WAVELET-DCT BASED IMAGE CODER
FOR VIDEO CODING APPLICATIONS

By

REGINA GANI
7168

FINAL PROJECT REPORT

submitted to the Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirements for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Wavelet-DCT Based Image Coder
for Video Coding Applications
(An Enhancement via Thresholding)

by

Regina Gani

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirements for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

Assoc. Prof. Dr. Varun Jeoti
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

June 2008

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by any unspecified sources or persons.

Regina Gani

ABSTRACT

Utilization of digital video raises important concern on storage space and transmission purpose. A video codec that compresses digital video signal has become an essential component in video application.

The interest of this project is to integrate a wavelet-based compression and at the same time providing a backward compatibility to the existing architecture of MPEG-2 standard. Wavelet-DCT is an algorithm that utilizes Discrete Wavelet Transform (DWT) to compute Discrete Cosine Transform (DCT) coefficients. The algorithm allows truncation of intermediate coefficient through DWT. This process is also known as thresholding. The Wavelet-DCT algorithm is integrated into a simulation of intra-frame coder in compliance with MPEG-2 standard. To enhance the system performance, threshold level selection is done via histogram energy analysis and selection of best wavelet type through statistical study.

The simulation was developed using SIMULINK and MATLAB. Wavelet-DCT based intra-frame coder was simulated by employing two different approaches. Different types of multimedia source (image and video) were tested on the simulation.

The result from both approaches for images shows marginal improvement at higher compression ratio. On the other hand, when it was tested on video, which consists of different types of images, both approaches does not show a reasonable improvement at various level of compression ratio.

The argument on why wavelet-DCT based intra-frame coder does not perform better arises from the objective's restriction that attempts to cater the backward compatibility. Some recommended areas to be delved for future work on video compression are distributed video coding, ridgelet transform and dual-tree complex wavelet transform.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my project supervisor, Assoc. Prof. Dr. Varun Jeoti who introduced me to the world of video compression and especially the theory of wavelet. Despite his busy schedule, Dr. Varun always spares some time to give advices, ideas and to address doubts and questions that I had in mind. His encouragement and guidance have provided a good basis for the present thesis.

I wish to express my warm and sincere thanks to Mr. Ho Tatt Wei who gave me untiring help and encouragement during difficult moments. His wide knowledge and way of thinking have been a great value for me.

I warmly thank Chai Beng Seow, who first started the feasibility study of this project. He has been helpful and supportive during the tough time at the beginning of this study.

My sincere thanks are due to the lecturers and graduate assistants who have given their review, constructive criticism and excellent advices during the preparation of this thesis.

During this work, I have received help and supports from my friends in UTP for whom I have great regard.

I would also like to express my love and gratitude to my parents, brothers and Barry Agustinus for their understanding, constant support and encouragement through the duration of my studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

B-Picture	Bidirectionally predictive-coded Picture
BR	Bit Rate
CR	Compression Ratio
CTI	Continuous Tone Image
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DTI	Discontinuous Tone Image
DVB-T	Digital Video Broadcasting
DWT	Discrete Wavelet Transform
DWPT	Discrete Wavelet Packet Transform
FFT	Fast Fourier Transform
FAFT	Fast Approximate Fourier Transform
HDTV	High Definition Television
I-Picture	Intra-coded Picture
ISO	International Organization for Standardization
IDCT	Inverse Discrete Cosine Transform
IDWT	Inverse Discrete Wavelet Transform
JPEG	Joint Photographic Experts Group
MPEG	Motion Picture Experts Group
MSE	Mean Squared Error
P-Picture	Predictive-coded Picture
PDF	Probability Density Function
PSNR	Peak to Signal Noise Ratio
RGB	Red, Green, Blue
SDTV	Standard Definition Television
TL	Threshold Level
YCbCr	Luminance, Chrominance blue, Chrominance red

CHAPTER 1
INTRODUCTION

Digital video has become an essential part in many aspects of life such as business, health-care, education, entertainment, and others. Digital video coder/codec is the module that compresses and decompresses a digital video signal which is necessary as bandwidth is still a highly valuable commodity and utilization of less memory is desired. Some applications of video codec are HDTV, DVD movies, Video Conferencing, Distant Learning, Video Streaming, etc. The aim of a video compression system is to reduce the amount of information of the picture sequence and to be able to retrieve an acceptable visual quality.

The currently existing video codec standard used for such application is MPEG-2. The intra-frame coder of MPEG-2 performs a DCT-based compression. However, many studies claim that wavelet-based compression offers better performance. Wavelet-DCT is an algorithm that able to provide a wavelet-based compression while maintaining the backward compatibility on MPEG 2 architecture.

1.1. Background Study

A video codec is a device or program with the capability of performing encoding and decoding on digital video signal. The word codec may be a combination of: 'Encoder/decoder' or 'compression-decompression algorithm'. Figure 1.1 illustrates a typical model of a codec as part of communication system.

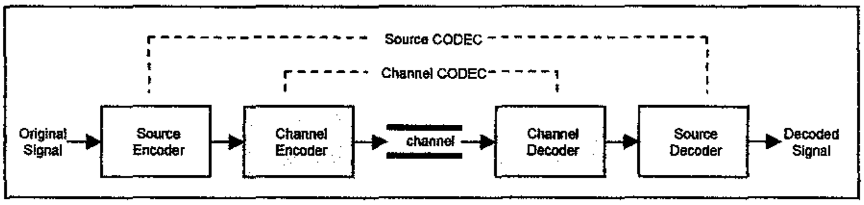


Figure 1.1 Source coder, channel coder, channel

In video coding system, a source encoder compresses (encodes) digital video signals to a form that requires fewer bits than the original signal. The source compressed signal is then further encoded by channel coder as an error protection before it is transmitted over the channel. At the receiver, a channel decoder corrects the transmission errors and source decoder decompresses the signal. If the decompressed signal exactly matches the original signal, it is called lossless compression. Meanwhile, most decompressed signals are distorted compared to the original signal, which is called as lossy compression.

Discrete Cosine Transform (DCT) is the most commonly used transform coding in video and image compression due to its benefits as listed:

- DCT coefficients are real-valued
- DCT has near-optimal energy compaction
- DCT computational efficiency is comparable to the FFT

JPEG2000 has introduced a wavelet-based image compression standard. It is proven to have superior compression performance over baseline JPEG, which uses DCT. This compression standard is designed for natural images whose intensity level changes gradually, thus categorized as Continuous Tone Image (CTI).

Within a video coder, an inter-frame coder subtracts the current frame of image with the reference frame to obtain difference image, which is categorized as Discontinuous Tone Image (DTI). This difference image is fed into the intra-frame coder, a still image coder that attempts to perform compression on spatial redundancy.

MPEG-2, known as the digital entertainment TV standard, is a follow up work from MPEG-1 when the need to have digital compression method for SDTV, HDTV, DVB-T and other entertainment TV arose. Thus, ISO developed a second standard known colloquially as MPEG-2 and officially as ISO 13818.

1.2. Problem Statement

Typical video coding system consists of an intra-frame and inter-frame coder. Inter-frame coder performs subtraction to obtain the difference image of current frame and previous frame. The aim of inter-frame coder is to reduce the size of the individual image fed into intra-frame coder. Then, the image which is basically difference images are further compressed in the intra-frame coder.

Discrete Cosine Transform (DCT) is the transform that is most commonly used in video coder due to its near-optimal energy compaction property. For CTI, DCT as the transform coding works fine in compressing energy. However, it is not the case for DTI that compose about 88% of all the frame sequence. In [3], a Wavelet-DCT algorithm was proposed as a new method in performing Fast DCT. It utilizes wavelet transform to compute DCT which then includes a wavelet-based compression within computation process of DCT coefficients. The feasibility study of this algorithm was done by [1] particularly for intra-frame coder in video coding application.

In video coding application, some parameters that determine the system performance are compression performance (quantified by CR) and image quality (quantified by PSNR). An intermediate-thresholding-capability introduced in this Wavelet-DCT intra-frame coder is essential to provide good trade off between both parameters. This capability also further reduces the computational complexity on top of fast DCT implementation.

Wavelet-DCT based intra-frame coder can offer even more compression if an appropriate threshold level is chosen. In the previous study, the threshold level was chosen by means of trial and error method. In this project, an attempt is made to equip the intra-frame coder with an algorithm that can automatically determine the best threshold level in order to improve the system performance.

In addition, the previous study by [1] was conducted only using Haar Wavelet as the basis. No study has been conducted to optimize the current performance of Wavelet-DCT algorithm in video coding application. For this reason, it is a necessity to investigate for the most suitable wavelet for difference image to improve the performance of Wavelet-DCT intra frame coder.

1.3. Objectives

The interest of this project is to integrate a wavelet-based compression and at the same time provide a backward compatibility to the existing MPEG-2 standard and to further improve the compression performance. Thus, the objectives are as listed:

- To develop the Wavelet-DCT algorithm in MATLAB.
- To develop a SIMULINK based full intra-frame video coding simulation, in compliance with MPEG-2 standard.
- To observe and improve the compression performance of Wavelet-DCT based intra-frame coder by study and analysis on Threshold Level (TL) selection.
- To investigate the best wavelet type to compress DTI.

1.4. Scope of Study

- The intra-frame coder developed is a replica model from a typical video coder.
- The input to the intra-frame coder is difference image (DTI) and/or natural images (CTI).
- The project does not cover full video coder development, but only the intra-frame coder.
- The DCT is applied to blocks of image samples 16×16 squares.

1.5. Chapter Organization

The report is organized into 5 chapters. Chapter 1 is an introduction of the study which includes background study, problem statement, objectives and scope of study. Chapter 2 provides an overview of the previous study on Wavelet-DCT based Image Coder for Video Coding Application. It also discusses some theoretical details on wavelet transform, DCT, MPEG-2 standard and Wavelet-DCT. Chapter 3 is an elaboration on the proposed techniques, steps taken to accomplish the objectives and system performance parameters. Chapter 4 provides the discussion and analysis of the result obtained from the simulation. Chapter 5 concludes the whole study of Wavelet-DCT based image coder for video coding application and provides some recommendations.

CHAPTER 2
LITERATURE REVIEW

2.1. Wavelet Transform

A "wavelet" is a small wave, whose energy is concentrated around a point in time/scale domain. Wavelet is very useful as a tool to analyze transient, non-stationary, or time-varying phenomena on a signal or functions. Wavelet basis is taken to represent a signal in the same way as Fourier series uses the sinusoidal wave to represent a signal. The following are three main characteristics of wavelet:

1. Wavelet system is a set of building blocks to represent a signal or function by two-dimensional expansion set.
   The wavelet series expansion is denoted in terms of two indices, the translation index k and the scales index j. The value k enables the basis function to specify the signal at different points of time. While the value j stretches or shrinks the basis function and controls the resolution of the signal representation.
2. Wavelet expansion provides time/scale-frequency localization of the signal.
3. The calculation of the coefficients from the signal can be done efficiently (Multiresolution Analysis).

For a signal with continuous variable, the continuous wavelet transform (CWT) can be defined by:

$$F(a,b) = \int f(t)\psi_{a,b}(t)dt \qquad \psi_{a,b}(t) = a^{-1/2}\psi(\frac{t-b}{a}) \tag{2.1}$$

While Discrete Wavelet Transform (DWT) can be defined by

$$f(t) = \sum_{k=-\infty}^{\infty} c(k)\varphi(t-k) + \sum_{j=0}^{\infty}\sum_{k=-\infty}^{\infty} d_j(k)\psi(2^j t-k) \tag{2.2}$$

Scaling function is given by $\varphi(t) = \sum_{n} h(n)\sqrt{2}\varphi(2t - n), \ n \in Z$ (2.3)

Wavelet function is given by $\psi(t) = \sum_{n} h_1(n)\sqrt{2}\varphi(2t - n), \ n \in Z$ (2.4)

For image processing, two dimensional DWT is employed. The wavelet transform decomposes spatial image data according to its frequency row-wise and column-wise. A single-stage 2-D wavelet transformation consists of a filtering operation that decomposes an image into four frequency bands; CA_1, CH_1, CV_1 and CD_1. CA (scaling coefficients) contains lowpass coefficients for both row and column pixels. CH (horizontal) contains lowpass coefficients for row pixels and highpass coefficients for column pixels. CV (vertical) contains highpass coefficients for row pixels and low lowpass coefficients column pixels. CD (diagonal) contains highpass coefficients for both row and column pixels.

The decomposition process may be repeated for CA_j component to produce another set of four components (CA_{j+1}, CH_{j+1}, CV_{j+1}, CD_{j+1}).

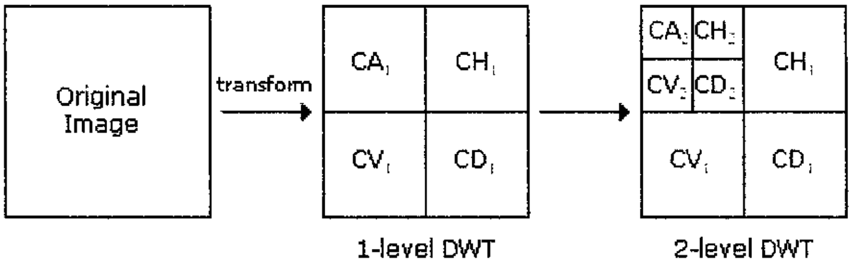


Figure 2.1 J-level 2-D Wavelet Decomposition

Multiresolution structure of 2-D wavelet transform is shown in Figure 2.2 Multiresolution structure of wavelet permits fast implementations to calculate the DWT coefficients.

