ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 1, Issue 5, July 2012

ANALYSIS OF WAVELET TRANSFORM AND FAST WAVELET TRANSFORM FOR IMAGE COMPRESSION: REVIEW

1. Er.Naveen Dhillon
H.O.D (Prof) of E.C.E Deptt
R.I.E.T, Phagwara

2. Er. Kuldeep Sharma
Asstt. Prof in ECE
R.I.E.T, Phagwara

3.Er. Kanika Kaushal
M.Tech Student
R.I.E.T, Phagwara

Abstract— **A Detailed study of discrete wavelet transform and fast wavelet transform has been presented in this paper. Older techniques for image compression such as FFT and DCT have also been discussed. Wavelet analysis represents the next logical step: a windowing technique with variable-sized regions. Wavelet analysis allows the use of long time intervals where we want more precise low-frequency information, and shorter regions where we want high-frequency information. Wavelet Based Analysis provide one of the common goals of image compression i.e. the signal or image clearance and simplification, which are parts of de-noising or compression. Fast Wavelet Transform is the latest technique of wavelet transform that is used to perform image analysis at a faster scale than discrete wavelet transform. Mallat Algorithm for Fast Wavelet transform have been presented. Image Compression for different techniques including FFT, DCT, DWT and FWT has been presented in terms of compression ratios and their PSNR values for image quality.**

Key Words— **Wavelet Analysis, Image Compression, Wavelet Decomposition.**

I. INTRODUCTION

In Recent years, many studies have been made on wavelets. An excellent overview of what wavelets have brought to the fields as diverse as biomedical applications, wireless communications, computer graphics or turbulence, is given in [1]. Image compression is one of the most visible applications of wavelets. The rapid increase in the range and use of electronic imaging justifies attention for systematic design of an image compression system and for providing the image quality needed in different applications. The basic measure for the performance of a compression algorithm is compression ratio (CR), defined as a ratio between original data size and compressed data size. In a lossy compression scheme, the image compression algorithm should achieve a tradeoff between compression ratio and image quality [4]. Higher compression ratios will produce lower image quality and vice versa. Quality and compression can also vary according to input image characteristics and content.

Transform coding is a widely used method of compressing image information. In a transform-based compression system two-dimensional (2-D) images are transformed from the spatial domain to the frequency domain. An effective transform will concentrate useful information into a few of the low-frequency transform coefficients. An HVS is more sensitive to energy with low spatial frequency than with high spatial frequency.

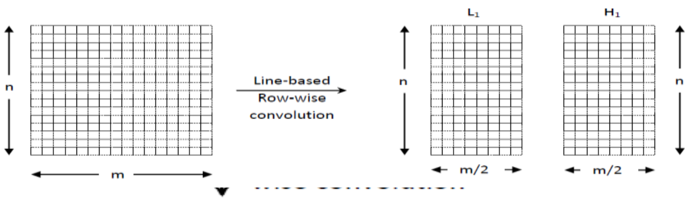
Therefore, compression can be achieved by quantizing the coefficients, so that important coefficients (low-frequency coefficients) are transmitted and the remaining coefficients are discarded. Very effective and popular ways to achieve compression of image data are based on the discrete cosine transform (DCT) and discrete wavelet transform (DWT).

In Discrete Wavelet Transform, signal energy concentrates to specific wavelet coefficients. This characteristic is useful for compressing images [7].The multiresolution nature of the discrete wavelet transform is proven as a powerful tool to represent images decomposed along the vertical and horizontal directions using the pyramidal multiresolution scheme. Discrete wavelet transform helps to test different allocations using sub band coding, assuming that details at high resolution and diagonal directions are less visible to the human eye. By using an error correction method that approximates the reconstructed coefficients quantization error, we minimize distortion for a given compression rate at low computational cost. The main property of DWT is that it includes neighborhood information in the final result, thus avoiding the block effect of DCT transform.

It also has good localization and symmetric properties, which allow for simple edge treatment, high-speed computation, and high quality compressed image [8]. The 2D DWT has also gained popularity in the field of image and video coding, since it allows good complexity-performance tradeoffs and outperforms the discrete cosine transform at very low bit rates. In general the wavelet transform requires much less hardware to implement than Fourier methods, such as the DCT. Recently, much of the research activities in image coding have been focused on the DWT, which has become a standard tool in image compression applications because of their data reduction capability [10]–[12]. In a wavelet compression system, the entire image is transformed and compressed as a single data object rather than block by block as in a DCT-based compression system. It allows a uniform distribution of compression error across the entire image. DWT offers adaptive spatial-frequency resolution (better spatial resolution at high frequencies and better frequency resolution at low frequencies) that is well suited to the properties of an HVS. It can provide better image quality than DCT, especially on a higher compression ratio [13]. However, the implementation of the DCT is less expensive than that of the DWT. For example, the most efficient algorithm for 2-D 8 x 8 DCT requires only 54 multiplications [14], while the complexity of calculating the DWT depends on the length of wavelet filters.

