x(k) with $A_1$ (with the caveat that the step of this convolution is two samples, rather than one). $b_1$ is the convolution of the signal x(k) with $B_1$. In other words, $a_1$ and $b_1$ are the results of a low pass filter and a high pass filter on x(k) (the vectors $A_1$ and $B_1$). Similarly, $a_2$ and $b_2$ are the trend and fluctuations of $a_1$ and so on.The Daub4 wavelet transform is only one of the Daubechies wavelet transforms.

### 1. Data compression example with the Daubechies Daub4 wavelet transform

Take the signal x(k) of length N = 256 that consists of two simple waves with frequencies 5 Hz and 9 Hz, given the sampling frequency 256 Hz. The phase of the 9 Hz frequency is 10 samples and the phase of the 5 Hz frequency is zero. The signal is

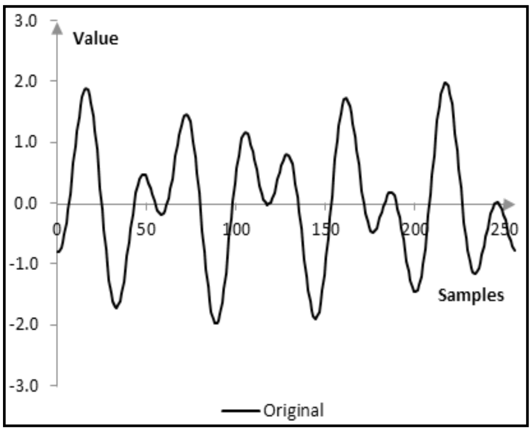
ISSN : 2394-2975 (Online)
ISSN : 2394-6814 (Print)

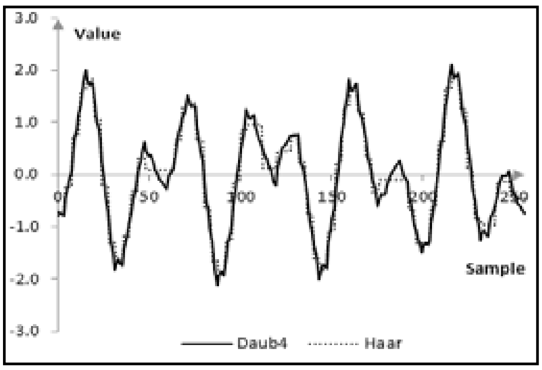
$$x(k) = \sin\left(\frac{2\pi k5}{256}\right) + \sin\left(\frac{2\pi(k-10)9}{256}\right)$$

A detailed example of compressing this signal is provided in the topic Haar Wavelet Transform. This example here compares the results of the Haar and Daubechies Daub4 wavelet transforms. Decompose this signal using the Daubechies Daub4 wavelet transform into $b_1$, $b_2$, …, $b_7$, and $a_7$. Sort all values of these vectors (as one single array) and remove the smallest values (replace them with 0). If we need to transmit the signal, we can transmit only the nonzero values. We can use bitmasks to denote where the nonzero values should be placed when the $b_1$, …, $a_7$ vectors are reconstructed and we can then reconstruct the signal x(k) from these vectors.

This will result in transmitting less data, than the signal x(k) itself. In the example described in the topic Haar Wavelet Transform, for example, removing half of all values (the smaller values) allowed the transform to retain 99.5% of the energy of the signal, where the energy is computed as the sum of the squares of the values of the transform. Since half of the values were zeroed out, the compression ratio in that example was 2:1. In the Daubechies Daub4 wavelet transform of this same signal, 99.5% of the energy can be retained if only a fifth of the values are retained and four fifths are zeroed out. That is, the same amount of energy can be retained with 5:1 compression. If 5:1 compression is used in the Haar wavelet transform, the loss of energy will be over 5%, which is significant.

The reconstructed signal x(k), after a number of values are removed from the vectors $b_1$, …, $a_7$, is not the same as the original, of course. The following graphs show the two signals.




## IV. Proposed Compression Method using DWT

This section illustrates the proposed compression technique with pruning proposal based on discrete wavelet transform (DWT). The proposed technique first decomposes an image into coefficients called sub-bands and then the resulting coefficients are compared with a threshold. Coefficients below the threshold are set to zero. Finally, the coefficients above the threshold value are encoded with a loss less compression technique. 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.

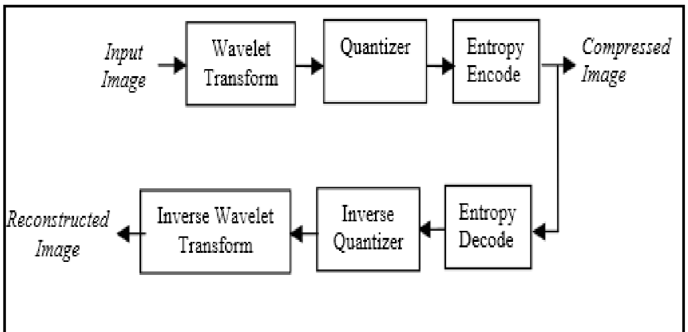



Fig. 2 The structure of the wavelet transform based compression.