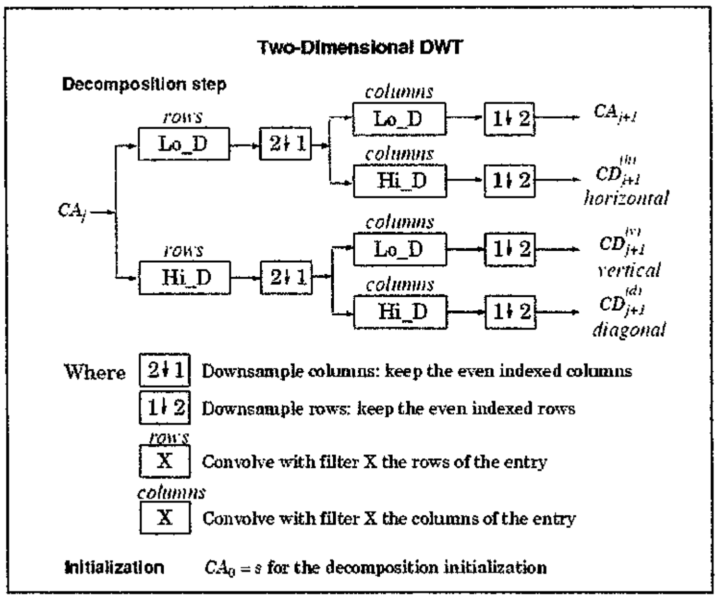

Two-Dimensional DWT
Decomposition step
rows
Lo_D
2↓1
columns
Lo_D
1↓2
CA_{j+1}
columns
Hi_D
1↓2
$CD_{j+1}^{(h)}$
horizontal
CA_j
rows
Hi_D
2↓1
columns
Lo_D
1↓2
$CD_{j+1}^{(v)}$
vertical
columns
Hi_D
1↓2
$CD_{j+1}^{(d)}$
diagonal
Where 2↓1 Downsample columns: keep the even indexed columns
1↓2 Downsample rows: keep the even indexed rows
rows X Convolve with filter X the rows of the entry
columns X Convolve with filter X the columns of the entry
Initialization $CA_0 = s$ for the decomposition initialization

Figure 2.2 Multiresolution Structure of Two-Dimensional DWT

2.2. Discrete Cosine Transform (DCT)

DCT has been the most popular transform for image and video coding because of its effectiveness at transforming image data into a form that is easy to compress and its efficiency on implementation to software and hardware. The forward DCT transforms a set of image samples into a set of transform coefficients. Then, the DCT of x(n) can be mathematically defined as:

$$C_x(k) = \alpha(k)\sum_{n=0}^{N-1} x(n)\cos\frac{(2n+1)k\pi}{2N} \qquad (2.5)$$

$$\text{where} \quad \alpha(0) = \frac{1}{\sqrt{N}}, \quad \alpha(k) = \sqrt{\frac{2}{N}}, \quad 1 \le k \le N-1 \qquad (2.6)$$

From equation (2.5), for $k = 0$, $C_x(0) = \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} x(n)$. Thus, the first transform coefficient is the average value of the sample sequence. This value is also referred to as the DC Coefficient. All other transform coefficients are called the AC coefficients [10].

The inverse DCT is shown as follow:

$$x(n)=\sum_{k=0}^{N-1}\alpha(k)C_x(k)\cos\frac{(2n+1)k\pi}{2N} \tag{2.7}$$

The DCT transforms a signal or an image from the spatial domain to the frequency domain. Normally, the input image is divided into block of 8×8 or 16×16 pixels and then transformed by using DCT. A two-dimensional DCT is used in image processing since image is 2-D in nature.

The 2-D DCT is a direct extension of the 1-D case and is given by

$$C_x(u,v)=\alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right), \tag{2.8}$$

For $u,v=0,1,2,\ldots,N-1$ and $\alpha(u)$ and $\alpha(v)$ are defined in (2.6).

The inverse 2-D transform is defined as

$$f(x,y)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\alpha(u)\alpha(v)C(u,v)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right), \tag{2.9}$$

For $x,y=0,1,2,\ldots,N-1$.

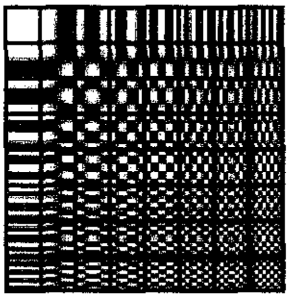


Figure 2.3 2-D DCT Basis Function for $N=8$

The histogram of transformed image coefficients (DCT coefficients) is regarded to follow the Laplacian probability density function (PDF). However, for Discontinuous Tone Image, the histogram of its DCT coefficients has wider *width* (refer to section 3.4).

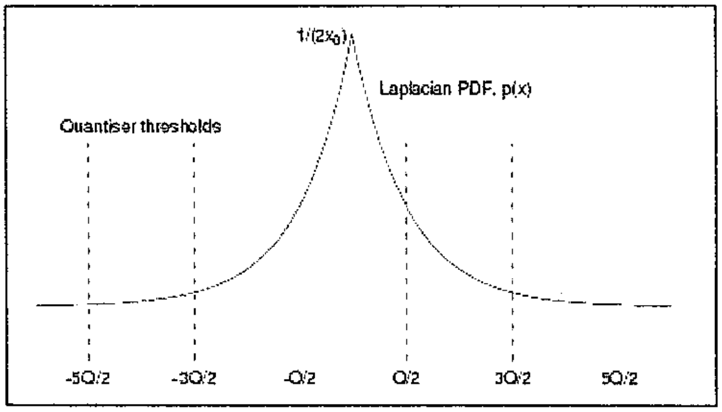


Figure 2.4 Laplacian Probability Density Function

2.3. Overview of MPEG2

Motion Picture Expert Group is a working group of ISO/IEC in charge of development of standards for coded representation of digital audio and video. MPEG-2 is the standard for the DVD-format, digital television (DVB) and the AAC audio format. However, this report concentrates on MPEG-2 part 2, ISO 13818-2 which defines the video coding standard.

2.3.1 Picture Reordering

Not all frames of a video sequence are coded in the same way. MPEG identified three types of frames in a video sequence.

- I - Frames – coded without the reference to previous picture.
- P - Frames – coded with reference to the previous I or P-coded picture.
- B - Frames – bidirectionally coded pictures in its prediction. B pictures are not used for predictions of future frames, and thus they can be coded with the highest possible compression.

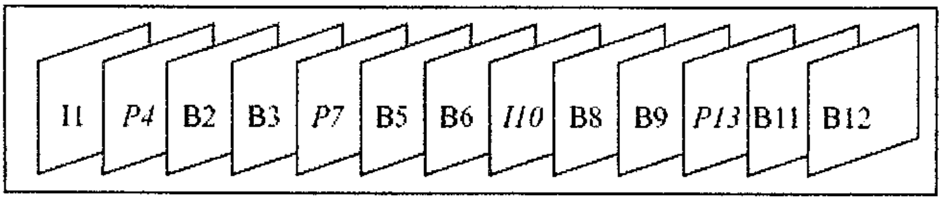


Figure 2.5 Transmission Ordered Sequence

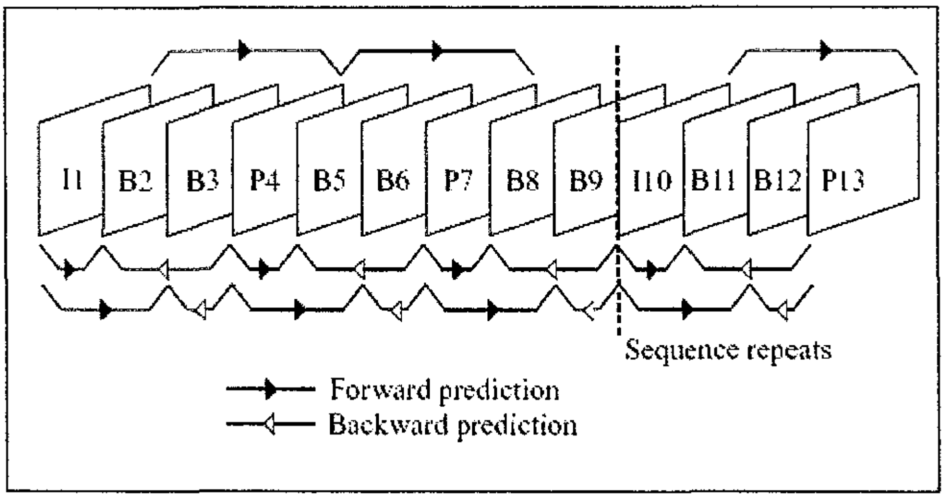


Figure 2.6 Display Ordered Sequence

2.3.2 Interlace Sequence Support

MPEG2 standard has an interlace sequence support. Interlacing is an old analogue compression technique. In MPEG-2 a progressive sequence one frame is one image. But in an interlaced sequence, a frame is actually two pictures interlaced together with odd and even scanlines.

2.3.3 Colorspace Conversion

In MPEG2 standard, the YCbCr colorspace representation is used instead of RGB representation. The colours are represented as luminance (Y), chrominance blue, (Cb) and chrominance red (Cr).

The equations to perform conversion from YCbCr to RGB:

$$R = Y + 1.402(Cr - 127) \tag{2.10}$$

$$G = Y - 0.34414(Cb - 128) - 0.71414(Cr - 127) \tag{2.11}$$

$$B = Y + 1.722(Cb - 127) \tag{2.12}$$

2.3.4 Block Compression

A typical video coder consists of three main sections; intra-frame coder, inter-frame coder, and lossless coder (run length coder and entropy coder).

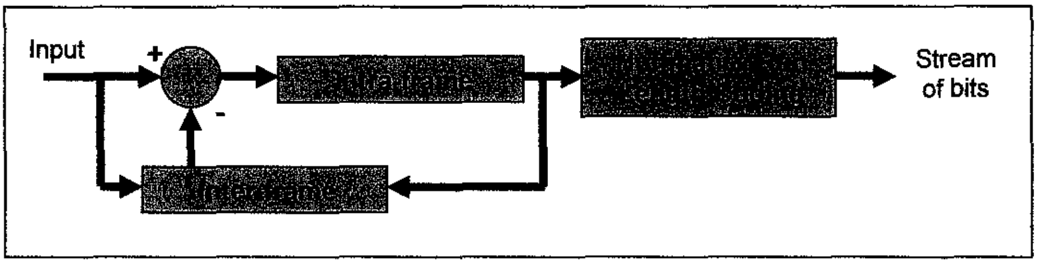


Figure 2.7 Simplified Block Diagram of a Typical Video Coder

An intra-frame coder works similar to the still image coder. A model of intra-frame encoder is shown in Figure 2.8.

The input image for intra-frame coder is broken down to blocks of 8x8, 16x16, or 8x16 (In this project, input image is always broken down to 16x16 blocks). 2-D DCT will be performed on each block, followed by quantization of DCT coefficients and entropy coding.

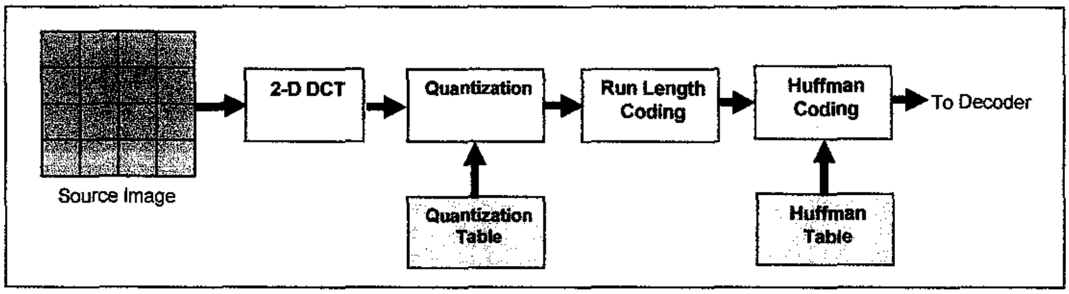


Figure 2.8 Conventional DCT based Intraframe Encoder

2.3.4.1 Two-Dimensional Discrete Cosine Transform

DCT is a transform domain coding which is mainly used to remove the spatial redundancies in images by mapping the pixels into a transform domain prior to data reduction. DCT has a good energy compaction property especially for natural image (Continuous Tone Image). Most of the time, the DCT coefficients have high values at low frequencies and near-zero values at high frequencies.

2.3.4.2 Quantization

The next operation is quantization of the 16x16 DCT coefficients using a carefully designed Quantization Table. This is where lossy compression happens. There are different Quantization Tables available for Continuous Tone Images (CTI) and Discontinuous Tone Images (DTI).

CTI quantization or intra quantization weighing matrix table has a variable step size for each coefficient in different spatial location (refer to Appendix A). This quantization table assigns different quantization step for each spatial location based on their degree of "importance". DTI quantization table or also known as inter quantization weighing matrix has a uniform step size. The quantized DCT coefficients are rounded to integer values to ensure efficient implementation on hardware.

After quantization, many coefficients are found to be zeros but only few numbers of non-zero coefficients which are concentrated on the top left corner. At the decoder, the quantized values are multiplied by the corresponding Quantization Table elements to recover the original dequantized values. However, quantization is a lossy process in which the dequantized values are very close to the original values but they are not exactly the same.

2.3.4.3 Entropy Coder

After quantization, zig-zag reordering of quantized coefficients is performed as shown in Figure 2.9.

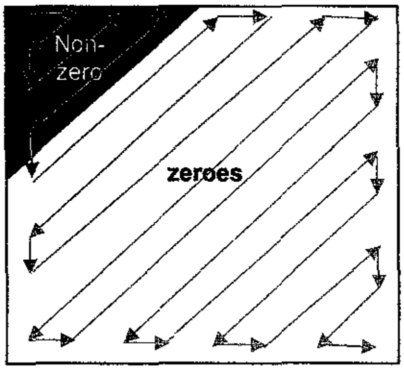


Figure 2.9 Zig-zag Sequence

This ordering helps to facilitate entropy encoding by placing low-frequency non-zero coefficients before high frequency coefficients. For natural images (CTI), its reordered coefficients are preceded by non-zero values, followed by mostly all zeros on the tail of the sequence. For difference images (DTI), non-zero values can be also found on the top left side with larger area compared to CTI. Thus, the resulting sequence is similar to CTI.

Upon zig-zag reordering, the coder achieves additional compression losslessly by encoding the quantized DCT coefficients based on their statistical characteristics. The baseline JPEG standard uses both Run-Length and Huffman Coding.

- Run-Length Coding (RLC)

  RLC is lossless method to compress sequential data. It works by counting the number of adjacent elements/pixels with the same intensity level. Run-Length Coding can be easily implemented in both software and hardware. However, its compression ability is very limited.

- Huffman Coding

  Huffman Coding is also a lossless compression technique that exploits the input probabilities for the image. It is based on the fact that in an input stream certain tokens occur more other than others

2.4. DWT-DCT Algorithm

DWT-DCT algorithm is a fast-DCT algorithm which utilizes DWT to compute DCT. To better appreciate the concept of this algorithm, it is necessary to review the fast Fourier transform algorithm.