II. DISCRETE WAVELET TRANSFORM

Wavelet transform (WT) represents an image as a sum of wavelet functions (wavelets) with different locations and scales [17]. Any decomposition of an image into wavelets involves a pair of waveforms: one to represent the high frequencies corresponding to the detailed parts of an image (wavelet function ψ) and one for the low frequencies or smooth parts of an image (scaling function Ø). The Discrete wavelet transform (DWT) has gained wide popularity due to its excellent decorrelation property, many modern image and video compression systems embody the DWT as the transform stage. It is widely recognized that the 9/7 filters are among the best filters for DWT-based image compression. In fact, the JPEG2000 image coding standard employs the 9/7 filters as the default wavelet filters for lossy compression and 5/3 filters for lossless compression. The performance of a hardware implementation of the 9/7 filter bank (FB) depends on the accuracy with which filter coefficients are represented. Lossless image compression techniques find applications in fields such as medical imaging, preservation of artwork, remote sensing etc [4] . Day-by-day Discrete Wavelet Transform (DWT) is becoming more and more popular for digital image compression. Biorthogonal (5, 3) and (9, 7) filters have been chosen to be the standard filters used in the JPEG2000 codec standard. Discrete wavelet transform as reported by Zervas et al., there are three basic architectures for the two-dimensional DWT: level-by-level, line-based, and block-based architectures. In implementing the 2-D DWT, a recursive algorithm based on the line based architectures is used. The image to be transformed is stored in a 2-D array. Once all the elements in a row are obtained, the convolution is performed in that particular row [2]. The process of row-wise convolution will divide the given image into two parts with the number of rows in each part equal to half that of the image. This matrix is again subjected to a recursive line-based convolution, but this time column-wise [2] . The result will DWT coefficients corresponding to the image, with the approximation coefficient occupying the top-left quarter of the matrix, horizontal coefficients occupying the bottom-left quarter of the matrix, vertical coefficients occupying the top-right quarter of the matrix and the diagonal coefficients occupying the bottom-right quarter of the matrix[3] .



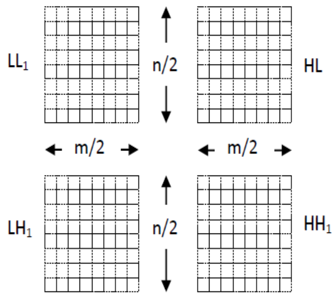


***Figure 1**: Line based Architecture for DWT*

After DWT was introduced, several codec algorithms were proposed to compress the transform coefficients as much as possible. Among them, Embedded Zerotree Wavelet (EZW), Set Partitioning In Hierarchical Trees (SPIHT) and Embedded Bock Coding with Optimized Truncation (EBCOT) are the most famous ones The embedded zero tree wavelet algorithm (EZW) is a simple, yet remarkably effective image compression algorithm, having the property that the bits in the bit stream are generated in top-right quarter of the matrix and the diagonal coefficients occupying the bottom-right quarter of the matrix. After DWT was introduced, several codec algorithms were proposed to compress the transform coefficients as much as possible. Among them, Embedded Zero tree Wavelet (EZW), Set Partitioning In Hierarchical Trees (SPIHT) and Embedded Bock Coding with Optimized Truncation (EBCOT) are the most famous ones. The embedded zerotree wavelet algorithm (EZW) is a simple, yet remarkably effective image compression algorithm, having the property that the bits in the bit stream are generated in order of importance, yielding a fully embedded code. The embedded code represents a sequence of binary decisions that distinguish an image from the "null" image. Using an embedded coding algorithm, an encoder can terminate the encoding at any point thereby allowing a target rate or target distortion metric to be met exactly [3]. Also, given a bit stream, the decoder can cease decoding at any point in the bit stream and still produce exactly the same image that would have been encoded at the bit rate corresponding to the truncated bit stream. In addition to producing a fully embedded bit stream, EZW consistently produces compression results that are competitive with virtually all known compression algorithms on standard test images. Yet this performance is achieved with a technique that requires absolutely no training, no pre-stored tables or codebooks and requires no prior knowledge of the image source. The EZW algorithm is based on four key concepts:

1) A discrete wavelet transform or hierarchical sub band decomposition.