The steps of the proposed compression algorithm based on DWT are described below:

### I. Decompose

Choose a wavelet; choose a level N. Compute the wavelet. Decompose the signals at level N.

### II. Threshold detail coefficients

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

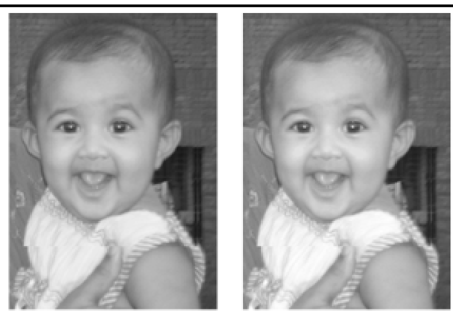
### III. Reconstruct

Compute wavelet reconstruction using the original approximation coefficients of level N and the modified detail coefficients of levels from 1 to N.

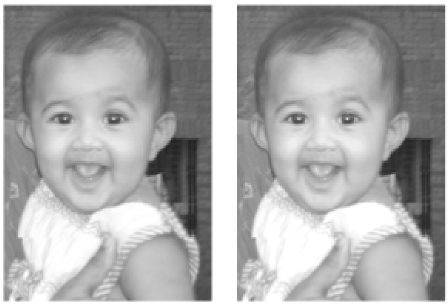
## V. Experimental Results and Discussion

In this research, an efficient compression technique based on discrete wavelet transform (DWT) is proposed and developed. The algorithm has been implemented using Visual C++. A set of test images (bmp format) are taken to justify the effectiveness of the algorithm.

Fig. 3 shows a test image and resulting compressed images using JPEG, GIF and the proposed compression methods.



a) Original image (bmp) (Resolution: 140x190)  (b) Compressed image using GIF



(c) Compressed image using JPEG (d) Compressed image using DWT

**Fig.3** A real image and corresponding compressed images with GIF, JPEG and proposed DWT methods.

Table shows the comparison between JPEG, GIF and the proposed compression method. Experimental results demonstrate that the proposed compression technique gives better performance compared to other compression techniques.

Table 1 : Comparison between DWT and Other Method.

| Compression Techniques | File sizes | Compression Ratio | PSNR (dB) |
|---|---|---|---|
| Original Image(Bmp) | 47.00 KB | - | - |
| GIF | 6.40 KB | 7.34:1 | 27.37 |
| JPEG | 3.38 KB | 13.90:1 | 24.42 |
| DWT (Proposed Method) | 1.94 KB | 24.22:1 | 19.86 |

## VI. Conclusions

A new image compression scheme based on discrete wavelet transform is proposed in this research which provides sufficient high compression ratios with no appreciable degradation of image quality. The effectiveness and robustness of this approach has been justified using a set of real images. The images are taken with a digital camera (OLYMPUS LI-40C). To demonstrate the performance of the proposed method, a comparison between the proposed technique and other common compression techniques has been revealed. From the experimental results it is evident that, the proposed compression technique gives better performance compared to other traditional techniques. Wavelets are better suited to time-limited data and wavelet based compression technique maintains better image quality by reducing errors. The future direction of this research is to implement a compression technique using neural network.

## References

[1] *S. Bhavani, K. Thanushkodi, "A Survey on Coding Algorithms in Medical Image Compression", International Journal on Computer Science and Engineering, Vol. 02, No. 05, pp. 1429-1434, 2010.*

[2] *G. K. Kharate, V. H. Pati, "Color Image Compression Based On Wavelet Packet Best Tree", International Journal of Computer Science, Vol. 7, No.3, March 2010.*

[3] *M. R. Haque and F. Ahmed, "Image data compression with JPEG and JPEG2000", 8th International Confrrence on Computer and Information Technology, pp. 1064-1069, 2005.*

[4] *M. J. Weinberger, G. Seroussi and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS", IEEE Trans. on Image Processing, Vol. 2, pp. 1309-1324, Aug. 2000.*

[5] *Madhuri A. Joshi, "Digital Image Processing, An Algorithmic Approach", PHI, New Delhi, pp. 175-217, 2006.*

[6] *C. M. Rahman and A. Y. Saber, "Image compression using dynamic clusturing and neural network", 5th International Confrrence on Computer and Information Technology, pp. 453-458, December 2002.*

[7] *N. A. Koli, M. S. Ali, "A Survey on Fractal Image Compression Key Issues", Information Technology Journal, 2008.*

[8] *L. Bottou, P. G. Howard, and Y. Bengio, "The Z-Coder Adaptive Binary Coder, Proc. IEEE DCC, pp. 13-22, Mar. 1998.*