2.4.1 Reviewing DFT and FFT

A fast Fourier transform (FFT) is an efficient algorithm to compute the discrete Fourier transform (DFT) and it's inverse. The sequence of N complex numbers $x_0,\ldots,x_{N-1}$ is transformed into the sequence of N complex numbers $X_0,\ldots,X_{N-1}$ by the DFT according to the formula:

$$X[k]=\sum_{n=0}^{N-1}x[n]W_N^{kn} \qquad ,k=0,1,2,\ldots,N-1 \tag{2.13}$$

$$W_N=e^{-j2\pi/N} \tag{2.14}$$

DFT can be efficiently calculated using different FFT algorithms, one way is using Radix-2 Decimation in Time (DIT) variant of FFT.

Figure 2.10 shows the block diagram of the last stage of length-8 Radix-2 DIT FFT. The input data are separated into even and odd groups before going through a length-4 DFT block.

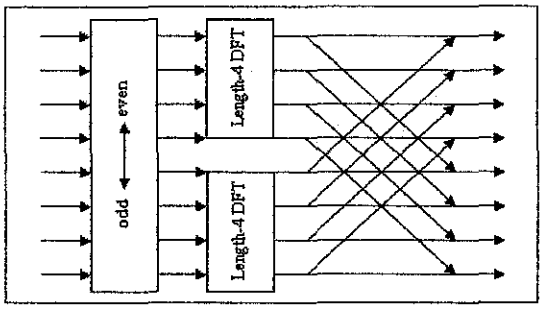


Figure 2.10 Block Diagram Illustrating Decimation-In-Time (DIT) FFT

2.4.2 DWT-based DCT and Its Approximation

The structure of DWT based DCT is very similar to the classical FFT. Figure 2.11 shows the block diagram of the last stage of the length-8 DWT-based DCT. This diagram is an equivalent comparison to Figure 2.11.

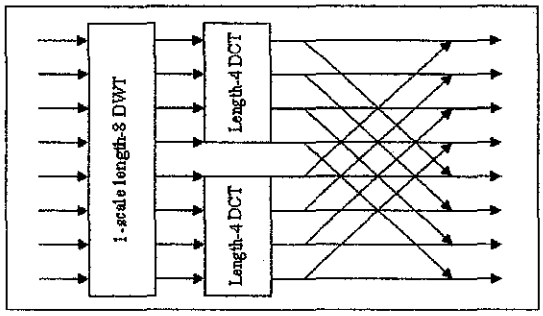


Figure 2.11 Block Diagram Illustrating Wavelet-DCT

The matrix factorization for DCT using Haar/Daubeschies1 wavelet can be written as [3]:

$$C_N = B_N \begin{bmatrix} C_{N/2} & 0 \\ 0 & C_{N/2} \end{bmatrix} \begin{bmatrix} I_{N/2} & 0 \\ 0 & (-1)^n I_{N/2} \end{bmatrix} H_N \quad , n \in \{0,1,\ldots,N/2-1\} \qquad (2.15)$$

Where C_N is the $N \times N$ matrix, H_N is the $N \times N$ matrix 1 level Haar wavelet matrix, I_N is the $N \times N$ identity matrix, and B_N is the $N \times N$ cross diagonal butterfly operation matrix defined as:

$$B_N = \left[B^H \quad B^L \right] \tag{2.16a}$$

$$B^H = \begin{cases} \cos\left(\dfrac{\pi k}{2N}\right) & 0 \le k \le \dfrac{N}{2} - 1 \\ 0 & k = \dfrac{N}{2} \\ -\cos\left(\dfrac{\pi k}{2N}\right).P & \dfrac{N}{2} + 1 \le k \le N - 1 \end{cases} \tag{2.16b}$$

$$B^L = \begin{cases} \cos\left(\dfrac{\pi(N-k)}{2N}\right).P & 0 \le k \le \dfrac{N}{2} - 1 \\ 1 & k = \dfrac{N}{2} \\ \cos\left(\dfrac{\pi(N-k)}{2N}\right) & \dfrac{N}{2} + 1 \le k \le N - 1 \end{cases} \tag{2.16c}$$

The approximation of DWT based DCT comes when thresholding scheme is employed. Thresholding happens to the DWT coefficients in which the values that are considered "unimportant" can be threshold by setting them to zeros. Hence, the DCT coefficients obtained at the far end is an approximation. When threshold level is equal to zero, the result is the same like a normal DCT coefficients.

The implementation of this fast DCT is apparently equivalent to normal fast DCT. On top of it, the intermediate thresholding capability increases the speed of this algorithm by approximating the DCT coefficients. Thus, the algorithm is also known as Fast Approximate DCT. The art of it is to selectively decide the threshold level that can result in a much faster process and still retrieving an acceptable approximation.

CHAPTER 3
METHODOLOGY

This chapter includes the elaboration of proposed techniques to achieve objectives of the study. It also covers the steps and methods taken throughout the whole project and the system performance parameter.

3.1. Proposed Techniques

This section elaborates the techniques of Wavelet-DCT algorithm, threshold level selection and wavelet selection.

3.1.1 Wavelet-DCT Algorithm

Wavelet-DCT is a new way to compute Fast DCT via wavelet transform.

3.1.1.1 2-D Wavelet-DCT

It is necessary to perform thresholding on 2-D wavelet coefficients The 2-D Wavelet-DCT block developed is a modification of the 1-D Wavelet-DCT which is performed twice (row-wise then followed by column-wise).

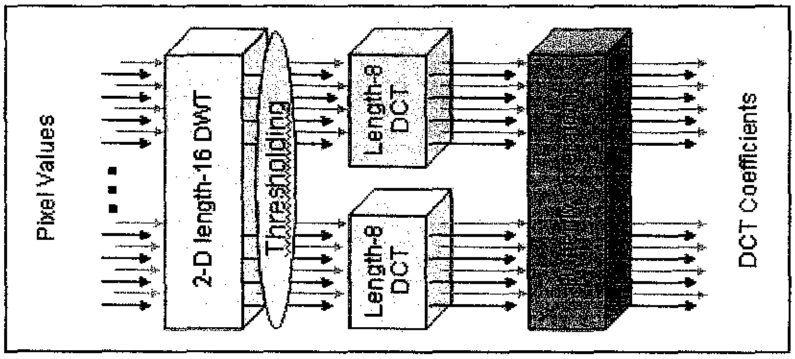


Figure 3.1 Block Diagram Illustrating 2-D Wavelet-DCT

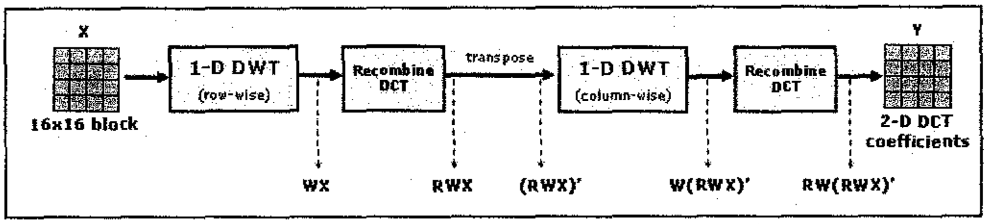


Figure 3.2 Development of 2-D Wavelet-DCT from 1-D Wavelet-DCT

X = matrix of input image

W = Haar Wavelet matrix

R = recombining matrix = $B_N*C_{N/2}*I_{by}$ (refer to equation 2.15)

D = 2-D Haar Wavelet matrix

Y = 2-D length-N DCT coefficients matrix

$$\begin{aligned} Y &= RW(RWX)' \\ &= RWX'W'R' \\ &= R(WX'W')R' \qquad (3.1) \end{aligned}$$

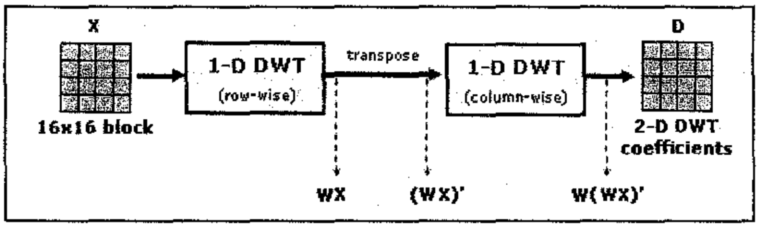


Figure 3.3 2-D DWT obtained from 1-D DWT

$$\begin{aligned} D &= W(WX)' \\ &= WX'W' \qquad (3.2) \end{aligned}$$

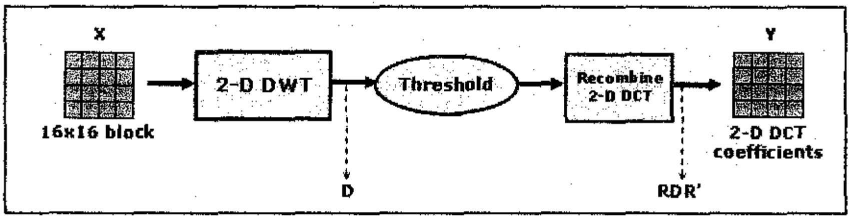


Figure 3.4 Development of 2-D Wavelet-DCT

Substitution (3.2) into (3.1) gives: $Y = RDR'$ (3.3)

3.1.1.2 J-level 2-D Wavelet-DCT

An attempt was done to include J-level 2-D DWT within the computation. The Wavelet-DCT algorithm is further expanded. The 2-level Haar wavelet transform can be written in the following matrix:

$$\begin{bmatrix} H_{N/2} & \\ & I_{N/2} \end{bmatrix} H_N \tag{3.4}$$

The Fast DCT using 2-level DWT can be written in the following equation:

$$C_N = B_N \begin{bmatrix} B_{N/2} & \\ & I_{N/2} \end{bmatrix} \begin{bmatrix} C_{N/4} & & & \\ & C_{N/4} & & \\ & & C_{N/4} & \\ & & & C_{N/4} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} I_{N/4} & \\ & (-1)^n I_{N/4} \end{bmatrix} & \\ & (-1)^n I_{N/2} \end{bmatrix} \begin{bmatrix} H_{N/2} & \\ & I_{N/2} \end{bmatrix} H_N \tag{3.5}$$

3.1.1.3 J-level 2-D Wavelet Packet-DCT

Wavelet packet system allows a finer and adjustable resolution of frequencies at high frequency. Input data are first filtered by pair of filter **h** and **g** (lowpass and highpass respectively) and then downsampled. The same analysis is further iterated on both low and high frequency bands.

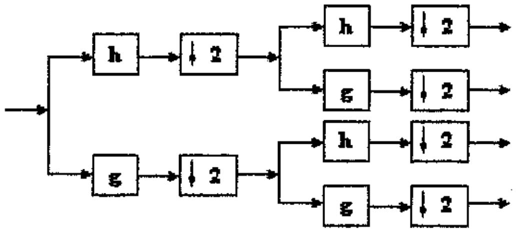


Figure 3.5 The full binary tree for 2-Level Wavelet Packet Transform

The 2-level Haar wavelet transform can be written in the following matrix:

$$\begin{bmatrix} H_{N/2} & \\ & H_{N/2} \end{bmatrix} H_N \tag{3.6}$$

The Fast DCT using 2-level DWT can be written in the following equation:

$$C_N = B_N \begin{bmatrix} B_{N/2} & \\ & B_{N/2} \end{bmatrix} \begin{bmatrix} C_{N/4} & & & \\ & C_{N/4} & & \\ & & C_{N/4} & \\ & & & C_{N/4} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} I_{N/4} & \\ & (-1)^n I_{N/4} \end{bmatrix} & \\ & (-1)^n I_{N/2} \end{bmatrix} \begin{bmatrix} H_{N/2} & \\ & H_{N/2} \end{bmatrix} H_N \tag{3.7}$$

3.1.2 Threshold Level Selection Algorithm

Wavelet-DCT owns an intermediate-thresholding capability. When thresholding of wavelet coefficients takes place, the result of it is an approximation of DCT coefficients. The question is how to determine the correct threshold level (TL) which led to a better system performance.

3.1.2.1 Investigation on Singularity Detection Using Wavelets Modulus

On an image, the discontinuities of intensity values indicate the location of object contours/edges. These singularities, which can be characterized by Lipschitz exponent, are necessary to be preserved because singularity contains "important" information especially in difference images that is rich in high frequency.

[2] has proved that the local maxima of the wavelet transform modulus detect the locations of singularities and they also provide numerical procedures to compute their Lipschitz exponent (the order α). The order α can give the understanding of how much information is carried by the local maxima of a wavelet transform modulus. Meyer proved [ref] that the local maxima of wavelet transform modulus do not characterize uniquely a function. However a numerical algorithm is able to reconstruct a close approximation of the original signal.

$$|Wf(s,x)| \leq A_\varepsilon s^\alpha \tag{3.8}$$

Initially, the attempt was to obtain the Lipschitz α in order to locate and characterize the singularities of a difference image. Both the location and the order α were to be used to compute the energy. They were expected to provide information to determine the apt threshold level for the Wavelet-DCT based intra-frame coder.

However, the theory of singularity detection with wavelet and the numerical procedures to compute Lipschitz exponent requires a higher degree of understanding in wavelet transform and mathematics.