2) Prediction of the absence of significant information across scales by exploiting the self -similarity inherent in images.

3) Entropy-coded successive-approximation quantization, and

4) Universal lossless data compression which is achieved

via adaptive arithmetic coding.

III. Need For Compression

Compression is necessary in modern data transfer and processing whether it is performed on data or an image/video file as transmission and storage of uncompressed video would be extremely costly and impractical [2] . Framensm with 352 x 288 contains 202,752 bytes of information. Recording of uncompressed version of this video at 15 frames per second would require 3 MB. As 180 MB of data storage would be required for 1 minute and hence one 24 hours day would be utilized to store 262 GB of database. Using Compression, at 15 frames per seconds, it takes 24 hrs would take only 1.4GB and hence 187 days of video could be stored using the same disk space that uncompressed video would use in one day. Hence, Compression while maintaining the image quality is must for digital data, image or video file transfer in fast way and lesser amount of time.

IV. Lossless And Lossy Image Compression

When hearing that image data are reduced, one could expect that automatically also the image quality will be reduced. A loss of information is, however, totally avoided in lossless compression, where image data are reduced while image information is totally preserved. It uses the predictive encoding which uses the gray level of each pixel to predict the gray value of its right neighbour [8]. Only the small deviation from this prediction is stored. This is a first step of lossless data reduction. Its effect is to change the statistics of the image signal drastically. Statistical encoding is another important approach to lossless data reduction. Statistical encoding can be especially successful if the gray level statistics of the images has already been changed by predictive coding. The overall result is redundancy reduction that is reduction of the reiteration of the same bit patterns in the data. Of course, when reading the reduced image data, these processes can be performed in reverse order without any error and thus the original image is recovered. Lossless compression is therefore also called reversible compression. Lossy data compression has of course a strong negative connotation and sometimes it is doubted quite emotionally that it is at all applicable in medical imaging. In transform encoding one performs for each image run a mathematical transformation that is similar to the Fourier transform thus separating image information on gradual spatial variation of brightness (regions of essentially constant brightness) from information with faster variation of brightness at edges of the image (compare: the grouping by the editor of news according to the classes of contents). In the next step, the information on slower changes is transmitted essentially lossless (compare: careful reading of highly relevant pages in the newspaper), but information on faster local changes is communicated with lower accuracy (compare: looking only at the large headings on the less relevant pages). In image data reduction, this second step is called quantization. Since this quantization step cannot be reversed when decompressing the data, the overall compression is ‘lossy’ or ‘irreversible’. Table 1 summarizes some of the characteristics of lossless and lossy image compression techniques [9] as described above:

Data compression strategy	lossless / reversible	lossy / irreversible
- based on:	statistics of data	meaning of information
- results in:	redundancy reduction	irrelevancy reduction
Key methods	predictive encoding plus statistical encoding	transform encoding plus quantization
Examples for implementation	lossless JPEG mode	lossy JPEG mode
Compression factor (typical)	2 to 3	6 to 12

Table 1: *Overview on principal strategies and methods in lossless (middle) and lossy (right) image compression.*

V. Different Techniques For Image Compression

A. DFT: Discrete Fourier Transform

The Fourier Transform is an important image processing tool which is used to decompose an image into its sine and cosine components. The output of the transformation represents the image in the *Fourier* or frequency domain, while the input image is the spatial domain equivalent. In the Fourier domain image, each point represents a particular frequency contained in the spatial domain image [11]. The Fourier Transform is used in a wide range of applications, such as image analysis, image filtering, image reconstruction and image compression. The DFT is the sampled Fourier Transform and therefore does not contain all frequencies forming an image, but only a set of samples which is large enough to fully describe the spatial domain image. The number of frequencies corresponds to the number of pixels in the spatial domain image, *i.e.* the image in the spatial and Fourier domain is of the same size. For a square image of size N×N, the two-dimensional DFT is given by:

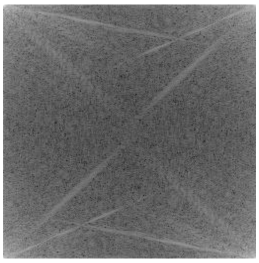
$$F(k,l) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i,j)\, e^{-\iota 2\pi\left(\frac{ki}{N}+\frac{lj}{N}\right)}$$