3.1.2.2 Histogram Analysis

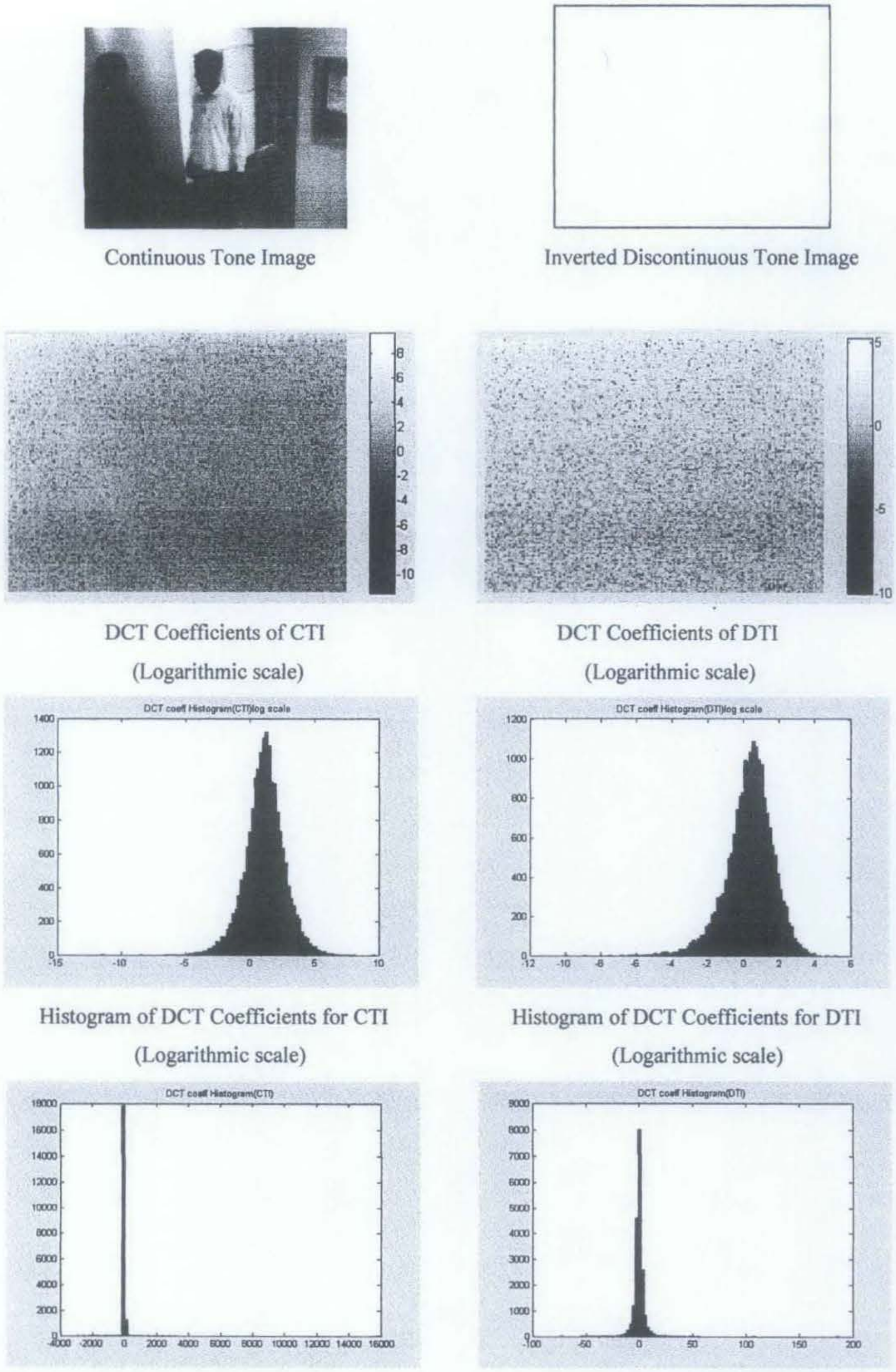
Histogram is a statistical representation of a set of data in a form of a bar chart where each entry/bin in horizontal axis corresponds to the coefficient value and the vertical axis correspond to the number of occurrence.

The investigation of both histograms of DCT and wavelet coefficients led to the development of Signal Energy Analysis on Histogram of wavelet coefficients. The technique of signal energy analysis is then employed on Threshold Level (TL) selection and best wavelet type selection.

3.1.2.2.1 Histogram of DCT Coefficients

Most image and video coding standard use a block based 2-Dimensional DCT or an approximation to it as part of the coding algorithm. The knowledge of the statistical behaviour of the DCT coefficients is important in the design of such encoder algorithms. The distribution of DCT coefficients is conjectured to have Gaussian, Laplacian or more complex distribution. Among these, the Laplacian distribution is probably the most popular, and used in practice [6].

The following figures show the distribution of DCT coefficients for both Continuous Tone Image (CTI) and Discontinuous Tone Image (DTI).



Continuous Tone Image

Inverted Discontinuous Tone Image

DCT Coefficients of CTI
(Logarithmic scale)

DCT Coefficients of DTI
(Logarithmic scale)

Histogram of DCT Coefficients for CTI
(Logarithmic scale)

Histogram of DCT Coefficients for DTI
(Logarithmic scale)

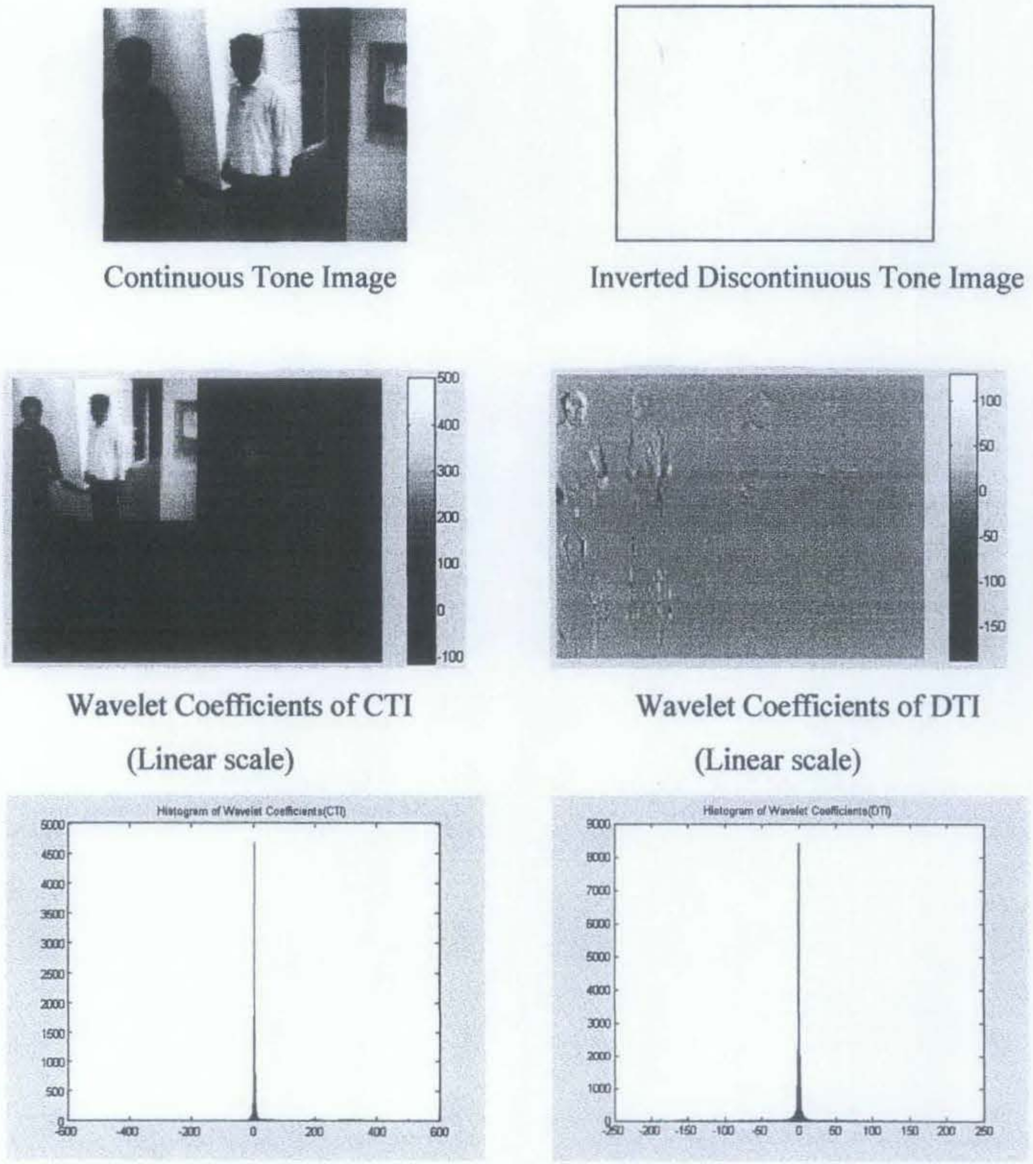
Histogram of DCT Coefficients for CTI
(Linear scale)

Histogram of DCT Coefficients for DTI
(Linear scale)

Figure 3.6 Histogram of DCT Coefficients

3.1.2.2.2 Histogram of Wavelet Coefficients

The following figures show the distribution of Wavelet coefficients for both Continuous Tone Image (CTI) and Discontinuous Tone Image (DTI) using Haar wavelet.



Continuous Tone Image

Inverted Discontinuous Tone Image

Wavelet Coefficients of CTI (Linear scale)

Wavelet Coefficients of DTI (Linear scale)

Histogram of Wavelet Coefficients (CTI) (Linear scale)

Histogram of Wavelet Coefficients (DTI) (Linear scale)

Figure 3.7 Histogram of Wavelet Coefficients

3.1.2.2.3 Signal Energy Analysis on Histogram

The attempt to determine the apt threshold level that achieves as much compression as possible while restraining the error brought the research into signal energy analysis of the image. Retained energy signal is closely related to the error. Consider the following set of data/coefficients:

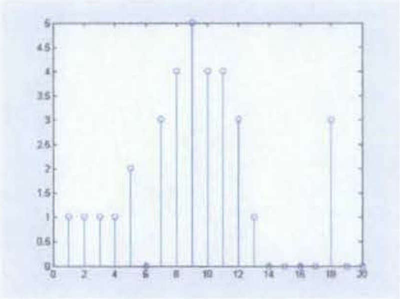


Figure 3.8 Plot of matrix A

Signal energy can be computed using the following formula:

For continuous signal: $E_t = \int_{-\infty}^{\infty} | a(t) |^2 \, dt$ (3.9)

For discrete signal: $E_t = \sum_{i=0}^{N} | A(n) |^2$ (3.10)

Et = $(1)^2 + (1)^2 + (1)^2 + (1)^2 + (2)^2 + (0)^2 + (3)^2 + (4)^2 + (5)^2 + (4)^2 + (4)^2 +$
$(4)^2 + (3)^2 + (1)^2 + (0)^2 + (0)^2 + (0)^2 + (0)^2 + (0)^2 + (3)^2 + (0)^2 + (0)^2$

Et = 109

Histogram of the signal A is shown below:

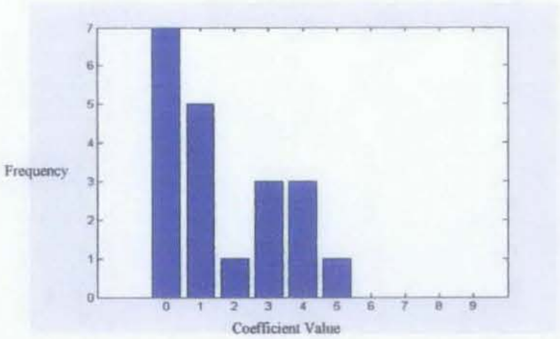


Figure 3.9 Histogram of matrix A

Energy contribution of each coefficient can be computed as:

Table 3.1 Energy Contribution per Each Value

Coefficient Value	Energy Contribution
0	$7 \times (0)^2 = 0$
1	$5 \times (1)^2 = 5$
2	$1 \times (2)^2 = 4$
3	$3 \times (3)^2 = 27$
4	$3 \times (4)^2 = 48$
5	$1 \times (5)^2 = 25$
Total	109 = Et

The author assumes that it is necessary to retain at least 90% of the total signal energy. Therefore, the remaining 10% of the total signal energy will be dropped by setting the values to zero. The coefficients to be threshold are the coefficients whose values are near zeros. Thus, thresholding is basically performing an approximation to these near-zero values.

The first 10% energy contribution is cumulatively computed from the smallest coefficient value:

Table 3.2 Cumulative Energy Approximated

Drop coefficients with value	Cumulative energy approximated
0	0/109 = 0%
0,1	(0+5)/109 = 4.6%
0,1, 2	(0+5+4)/109 = 8.26%
0,1, 2, 3	(0+5+4+27)/109 = 33%

From the table, a threshold level (TL = 3) is determined. That is, any coefficients with value less than the threshold level will be set to zero. In this case, after thresholding all coefficients with the value below TL = 3, the energy retained is 100% - 8.26% = 91.74%.

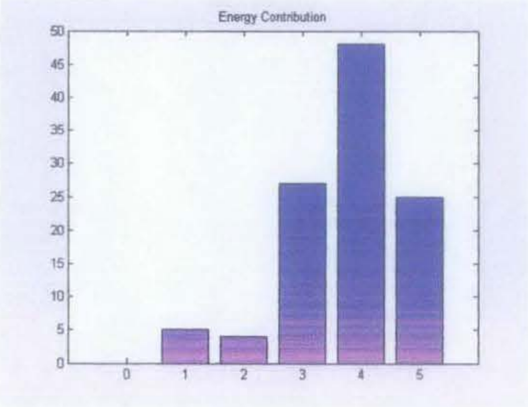


Figure 3.10 Histogram Energy Contributions

Since wavelet transform is able to represent signals sparsely, the selected percentage to be retained is chosen to be 99.9%.

3.1.3 Wavelet Selection

Histogram energy analysis is used to determine the best wavelet types to represent difference images (DTI). The wavelet employed in the previous study was merely Haar wavelet. An investigation was done to determine which wavelet type is the best to represent a difference image.

The best wavelet is wavelet that can represent a DTI with less high-valued coefficients and more near-zeros-valued coefficients (smaller *width*). Hence, more pels can be threshold to zero, and therefore more compression.

Wavelet selection was done by statistical approach. From each wavelet coefficients, it is observed which wavelet is able to represent 90%, 80%, and 70% energy signal with the least number of coefficients.

DWT coefficients using various type of wavelet were obtained. Some wavelet types that are tested are: *db1*, *db2*, *db3*, *db4*, *sym2*, *sym3*, *sym4*, *coif1* and *coif2*.

The observation was performed on 420 blocks (16x16 block). Preserving 90% of total energy in each block, 288 blocks (68.57%) are best represented by Haar/db1 wavelet. Preserving 80% of total energy, 246 blocks (58.57%) of all 420 blocks are best represented by Haar/db1 wavelet. And 244 blocks (58.09%) is best represented by Haar/db1 wavelet, preserving 70% of total energy. The complete statistical result can be found in Appendix B.

3.2 Project Methodology

This section discusses the steps and methods taken to code the Wavelet-DCT algorithm and histogram energy analysis, build the MPEG-2 simulation and select the types of wavelet.

3.2.1 Wavelet-DCT code Development

Rewriting simplified form of equation (2.15),

$$C_N = B_N C_{N/2} I_{by} H_N \quad ,n \in \{0,1,\ldots,N/2-1\} \tag{3.11}$$

H_N, the 1-level Haar Wavelet matrix of length-16 is obtained from the forward DWT function "fwdDWTN.m" from [1], [5]. A half size DCT matrix $C_{N/2}$ is coded by borrowing MATLAB function "dctmtx.m". The permutation matrix I_{by} is coded in m-file using "eye(N)" function. The butterfly matrix B_N is written in m-file based on equations (2.16). Verification of equation (3.2) is done by taking the summation of absolute error between the result of Wavelet-DCT and the actual DCT coefficients obtained from dctmtx(16). These codes are compiled in a function called "fadct.m". Based on equation (3.11), recombination matrix is coded in an m-file named "recombination.m"

From equation (3.3), it is known that 2-D DCT coefficients *(Y)* can be obtained from computation of 2-D DWT coefficients *(W)*. The codes to obtain 2-D DCT coefficients through 2-D DWT coefficients are written in m-file function named "fadct2.m". All components are obtained from the same way like "fadct.m" except for the 2-D Wavelet matrix which is obtained from function "fwdDWTN2.m". Function "fwdDWTN2.m" computes 2-D wavelet coefficients and it is developed based on equation (3.2).

Based on equation (3.6), the m-file function "fwdDWTN2.m" is then equipped with J-level 2-D wavelet-transform within the computation. The m-file function "fwdDWPTN2.m" is also developed based on equation (3.8) to allow access to J-level Wavelet Packet coefficients. It is then integrated into the fast DCT algorithm in the m-file function "fadctWP2.m".

The Wavelet-DCT algorithm is then integrated into the simulation of intra-frame coder in SIMULINK using embedded-MATLAB function block.

3.2.2 Histogram Analysis and Thresholding code Development

In the implementation on block compression for intra-frame coder, the optimum percentage of energy retained is chosen to be 99.9%. The signal energy analysis through histogram is coded in m-file function "histA.m" with its inputs are the wavelet coefficients matrix and percentage of energy dropped. The function "histA.m" results in a value of Threshold Level (TL). Then, TL value is fed into the m-file function "threshold.m" to perform hard-thresholding on matrix of wavelet coefficients.

3.2.3 Data Generation

Multimedia file like natural still image is a type of CTI. This type of images can be inputted to the system directly. However, a difference image (DTI) is obtained by subtraction of one frame with its reference frame. The two frames from a video file can be obtained by using MATLAB code, as documented in Appendix E. The video file with reordered frames can be obtained from the MATLAB code "frame_reordered.m"

3.2.4 Development of Intra-frame Coder in SIMULINK

SIMULINK and MATLAB are the tools used to develop a full intra-frame video coding simulation. The initial step was started from the Video and Image Processing Blockset for video and image compression in SIMULINK demo. From which, the simulation of intra-frame video coding was then developed. The intraframe coder was built and simulated as illustrated in Figure 3.10.

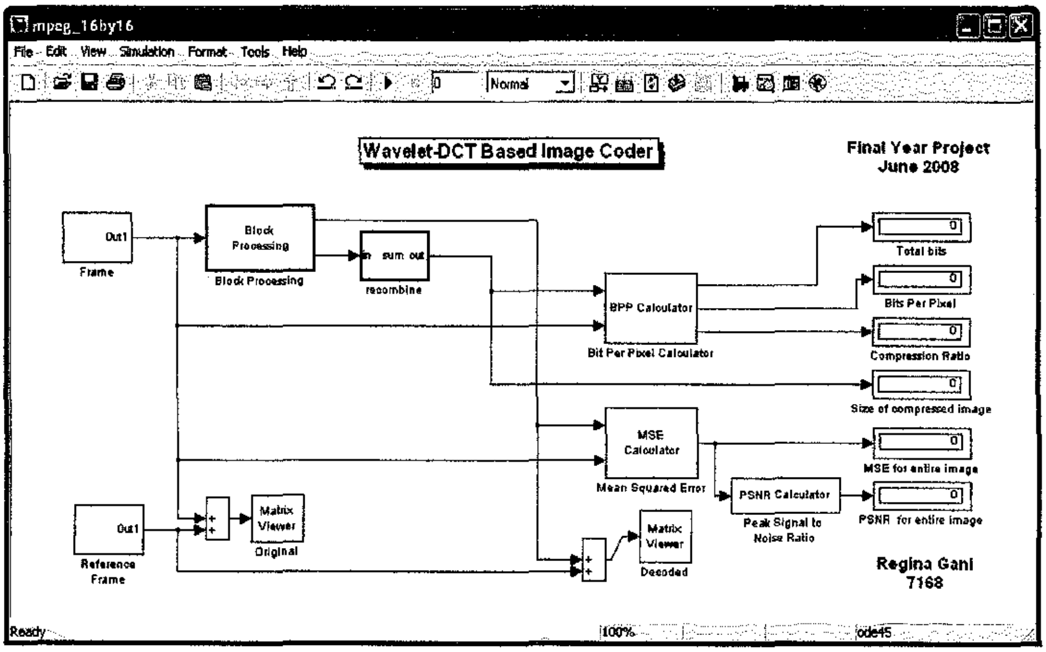


Figure 3.11 Intra-frame Coder as Image coder

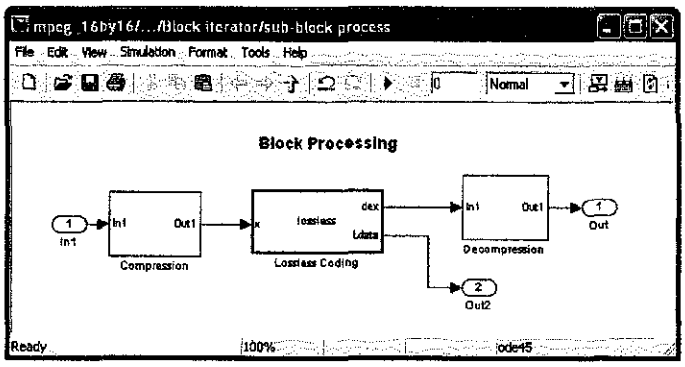


Figure 3.12 Block Processing

The compression of 16x16 blocks (encoder, Run-Length, Huffman coding, and decoder) takes place inside the block processing. The matrix viewer "Original" visualizes the input multimedia file while the decoded output is displayed on the matrix viewer "Decoded".

This project attempts to simulate the Intra-frame coder of MPEG2 standard. There are 3 types of frames; Intraframes (I-frames) and Non-Intra Frames (P-frames and B-frames).

First, the RGB colour space is represented by YCbCr. In the PAL system, the luminance bandwidth is normally 5MHz and the bandwidth of each colour component is only 1.5MHz, because the human eye is less sensitive to colour resolution [7]. Therefore, the input for intra-frame coder is the luminance of a difference image which is then broken into blocks of 16x16. For video input, the picture reordering and colorspace conversion are coded in an m-file function "frame_reorder.m". Then the input video is fed into the Multimedia Source subsystem.

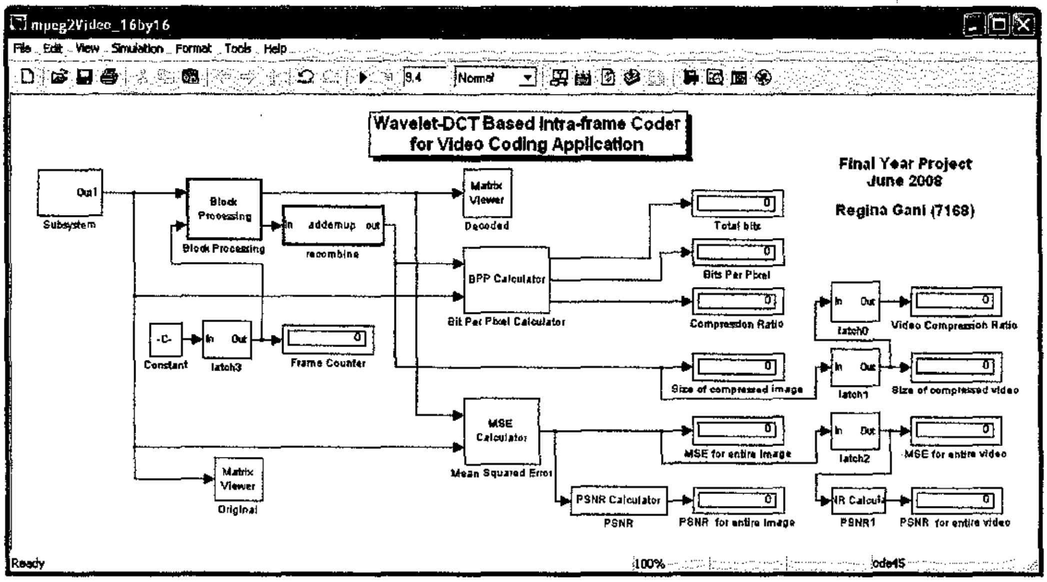


Figure 3.13 Intra-frame Coder for Video Input

3.2.4.1 The Encoder

Figure 3.12 shows the blocks within the encoder which includes 2-D DCT, quantization and zigzag scanning.

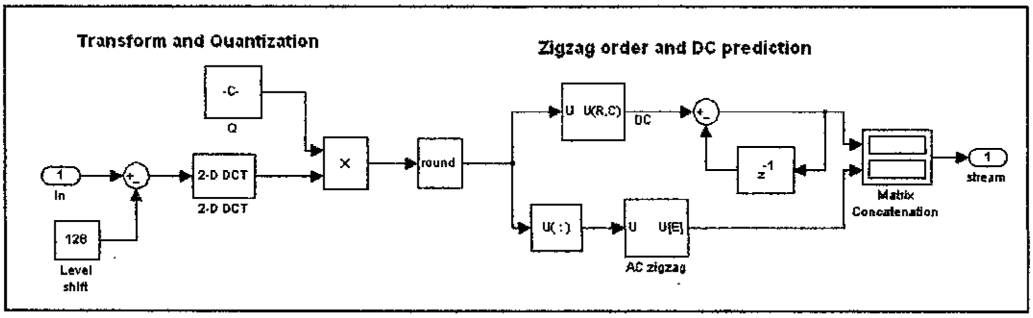


Figure 3.14 DCT based Intra-frame Encoder

The pixel intensity values normally lie in the range of 0-255. If this block is directly passed to 2-D DCT block, the AC coefficients will have a range of -1023 to +1024 and DC coefficient will range between 0 to 2040 which requires 11 bit signed integer [7]. Therefore, prior to DCT, the intensity values are subtracted by 128 which then shifts the DC coefficient in the range of 0 to 255.

The DCT coefficients are quantized using standard quantization table (Intra Quantization Table for CTI and Inter Quantization Table for DTI) and then rounded to ease the computation with only integer values. Next, zigzag reordering is performed on the AC coefficients using vector selector block which is then concatenated with the DC value to form a long sequence of integers.

3.2.4.2 Lossless Coding

Run-Length Coding and Huffman Coder are implemented into SIMULINK using an embedded-MATLAB function. The code for Run-Length coding “rle.m” is borrowed from [11]. The modified code for Huffman Coding “enHuff06.m” and “deHuff06.m” are borrowed from [1], [12]. Both encoding and decoding process are performed in the Lossless Coding block.

3.2.4.3 The Decoder

The decoder is an inverse process of its encoder. The long sequence of integers are reordered (de-zigzag scanning), multiplication with the respective Quantization table, 2-D IDCT and addition of pixel intensity values with constant 128.

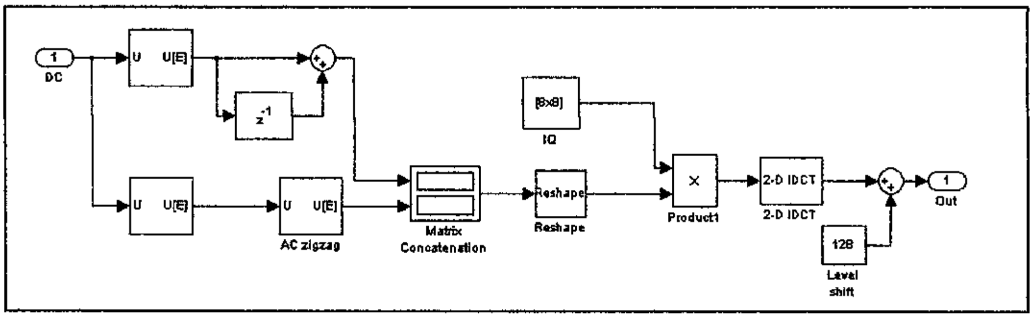


Figure 3.15 DCT based Intra-frame Decoder

3.2.4.4 System Performance Measurement Block

SIMULINK blocks are used to construct the system performance parameters based on section 2.5, equation (3.12), (3.13), (3.14), (3.15). CR/BPP calculator indicates the compression performance while MSE/PSNR calculator indicates the quality of reconstructed image.

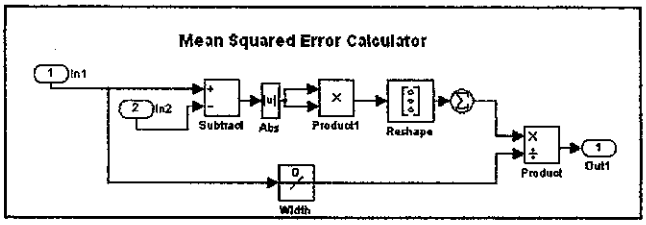


Figure 3.16 Mean Squared Error Calculator

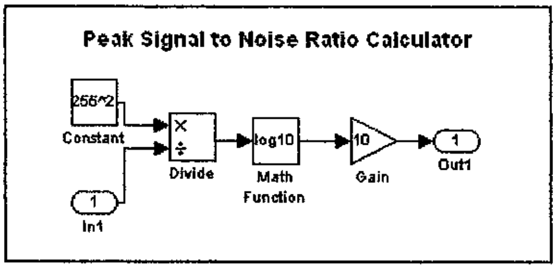


Figure 3.17 Peak Signal-to-Noise Ratio Calculator

3.2.4.5 Simulation Procedures

Multimedia file (image or video source) is inputted into the system to perform intra-frame coder compression. To perform still image compression, "mpeg_16by16.mdl" is run. For CTI, set the input for reference frame to "black.bmp" (matrix of all zeros). For DTI, set the input for reference frame to the respective reference frame.