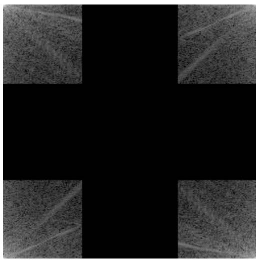
where *f(a,b)* is the image in the spatial domain and the exponential term is the basis function corresponding to each point *F(k,l)* in the Fourier space [11]. The equation can be interpreted as: the value of each point *F(k,l)* is obtained by multiplying the spatial image with the

$$f(a,b) = \frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} F(k,l)\, e^{\iota 2\pi\left(\frac{ka}{N}+\frac{lb}{N}\right)}$$

corresponding base function and summing the result. In a similar way, the Fourier image can be re-transformed to the spatial domain. The inverse Fourier transform is given by:

(a)





(b) *(c)*

(d)

Figure 2 *: a) Input Image without compression*
b) and c) Discrete Fourier Transforms
d) Output Image with Inverse transform that has been reconstructed

B. FFT: FAST FOURIER TRANSFORM

A fast Fourier transform (FFT) is an efficient algorithm to compute the discrete Fourier transform (DFT) and its inverse. “There are many distinct FFT algorithms involving a wide range of mathematics, from simple arithmetic to group theory and number theory. A DFT decomposes a sequence of values into components of different frequencies. This operation is useful in many fields but computing it directly from the definition is often too slow to be practical [11]. An FFT is a way to compute the same result more quickly: computing a DFT of N points in the naive way, using the definition, takes $O(N^2)$ arithmetical operations, while an FFT can compute the same result in only $O(N \log N)$ operations. The difference in speed can be substantial, especially for long data sets where N may be in the thousands or millions—in practice, the computation time can be reduced by several orders of magnitude in such cases, and the improvement is roughly proportional to $N / \log (N)$. This huge improvement made many DFT-based algorithms practical; FFTs are of great importance to a wide variety of applications, from digital and solving partial differential equations to algorithms for quick multiplication of large integers [11].

a) *b)*

c) *d)*

Figure 3: a) The Original Cameraman Image
b) Cameraman Image after decompression (cut off=20, MSE=36.82) using FFT
c) Cameraman Image after decompression (cut off=40, MSE=102.43) using FFT
d) Cameraman Image after decompression (cut off=60, MSE=164.16) using FFT

C. DCT: DISCRETE COSINE TRANSFORM

The 1D DCT is defined as

$$\mathbf{D}_X\{f\}(\omega) = a(\omega)\sum_{x=0}^{N-1} f(x)\cos\left(\frac{(2x+1)\omega\pi}{2N}\right) \quad ; \quad a(\omega) = \begin{cases} \sqrt{\frac{1}{N}} & \omega = 0 \\ \sqrt{\frac{2}{N}} & else \end{cases}$$

which is similar to the DFT

$$F(\omega) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)e^{-\frac{j2\pi x\omega}{N}}$$

2D DCT is defined using the separability property as 1D transform on the rows and on the columns, applied separately:

$$F(u,v) = \mathbf{D}_Y\{\mathbf{D}_X\{f(x,y)\}\}$$

One of the advantages of DCT is the fact that it is a real transform, whereas DFT is complex. This implies lower computational complexity, which is sometimes important for real-time applications. DCT is used in some lossy compression algorithms, including JPEG. (The JPEG standard codec is more complicated, for it includes a quantizer for DCT coefficients and DPCM statistical prediction scheme. The output of the codec is the prediction error, which is encoded using some lossless entropy code.) In the transform image, DC is the matrix element (1,1), corresponding to transform value X(0,0). High spatial X and Y frequencies correspond to high column and row indexes, respectively.

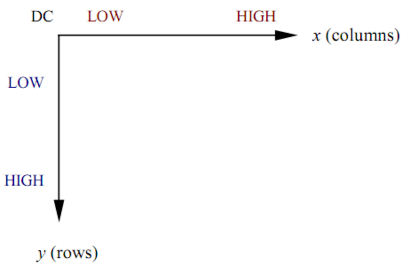


Figure 4: DCT image. Red labels denote horizontal spatial frequencies, blue label denote vertical frequencies [12].