To perform intra-frame video compression, "mpeg2Video_16by16.m" is run. The frames of video file are first reordered using an m-file "frame_reorder.m" which is then inputted to the system. It is important to ensure that input multimedia source is dividable by 16x16 block.

3.2.5 Development of Wavelet-DCT based Intra-frame Coder in SIMULINK

Two techniques are proposed to integrate wavelet-based compression on the intra-frame coder. The first approach is to replace the 2-D DCT block with Wavelet-DCT block and equip it with Histogram Analysis block that automatically determines the TL for each 16x16 block. The second approach is to perform a wavelet-based compression outside the intra-frame coder.

3.2.5.1 Wavelet-DCT block Replacing DCT block

The Wavelet-DCT blocks employs embedded-MATLAB function block. The simulation includes 2-D DWT (fwdDWTN2.m), Histogram Energy Analysis (histA.m), Thresholding (threshold.m) and Recombination (recombine.m) to obtain DCT coefficients.

To choose the threshold level (TL), Histogram Energy analysis is performed to every 16x16 block of DWT coefficients. The energy preserved is chosen to be 99.9% of total energy within the block. After TL is obtained, the DWT coefficients are pruned using hard thresholding scheme. For most cases, the low frequency subband of DWT coefficients is always preserved due to its high valued coefficients.

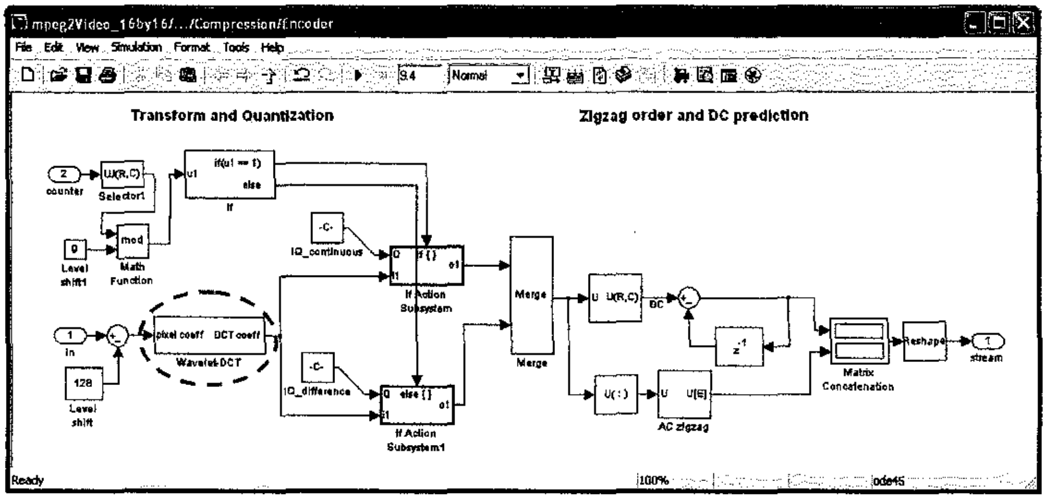


Figure 3.18 Wavelet-DCT based Intra-frame Encoder (Video)

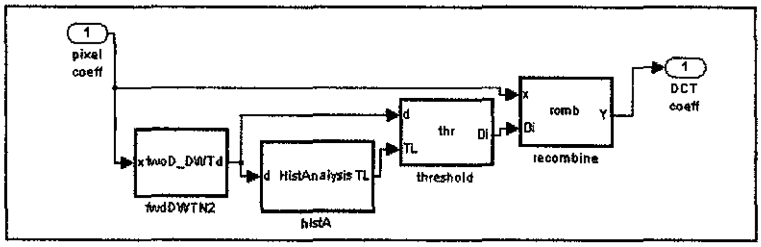


Figure 3.19 Wavelet-DCT Block

Since the reordered video frames consist of both CTI and DTI, both type of quantization table are available in the coder depending which frame is processed. Even though the encoder employs Wavelet-DCT algorithm, it does not change the decoder part complexity as the algorithm provide the backward compatibility to the existing architecture.

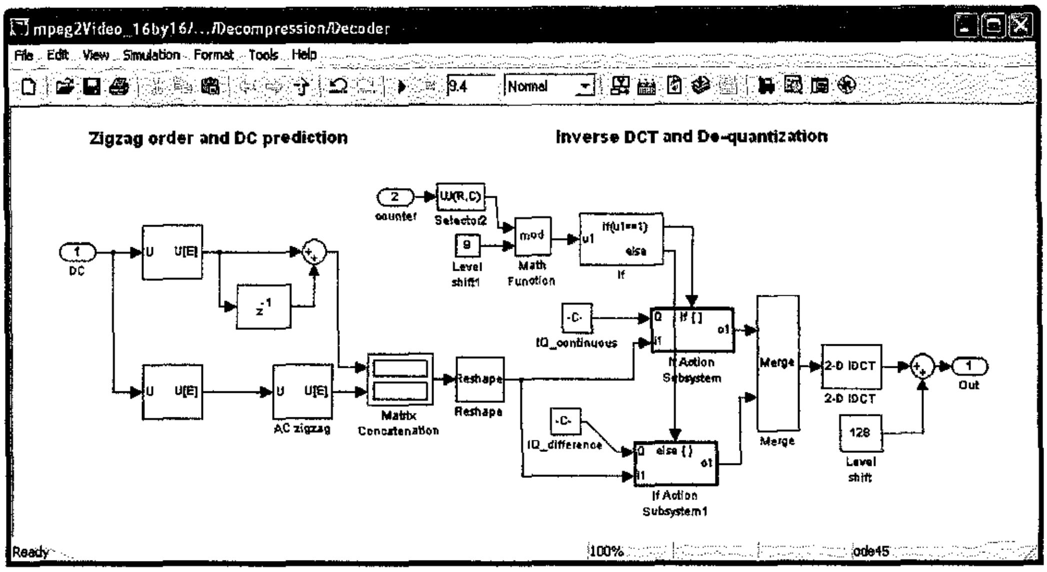


Figure 3.20 Wavelet-DCT based Intra-frame Decoder (Video)

3.2.5.2 Intra-frame Coder with DWT and IDWT preamble

DCT based intra-frame coder with preceding DWT&IDWT is basically a conventional intra-frame coder preceded by DWT performed on the whole image, thresholding of DWT coefficients, and IDWT to obtain back the scale domain coefficients.

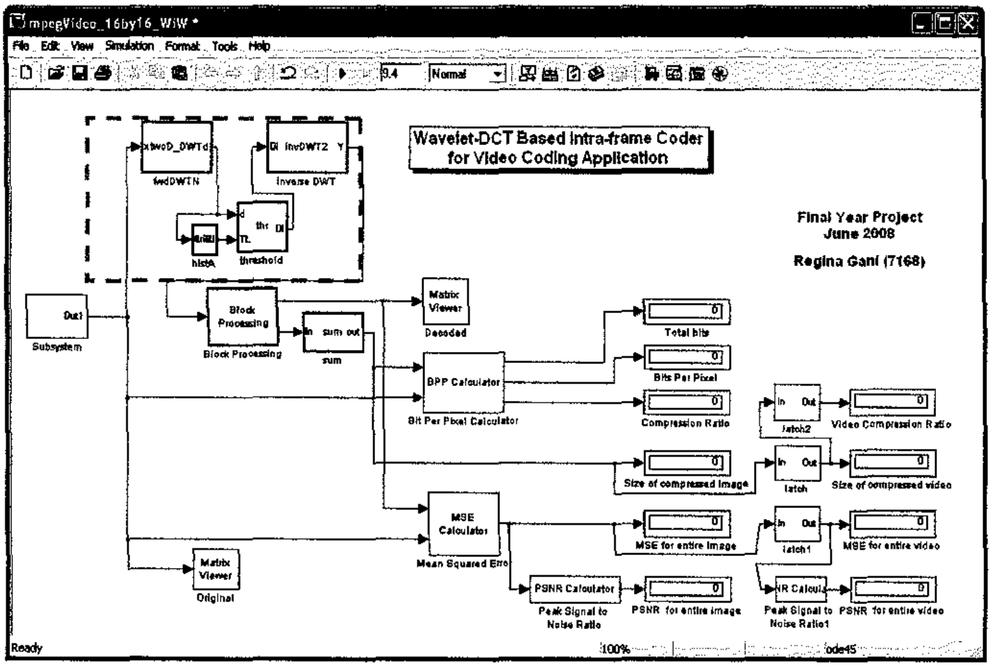


Figure 3.21 Inter-frame Coder with DWT&IDWT Preamble

3.3 System Performance Parameters

There are some important parameters used to quantify the performance of the intra-frame coder. The parameters are: Bit Rate/ Bit per pixel (BR/BPP), Compression Ratio (CR), Mean Squared Error (MSE), and Peak Signal-to-Noise Ratio.

BR is the average number of bits per each pixel in the encoded image, while CR is the compression ratio obtained from the encoding process. Both parameters can be mathematically represented as:

$$BR = \frac{N_c}{N_o}(8bits) \qquad (3.12)$$

$$CR = \frac{N_o}{N_c} \qquad (3.13)$$

Where N_c is the number of byte in the compressed sequence while N_o is the number of byte in the original image.

MSE is the square of the error of the output, which makes the output differ from the original value. PSNR is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Both parameters are used as a measure of quality for reconstruction in image compression. They are mathematically represented as:

$$MSE = \frac{1}{mn}\sum_{i-0}^{m-1}\sum_{j=0}^{n-1}\|I(i,j)-K(i,j)\|^2 \qquad (3.14)$$

$$PSNR = 10\log_{10}\left(\frac{\max\{I\}^2}{MSE}\right) = 20\log_{10}\left(\frac{\max\{I\}}{\sqrt{MSE}}\right) \qquad (3.15)$$

Where *m* is image vertical length and *n* is its horizontal length. *I* is the original image while *K* is the decoded image. *max{I}* is the maximum pixel value which is 255, since a pixel is represented by 8-bit value, ranging from 0 to 255.

CHAPTER 4

RESULTS AND DISCUSSION

This chapter shows the outcome of the simulation conducted in accordance to methodologies discussed in the previous chapter. The simulation was carried out for CTI, DTI and video input each using two proposed approach. The discussion and analysis are based on the system performance parameters; compression ratio and PSNR for image quality.

1st Approach:

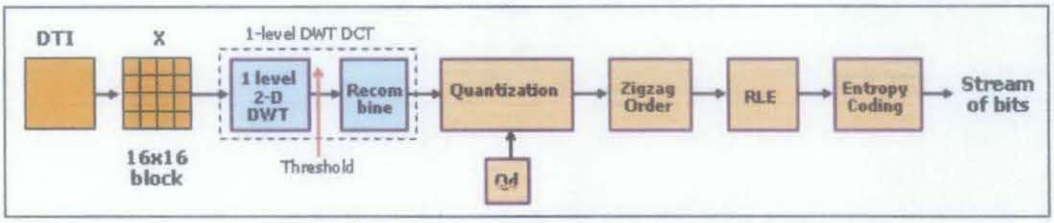


Figure 4.1 Wavelet-DCT block Replacing DCT-block

2nd Approach:

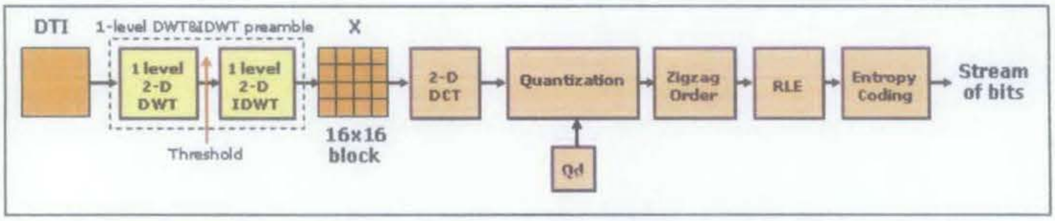


Figure 4.2 Intra-frame Coder with DWT and IDWT preamble

4.1. Result and Discussion for CTI Input

The intra-frame coders using 2 different approaches were tested using few standard test images from [13]. Besides the standard test images, frames from video vipmen.avi (CTI) were also tested to verify the reliability of the results. Some simulation results of the images are shown as follow.

1st Approach: Replacement with Wavelet-DCT block

- Simulation on Lenna.bmp (512×512)

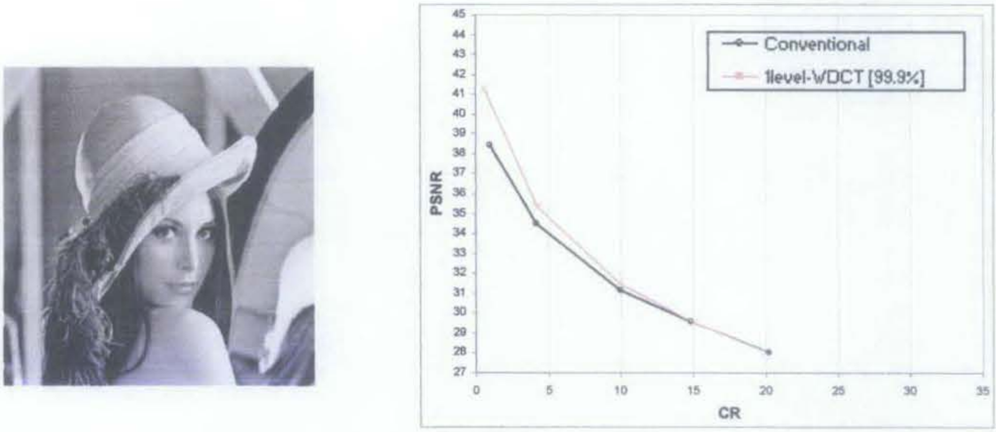


Figure 4.3 PSNR vs. CR plot for Lenna Image

- Simulation on frame220 (160×112)

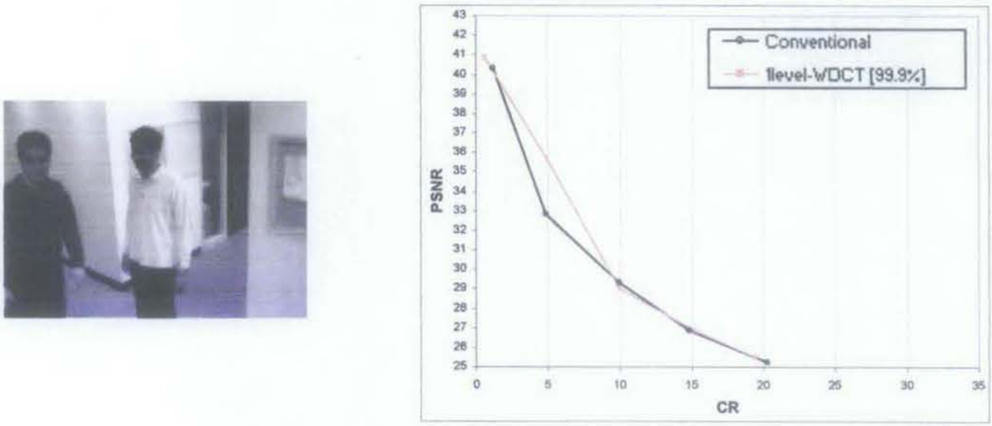


Figure 4.4 PSNR vs. CR plot for frame220

- Simulation is run on 280 images and the average of all PSNR for each respective CR is plotted on the following graph.