According to the DCT properties, a DC is transformed to discrete delta-function at zero frequency. Hence, the transform image contains only the DC component. This can be seen in the transform image. The DC value is a sum over the whole image. The majority of the DCT energy is concentrated on low frequencies. The reason is the fact that natural images possess mostly low-frequency features and high-frequency features (edges) are rarely encountered. The advantages of DCT compression are based on the fact that most natural images have sparse edges. Hence, most blocks contain primarily low frequencies, and can be represented by a small number of coefficients without significant precision loss. Edges are problematic since are associated with high spatial frequency. Consequently, the DCT at blocks where the edges pass has high-amplitude coefficients at high frequencies, which cannot be removed without significant distortion. This effect was seen on the coin image, where small number of coefficients resulted in very significant distortion of the edges. Images containing non-sparse edges, such as the standard MATLAB image of an integral circuit (IC), are very Problematic for such compression method, since they primarily consist of edges. The compression algorithm can be significantly improved if the coefficients selection would be adaptive, i.e. in each DCT block we would select a different number of coefficients with the largest amplitude. Thus, smooth regions of an image can be represented by a small number of coefficients, whereas edges and high-frequency textures would be represented by large number of coefficients. This will solve the problem of edges, whilst leaving the algorithm efficient. The number of coefficients required for edge representation can be reduced if the idea of over complete base is used.

a) b)

Figure 5: a) Original Image Of Cameramen
b) Reconstructed Image after applying 8 x 8 block subset using DCT

Although there is some loss of quality in the reconstructed image, it is clearly recognizable, even though almost 85% of the DCT coefficients were discarded. To experiment with discarding more or fewer coefficients, and to apply this technique to other images, we have developed a modified algorithm using given equations:

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \quad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p = 0 \\ \sqrt{2/M}, & 1 \le p \le M-1 \end{cases} \qquad \alpha_q = \begin{cases} 1/\sqrt{N}, & q = 0 \\ \sqrt{2/N}, & 1 \le q \le N-1 \end{cases}$$

Where, M and N are the row and column size of A, respectively. If you apply the DCT to real data, the result is also real. The DCT tends to concentrate information, making it useful for image compression applications.

D. DWT: DISCERETE WAVELET TRANSFORM

Traditionally, image compression adopts discrete cosine transform (DCT) in most situations which possess the characteristics of simpleness and practicality. DCT has been applied successfully in the standard of JPEG, MPEGZ, etc. However, the compression method that adopts DCT has several shortcomings that become increasing apparent. One of these shortcomings is obvious blocking artifact and bad subjective quality when the images are restored by this method at the high compression ratios [2]. In recent years, many studies have been made on wavelets. An excellent overview of what wavelets have brought to the fields as diverse as biomedical applications, wireless communications, computer graphics or turbulence. Image compression is one of the most visible applications of wavelets. The rapid increase in the range and use of electronic imaging justifies attention for systematic design of an image

compression system and for providing the image quality needed in different applications [4].
The discrete wavelet transform (DWT) refers to wavelet transforms for which the wavelets are discretely sampled. A transform which localizes a function both in space and scaling and has some desirable properties compared to the Fourier transform. The transform is based on a wavelet matrix, which can be computed more quickly than the analogous Fourier matrix. Most notably, the discrete wavelet transform is used for signal coding, where the properties of the transform are exploited to represent a discrete signal in a more redundant form, often as a preconditioning for data compression. The discrete wavelet transform has a huge number of applications in Science, Engineering, Mathematics and Computer Science. Wavelet compression is a form of data compression well suited for image compression (sometimes also video compression and audio compression). The goal is to store image data in as little space as possible in a file. A certain loss of quality is accepted (lossy compression).Using a wavelet transform, the wavelet compression methods are better at representing transients, such as percussion sounds in audio, or high-frequency components in two-dimensional images, for example an image of stars on a night sky. This means that the transient elements of a data signal can be represented by a smaller amount of information than would be the case if some other transform, such as the more widespread discrete cosine transform, had been used.
First a wavelet transform is applied. This produces as many coefficients as there are pixels in the image (i.e.: there is no compression yet since it is only a transform). These coefficients can then be compressed more easily because the information is statistically concentrated in just a few coefficients. This principle is called transform coding. After that, the coefficients are quantized and the quantized values are entropy encoded and/or run length encoded.
Examples for Wavelet Compressions:
· JPEG 2000
· Ogg
· Tarkin
· SPIHT
· MrSID
· Dirac
Wavelet analysis represents the next logical step: a windowing technique with variable-sized regions. Wavelet analysis allows the use of long time intervals where we want more precise low-frequency information, and shorter regions where we want high-frequency information.