Table 4.1 Performance Comparison for DCT and Wavelet-DCT on CTI (1st Approach)

DCT		Wavelet-DCT	
PSNR	CR	PSNR	CR
41.13	1.186	41.06	0.8497
32.86	5.086	33.37	4.694
32.34	5.547	30.81	7.358
29.7	9.143	29.4	9.313
26.33	15.55	26.4	15.56
25.24	18.66	24.95	19.78

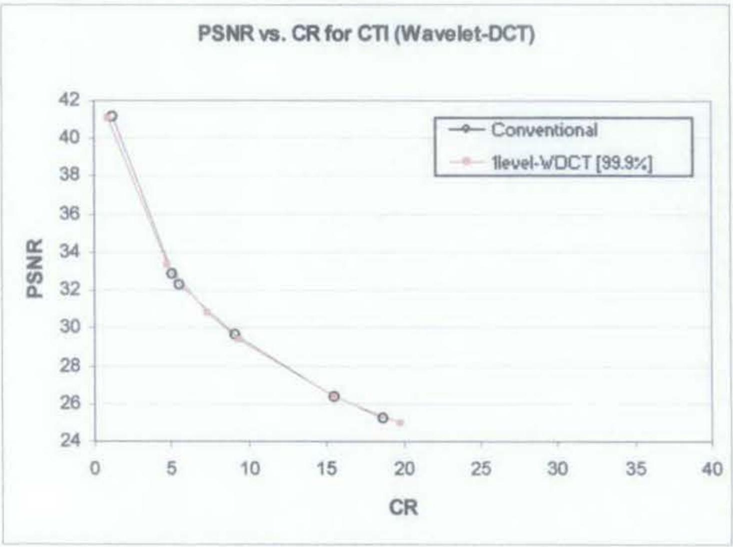


Figure 4.5 PSNR vs. CR plot for CTI (1st Approach)

Even though the individual simulation on Lenna image and frame-220 shows improvement in performance at lower CR, the simulation of many CTIs does not verify the improvement. The plot shows that both lines fall on almost the same points.

- Simulation is run on some images and the mean PSNR for each respective CR is plotted on the following graph.

Table 4.2 Performance Comparison for DCT and Wavelet-DCT on CTI (2^{nd} Approach)

DCT		Wavelet-DCT	
PSNR	CR	PSNR	CR
40.78	0.9657	35.22	0.6118
36.77	3.507	32.34	3.588
32.89	4.691	31.59	4.553
31.31	6.446	30.85	6.171
28.91	10.87	28.14	11.46
27.2	14.29	26.9	15.24
25.11	20.41	25.44	20.27
23.99	25.1	24.32	25.24
23.29	29.47	23.44	30.17

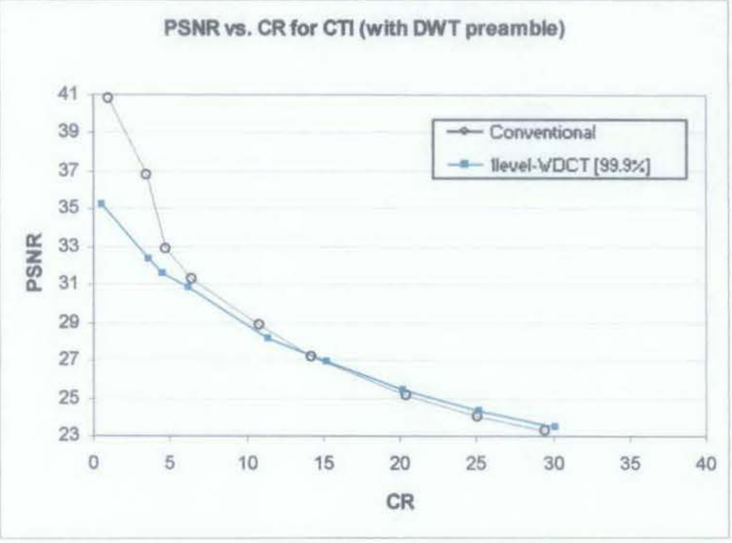


Figure 4.6 PSNR vs. CR plot for CTI (2^{nd} Approach)

The plot shows that the performance of 2^{nd} Approach is worse than the conventional one at lower compression. However, when it reaches compression ratio of 15, the turquoise line shows improvement on its performance.

4.2. Result and Discussion for DTI Input

1st Approach: Replacement with Wavelet-DCT block

- Simulation on frame251-frame250 (160×112)

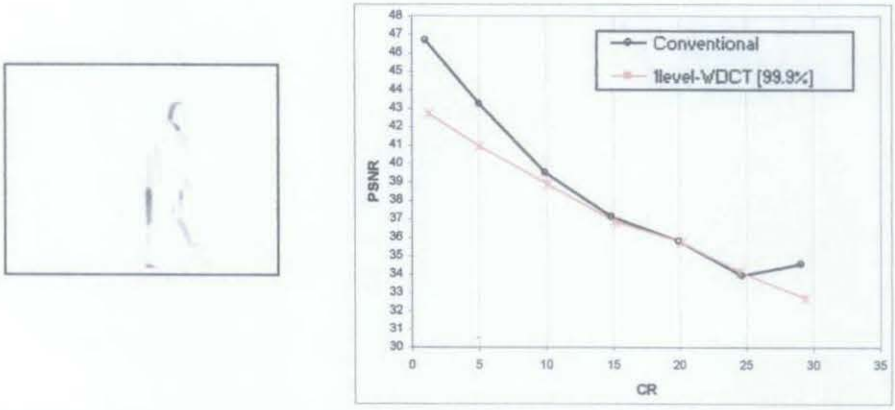


Figure 4.7 PSNR vs. CR plot for frame251-frame250 (inverted color)

- Simulation on frame221-frame220 (160×112)

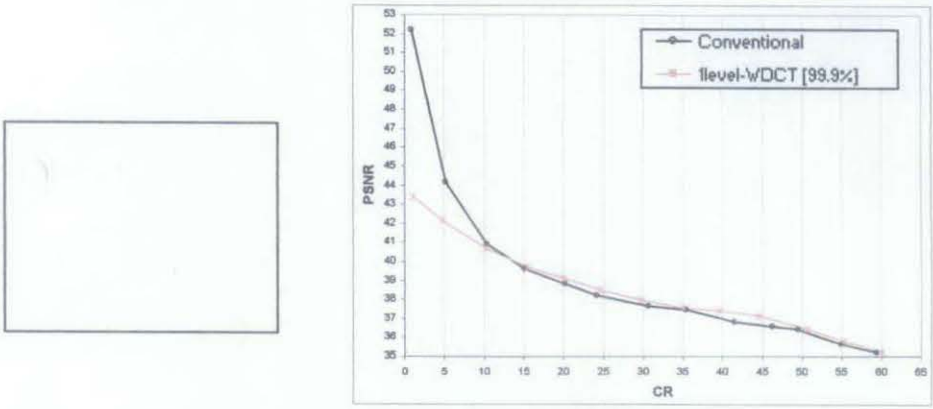


Figure 4.8 PSNR vs. CR plot for frame221-frame220 (inverted color)

- Simulation is run on 280 images and the mean PSNR for each respective CR is plotted on the following graph.

Table 4.3 Performance Comparison for DCT and Wavelet-DCT on DTI (1st Approach)

DCT		Wavelet-DCT	
PSNR	CR	PSNR	CR
52.25	1.757	47.33	2.275
50.87	5.221	46.52	6.268
47.59	10.65	45.40	10.64
45.43	15.50	44.55	15.23
44.10	19.82	43.49	20.90
42.29	25.26	42.49	25.31
41.49	30.17	41.59	30.37
40.95	34.43	40.90	35.24
39.40	40.47	39.90	40.74
39.18	45.30	39.22	45.66
36.66	51.03	37.83	51.33
35.47	54.05	37.47	55.41

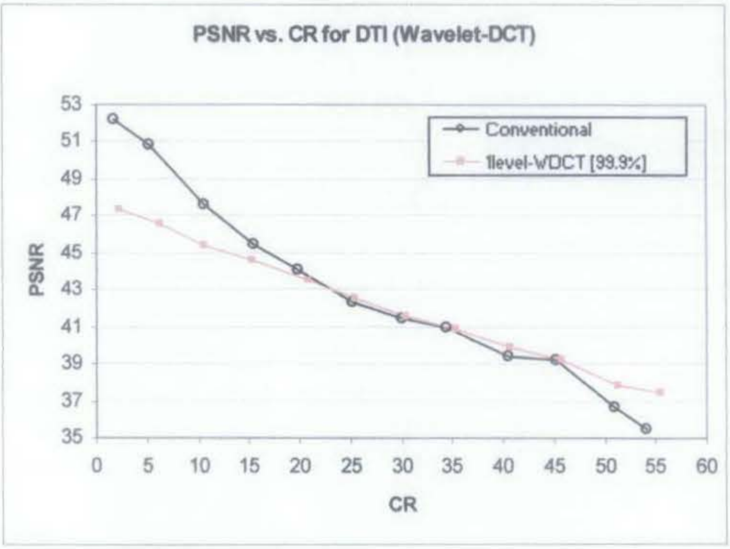


Figure 4.9 PSNR vs. CR plot for DTI (1st Approach)

The individual plots show some marginal improvement at high CR. These results are verified by the plot on figure 4.7 whose performance is worse than the conventional one at lower compression. However, when it reaches compression ratio of 30, the pink and black line are closely aligned and at higher compression (CR>40), improvement on its performance can be seen clearly.

- Simulation is run on some images and the mean PSNR for each respective CR is plotted on the following graph.

Table 4.4 Performance Comparison for DCT and Wavelet-DCT on DTI (2^nd Approach)

DCT		Wavelet-DCT	
PSNR	CR	PSNR	CR
52.22	0.9158	52.17	0.9133
44.16	5.164	44.17	5.167
40.91	10.38	40.91	10.37
39.61	15.16	39.75	14.69
38.83	20.18	38.96	19.19
38.24	24.15	38.05	25.67
37.69	30.68	37.69	30.68
37.52	35.42	37.52	35.70
36.87	41.48	37.36	39.82
36.64	46.19	37.20	44.14
36.46	49.5	36.79	50.62
35.70	55.00	36.42	53.65
35.23	59.34	36.21	57.81

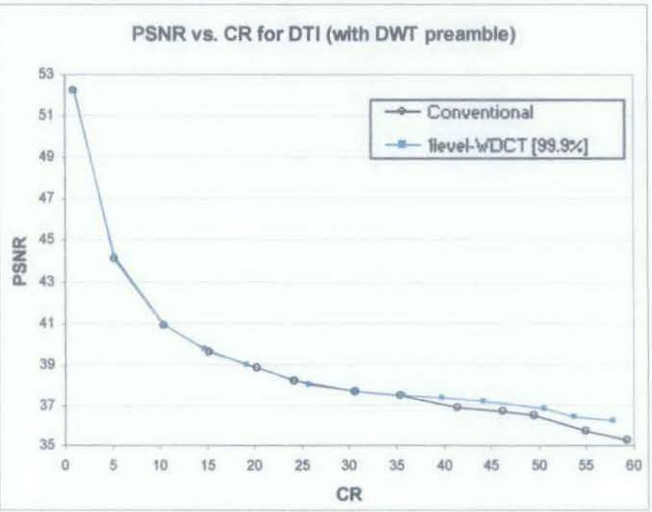


Figure 4.10 PSNR vs. CR plot for DTI (2^nd Approach)

The plot shows that its performance is worse than the conventional one at lower compression. However, when it reaches compression ratio of 15, the turquoise line shows marginal improvement and even better at higher compression.

4.3. Result and Discussion for Video Input

When video is fed as multimedia source, both CTI (I-frame) and DTI (P-frame and B-frame) are included in the sequence of frames. Both quantization tables are used accordingly.