Figure 6: Wavelet Transform on a signal.

Wavelet Transform in contrast with the time-based, frequency-based, and STFT views of a signal:

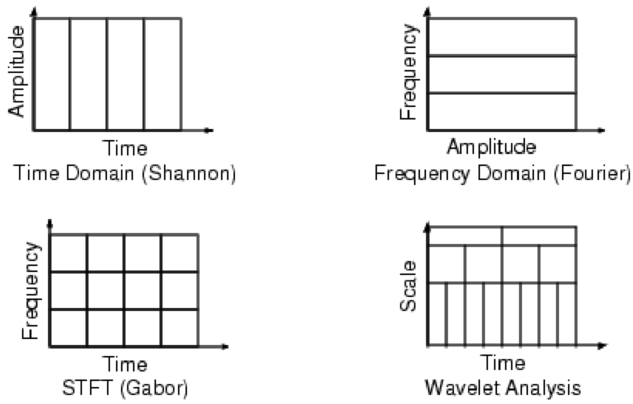


Figure 7: Comparison of Various Transform Techniques

Wavelet analysis does not use a time-frequency region, but rather a time-scale region.
Wavelets have scale aspects and time aspects; consequently every application has scale and time aspects. To clarify them we try to untangle the aspects somewhat arbitrarily.

5.1 MULTILEVEL DECOMPOSITION

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components. This is called the wavelet decomposition tree.

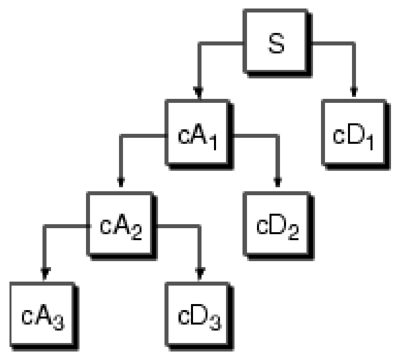


Figure 8: Multilevel Decomposition

Lifting schema of DWT has been recognized as a faster approach

- The basic principle is to factorize the polyphase matrix of a wavelet filter into a sequence of alternating upper and lower triangular matrices and a diagonal matrix.
- This leads to the wavelet implementation by means of banded-matrix multiplications

ALGORITHM follows a quantization approach that divides the input image in 4 filter coefficients as shown below, and then performs further quantization on the lower order filter or window of the previous step. This quantization depends upon the decomposition levels and maximum numbers of decomposition levels to be entered are 3 for DWT.

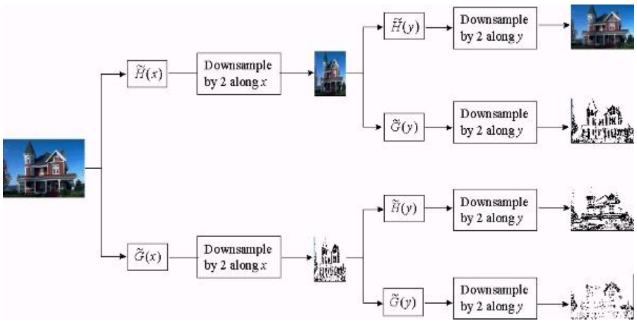


Figure 9: Wavelet Decomposition of Image

5.2 WAVELET RECONSTRUCTION

The filtering part of the reconstruction process also bears some discussion, because it is the choice of filters that is crucial in achieving perfect reconstruction of the original signal. The down sampling of the signal components performed during the decomposition phase introduces a distortion called aliasing. It turns out that by carefully choosing filters for the decomposition and reconstruction phases that are closely related (but not identical), we can "cancel out" the effects of aliasing.

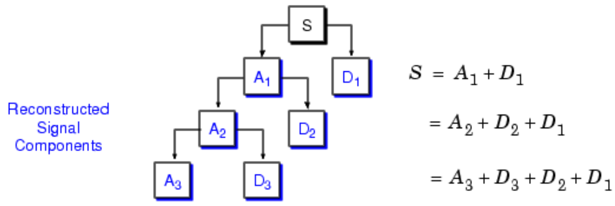


Figure 10: Wavelet Reconstruction

Results of Wavelet transform on Lena image for compression ratio of 2:1 is shown below [6] :

Figure 11: *Comparison of optimal wavelet functions for image Lena (PSNR = 36 dB). (a) Original. (b) DW-5 (PQS = 2.93). (c) CW-3 (PQS = 3.10). (d) BW-2.2*

6. FAST WAVELET TRANSFORM

In 1988, Mallat produced a fast wavelet decomposition and reconstruction algorithm [Mal89]. The Mallat algorithm for discrete wavelet transform (DWT) is, in fact, a classical scheme in the signal processing community, known as a two-channel sub band coder using conjugate quadrature filters or quadrature mirror filters (QMFs).

- The decomposition algorithm starts with signal *s*, next calculates the coordinates of A_1 and D_1, and then those of A_2 andD_2, and so on.
- The reconstruction algorithm called the inverse discrete wavelet transform (IDWT) starts from the coordinates of $A_J$and D_J, next calculates the coordinates of A_{J-1}, and then using the coordinates of A_{J-1} and D_{J-1} calculates those of A_{J-2}, and so on.

In order to understand the multiresolution analysis concept based on Mallat's algorithm it is very useful to represent the wavelet transform as a pyramid, as shown in figure 12. The basis of the pyramid is the original image, with C columns and R rows.

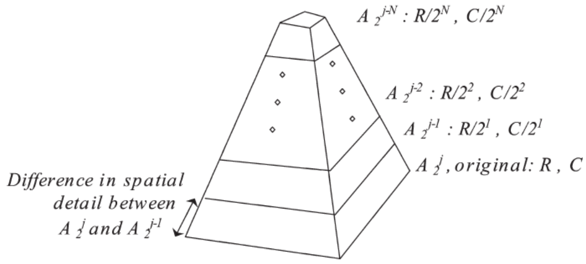
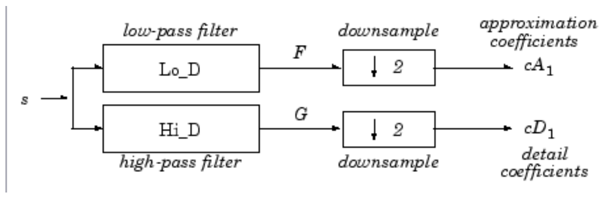


Figure 12: *Pyramidal representation of Mallat's wavelet decomposition algorithm.*

Given a signal *s* of length *N*, the DWT consists of $\log_2 N$ stages at most. Starting from *s*, the first step produces two sets of coefficients: approximation coefficients cA_1, and detail coefficients cD_1. These vectors are obtained by convolving *s* with the low-pass filter Lo_D for approximation, and with the high-pass filter Hi_D for detail, followed by dyadic decimation.



The length of each filter is equal to $2n$. If N = length (s), the signals F and G are of length $N + 2n - 1$, and then the coefficients cA_1 and cD_1 are of length

$$\text{floor}\left(\frac{(N-1)}{2}+n\right)$$

The next step splits the approximation coefficients cA_1 in two parts using the same scheme, replacing *s* by cA_1 and producing cA_2 and cD_2, and so on.

Classically, the DWT is defined for sequences with length of some power of 2, and different ways of extending samples of other sizes are needed. Methods for extending the signal include zero-padding, smooth padding, periodic extension, and boundary value replication (summarization). The basic algorithm for the DWT is not limited to dyadic length and is based on a simple scheme: convolution and downsampling [13]. As usual, when a convolution is performed on finite-length signals, border distortions arise.

To Remove these border effects , Fast Wavelet Transform was introduced. This algorithm is a method for the extension of a given finite-length signal [12].
Let us denote h = Lo_R and g = Hi_R and focus on the one-dimensional case.
We first justify how to go from level j to level j+1, for the approximation vector. This is the main step of the decomposition algorithm for the computation of the approximations. The details are calculated in the same way using the filter g instead of filter h.
Let $(A_k^{(j)})k \in Z$ be the coordinates of the vector A_j:

$$A_j = \sum_k A_k^{(j)} \phi_{j,k}$$

and $A_k^{(j+1)}$ the coordinates of the vector A_{j+1}:

$$A_{j+1} = \sum_k A_k^{(j+1)} \phi_{j+1,k}$$

$A_k^{(j+1)}$ is calculated using the formula

$$A_k^{(j+1)} = \sum_n h_{n-2k} A_n^{(j)}$$

This formula resembles a convolution formula.

The computation is very simple.

Let us define

$$\tilde{h}(k) = h(-k), \text{ and } F_k^{(j+1)} = \sum_n \tilde{h} k - n\, An^{(j)}$$

The sequence $F^{(j+1)}$ is the filtered output of the sequence $A^{(j)}$ by the filter $\tilde{h}$.

We obtain

$$A_k^{(j+1)} = F_{2k}^{(j+1)}$$

We have to take the even index values of F. This is downsampling.
The sequence $A^{(j+1)}$ is the downsampled version of the sequence $F^{(j+1)}$.
The initialization is carried out using $A_k^{(0)} = s(k)$, where $s(k)$ is the signal value at time k.
There are several reasons for this surprising result, all of which are linked to the multiresolution situation and to a few of the properties of the functions $\varphi_{j,k}$ and $\psi_{j,k}$.

REFERENCES

[1] *Image Compression using Wavelets:* Sonja Grgc, Kresimir Kers, Mislav Grgc, University of Zagreb, IEEE publication, 1999.

[2] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, *Image coding using wavelet transform*, IEEE Trans. Image Processing, vol. 1, pp.205-220, 1992.

[3] P.L. Dragotti, G. Poggi, and A.R.P. Ragozini, *Compression of multispectral images by three-dimensional SPIHT algorithm*, IEEE Trans. on Geoscience and remote sensing, vol. 38, No. 1, Jan 2000.

[4] *Image compression using Fast Fourier Transform* by Parminder Kaur, Thapar Institute of Engineering and Technology, Patiala.

[5] S-T. Hsiang and J.W. Woods, *Embedded image coding using zeroblocks of subband/wavelet coefficients and context modeling*, IEEE Int. Conf. on Circuits and Systems (ISCAS2000), vol. 3, pp.662-665, May 2000.

[6] *Implementation of Data Compression and FFT* on TinyOS: Ning Xu, Embedded Networks Laboratory, Computer Science Dept. USC. Los Angeles.

[7] B. Kim and W.A. Pearlman, *An embedded wavelet video coder using three-dimensional set partitioning in hierarchical tree*, IEEE Data Compression Conference, pp.251-260, March 1997.

[8] Y. Kim and W.A. Pearlman, *Lossless volumetric medical image compression*, Ph.D Dissertation, Department of Electrical, Computer,and Systems Engineering, Rensselaer Polytechnic Institute, Troy, 2001.

[9] J. Li and S. Lei, *Rate-distortion optimized embedding*, in Proc. Picture Coding Symp., Berlin, Germany, pp. 201-206, Sept. 10-12, 1997.

[10] E. Ordentlich, M. Weinberger, and G. Seroussi, *A low-complexity modeling approach for embedded coding of wavelet coef_cients*, in Proc. IEEE Data Compression Conf., Snowbird, UT, pp. 408-417, Mar. 1998.

[11] A.N. Netravali and B.G. Haskell, *Digital pictures, representation and compression*, in Image Processing, Proc. of Data Compression Conference, pp.252-260, 1997.

[12] S. Mallat, *Multifrequency channel decompositions of images and wavelet models*, IEEE Trans. Acoust., Speech, Signal Processing, vol. 37, pp.2091-2110, Dec. 1989.

[13] W.A. Pearlman, *Performance bounds for subband codes*, Chapter 1 in *Subband Image Coding*, J. W. Woods and Ed. Klvwer. Academic Publishers, 1991.

[14] *Proposal of the arithmetic coder for JPEG2000*, ISO/IEC/JTC1/SC29/WG1 N762, Mar. 1998.

[15] A. Said and W.A. Pearlman, *A new, fast and efficient image codec based on set partitioning in hierarchical trees*, IEEE Trans. on Circuits and Systems for Video Technology 6, pp. 243-250, June 1996.

[16] P. Schelkens, *Multi-dimensional wavelet coding algorithms and implementations*, Ph.D dissertation, Department of Electronics and Information Processing, Vrije Universiteit Brussel, Brussels, 2001.

[17] J.M. Shapiro, *Embedded image coding using zerotrees of wavelet coef_cients*, IEEE Trans. Signal Processing, vol. 41, pp.3445-3462, Dec. 1993.
[18] D. Taubman, *High performance scalable image compression with EBCOT*, IEEE Trans. on Image Processing, vol. 9, pp.1158-1170, July,2000.
[19] J.W. Woods and T. Naveen, *Alter based bit allocation scheme for subband compression of HDTV*, IEEE Transactions on Image Processing, IP-1:436-440, July 1992.November 7.
[20] I.H. Witten, R.M. Neal, and J.G. Cleary, *Arithmetic coding for data compression*, Commun. ACM, vol. 30, pp. 520-540, June 1987.