1st Approach: Replacement with Wavelet-DCT block

Table 4.5 Performance Comparison for DCT and Wavelet-DCT on Video (1st Approach)

DCT		Wavelet-DCT	
PSNR	CR	PSNR	CR
48.01	1.428	44.45	1.696
44.9	4.921	43.11	4.859
41.65	9.961	41.01	10.17
39.72	14.57	39.74	14.23
38.02	19.55	37.98	19.73
36.67	24.7	36.55	24.65
35.18	30.01	35.42	29.96
33.57	34.89	33.57	35.16
33.45	39.45	33.45	39.85
32.12	45.22	32.12	45.33

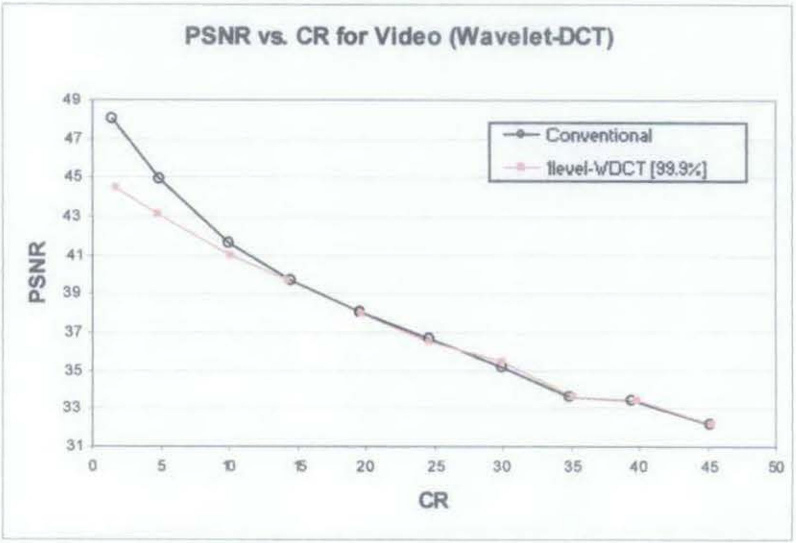


Figure 4.11 PSNR vs. CR plot for Video Input (1st Approach)

2^{nd} Approach: with Preamble of DWT&IDWT

Table 4.6 Performance Comparison for DCT and Wavelet-DCT on Video (2^{nd} Approach)

DCT		Wavelet-DCT	
PSNR	CR	PSNR	CR
49.93	0.925	48.78	0.853
44.46	3.094	43.07	3.52
41.90	5.069	41.65	5.044
40.37	7.183	40.28	7.114
38.51	10.478	38.36	10.588
37.13	14.986	37.18	14.80
36.08	20.226	36.26	19.406
35.39	24.340	35.30	25.584
34.81	30.438	34.84	30.578

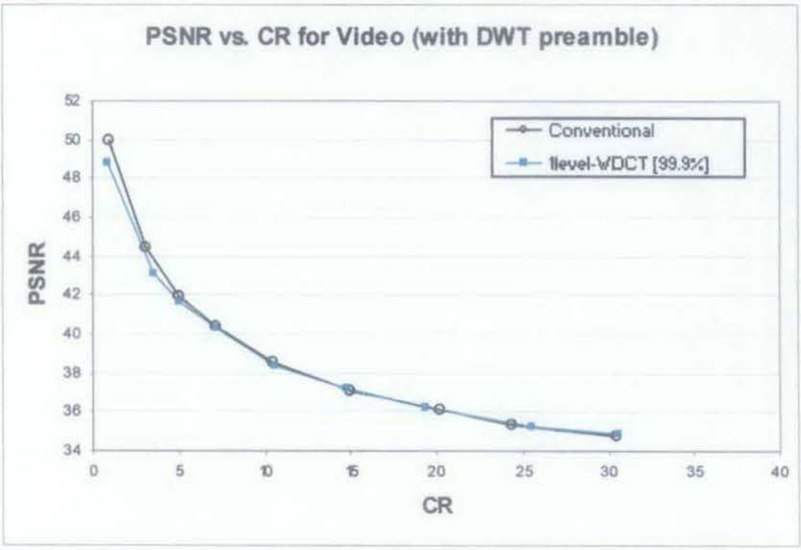


Figure 4.12 PSNR vs. CR plot for Video Input (2^{nd} Approach)

On the 1^{st} Approach, the plot shows worse performance at lower compression ratio. And on the 2^{nd} Approach, the graph shows that both lines fall on almost the same points.

4.4. Analysis on Performance of Wavelet-DCT based Intra-frame Coder

Both approaches to integrate a wavelet-based compression within MPEG-2 intra-frame coder do not show remarkable improvement in its performance. However, for some cases, wavelet-DCT based intra-frame coder shows marginal improvement at high compression ratio.

Besides the two approaches, some other techniques have also been tested. Those techniques include J-level Wavelet Packet based DCT and using Wavelet transform as the transform coding without recombination to DCT coefficients. Nevertheless, these techniques do not offer improvement in performance.

There are some reasons that contribute to the under-performance of Wavelet-DCT based intra-frame. The focus of the project should not be too restricted with the objective to provide backward compatibility. If wavelet-based compression is performed, the other blocks that precede or follow the wavelet transform block should have some adjustments in order to maximize the compression capability. The other reason is that a Discrete Continuous Image (DTI) has already had very small amount of information in it. Thus, additional spatial-based compression on this type of image is inflexible. Another thing is that wavelet transform, like in JPEG2000, is usually performed on the whole image at higher level (level 4 or 5), not on a small block of 16 by16. Wavelet Transform has more contribution to compression if a proper way to losslessly encode the wavelet coefficients is employed.

Wavelet transform can provide an optimal representation for many signals containing singularities. The key to the sparsity is that since wavelets oscillate locally, only wavelets overlapping a singularity have large wavelet coefficients while other coefficients are small. However, in spite of its efficient computational algorithm and sparse representation, the wavelet transform suffers from four fundamental shortcomings [14]:

- Oscillations

  Wavelet coefficients tend to oscillate positive and negative around singularities which complicates singularity extraction very challenging. The conventional wisdom that singularities yield large wavelet coefficients is overstated.

- Shift Variance

  A small shift of the signal greatly disturbs the wavelet coefficient oscillation pattern around singularities. In video file, most of the frames in sequence are almost the same only experience small shift.

- Aliasing

  Substantial aliasing become a hindrance to the forward and inverse transform, leading to artefacts in the reconstructed signal.

- Lack of Directionality

  It greatly complicates modelling and processing of geometric images like ridges and edges.

CHAPTER 5
CONCLUSION AND RECOMMENDATION

The last chapter of the report contains two sections; conclusion and recommendation. The conclusion reviews the objectives of this project followed by a brief discussion on methodology and the result obtained from the simulation. The recommendation states some suggestions on possible areas to be delved in for image and video compression field.

5.1 Conclusion

A full intra-frame coder has been developed in compliance with the MPEG2 standards. The intra-frame coder employs 16x16 block processing. The internal blocks of the simulation are built on SIMULINK except for some blocks (Wavelet-DCT and lossless coding) which are coded in MATLAB m-file. The algorithm to selectively determine the threshold level by signal energy analysis of histogram has been developed in MATLAB and integrated onto the simulation. A statistical investigation to find the best wavelet for DTI has been done and it proves that Haar/Daubechies1 wavelet offers the best representation for DTI.

Different types of images are used as test images; DTI and CTI. Video files that comprises of both CTI (I-frame) and DTI (P-frame and B-frame) were also tested. Two approaches were used in the attempt of performing a wavelet-based compression and maintaining the backward compatibility of MPEG2 performance.

When image is fed as the input, the performance of Wavelet-DCT based intra-frame coder is inferior at low compression ratio, but it shows minor improvement at higher compression ratio. The 2nd approach using preamble DWT&IDWT shows marginal improvement starting from a rather low compression ratio and it increases towards higher compression.

However, when video file is fed as an input, the performance of both approaches does not show reasonable improvement at various level of compression ratio.

A discussion on why Wavelet-DCT based intra-frame coder does not perform as expected has been elaborated in the previous chapter. The objective to provide the backward compatibility has been a restriction to improve the video coder performance. More adjustments should be employed to the other blocks preceding/following the wavelet-based compression block.

Wavelet transform is able to perform efficient computation and represents signals in a sparse matrix. Even so, it suffers from four fundamental shortcomings which have been further discussed in the previous chapter. One possible solution that able to handle such problems is to employ complex wavelet transform [14].

5.2 Recommendation

For future work on image and video compression, these are some suggestions of possible fields to delve in.

5.2.1 Distributed Video Coding

A distributed video coding attempt to specify classes of frames and it determines whether it is necessary to perform inter-frame coding or to skip the process.

5.2.2 Curvelet and Ridgelet Transform

The ridgelet transform was introduced as a new multiscale representation for functions on continuous spaces that are smooth away from discontinuing along lines.

The transform is invertible, non-redundant and able to be computed via fast algorithm [15].

5.2.3 Dual-Tree Complex Wavelet Transform

The dual-tree complex wavelet-transform is enhancement to the DWT with important additional properties: It is nearly shift-invariant and directionally selective in two or higher dimensions [14].

REFERENCES

[1] : Chai Beng Seow, *Wavelet-DCT Based Image Coder for Video Coding Applications*, Bachelor Thesis, Electrical and Electronics Engineering Department, Universiti Teknologi PETRONAS, Malaysia, June 2007.

[2] : S. Mallat and Wen Liang Hwang, *Singularity Detection and Processing with Wavelets*, IEEE Transactions on Information Theory, Vol.38, No.2, March 1992.

[3] : How Sun Dee, *Wavelet-Based Block Transform and Their Applications*, Master Thesis, Electrical and Electronics Engineering Department, Universiti Sains Malaysia, Malaysia, January 2001.

[4] : C. Sidney Burrus, Ramesh A. Gopinath, and Haitao Guo, (1998), *Introduction to Wavelets and Wavelet Transforms: A Primer*, New Jersey: Prentice Hall.

[5] : Ho Tatt Wei, *Implementation of The Fast Approximate Fourier Transform via Discrete Wavelet Transform on the TI TMS320C5402*, Digital Signal Processing Project, Universiti Teknologi PETRONAS, Tronoh, Perak, Malaysia, November 2002.

[6] : Yucel Altunbasak and Nejat Kamaci, *An Analysis of The DCT Coefficient Distribution with The H.264 Video Coder*, 0-7803-8484-9/04/$20.00©2004 IEEE.

[7] : Mohammed Ghanbari, *Standard Codecs: Image Compression to Advanced Video Coding*, (2003), London: The Institution of Electrical Engineers.

[8] : Peter Symes, *Digital Video Compression*, (2004), New York: McGraw Hill.

[9] : Barry G. Haskell, Atun Puri, and Arun N. Netravali, *Digital Video: An Introduction to MPEG-2*, (1999), Massachusetts: Kluwer Academics Publishers

[10] : Syed Ali Khayam, *The Discrete Cosine Transform (DCT): Theory and Application*, (2003), Department of Electrical & Computer Engineering, Michigan State University.

[11] : Stefan Eireiner, based on Code by Peter J. Acklam, *Run Length Coding*, Last Update: May.14. 2004.

[12] : Karl Skretting, *Arithmetic Coding and Huffman Coding in MATLAB*, Cybernetics Group at University of Stavanger, Last Update: Oct.11. 2005. http://www.ux.uis.no/~karlsk/proj99/, 10 January 2008.

[13] : Standard Test Image – Web-based free-content multilingual encyclopaedia http://en.wikipedia.org/wiki/Standard_test_image, 05 May 2008.

[14] : Ivan W. Selesnick, Richard G. Baraniuk, and Nick G. Kingsbury, *The Dual-Tree Complex Wavelet Transform*, 1053-5888/05/$20.00©2005 IEEE.

[15] : Minh N. Do and Martin Vetterli, *The Finite Ridgelet Transform for Image Representation*, (2001), Audio-Visual Communications Laboratory, Swiss Federal Institute of Technology.

[16] : D.L. Donoho, *Denoising by soft-thresholding*, IEEE Transaction on Information Theory, Vol. 41, May 1993.

[17] : Kaj Rosengren, *Modelling and Implementation of an MPEG-2 Video Decoder Using a Gals Design Path*, Thesis, Linkoping Universiteit, June 2006.

[18] : Shanshank Khanvilkar, *MPEG-2 Encoder/Decoder SW Report*.

[19] : http://www.bulents-byteme.com/digimed/MPEG_video_encoding.asp

[20] : Gorry Fairhust, http://www.erg.abdn.ac.uk/research/future-net/digital-video/mpeg2.html, gory@erg.abdn.ac.uk, January 2001.

[21] : http://www.owlnet.rice.edu/~elec301/Projects00/ wavelet_image_comp /index.html

[22] : S. Grace Chang, Bin Yu, and Martin Vetterli, *Adaptive Wavelet Thresholding for Image Denoising and Compression,* IEEE Transaction on Image Processing, Vol. 9, September 2000.

[23] : Nicu Sebe, Claudiu Lamba, Michael S. Lew, An Overcomplete Discrete Wavelet Transform for Video Compression.

Appendix A. Default Intra and Inter Quantization Weighing Matrices

```
Qc_16x16 = [16  14  11  11  10  12  15  18  22  27  35  41  47  52  56  61;
            14  13  12  12  12  14  16  19  23  30  41  49  52  55  57  58;
            12  12  12  13  13  15  18  21  24  32  46  57  58  59  57  55;
            13  13  12  13  14  17  20  24  29  37  49  58  61  63  59  55;
            14  13  13  14  15  18  22  27  34  42  51  59  64  67  61  56;
            14  14  14  16  18  21  24  30  39  48  59  68  70  72  65  58;
            14  15  16  18  20  23  27  34  43  55  70  81  79  77  69  61;
            15  17  18  21  25  29  34  41  50  63  80  92  89  85  75  66;
            17  19  20  26  31  38  46  53  60  71  90 102  99  95  84  73;
            19  22  24  31  38  46  54  61  67  78  96 108 106 103  92  80;
            22  26  30  38  47  53  59  65  73  82  96 106 108 108  98  87;
            27  33  38  47  56  61  65  72  79  88  99 107 111 113 103  93;
            39  45  52  59  67  72  76  82  90  98 108 115 116 116 107  97;
            51  58  65  71  77  82  86  92  99 107 114 120 119 118 109 101;
            61  70  78  82  86  89  92  97 104 108 109 110 110 110 105 100;
            72  81  91  93  95  96  97 102 108 110 104 100 102 103 101  99];
```

Intra Quantization Matrix

```
Qd_16x16 = [16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16;
            16  16  16  16  16  16  16  16  16  16  16  16  16  16  16  16];
```

Inter Quantization Matrix

Appendix B. Statistical Result for Wavelet Selection

Summary

Wavelet Selection (preserving 90% energy)									
Number of occurrence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
f221-f220	39	14	4	1	14	4	11	7	9
f21-f20	60	7	0	0	7	0	2	2	2
f151-f150	44	13	0	0	13	0	9	4	4
f151-f150	53	7	2	2	7	2	8	3	5
f121-f120	43	10	3	4	10	3	9	3	4
f251-f250	49	13	2	2	13	2	13	5	6
Total	288	64	11	9	64	11	52	24	30
%	68.57	15.24	2.62	2.14	15.24	2.62	12.38	5.71	7.14

Wavelet Selection (preserving 80% energy)									
Number of occurrence as the best wavelet	db1	db2	db3	db4	sym2	sym3	.sym4	coif1	coif2
f221-f220	39	18	4	3	18	4	14	6	8
f21-f20	49	15	4	1	15	4	8	2	5
f151-f150	33	17	1	3	17	1	15	5	9
f151-f150	48	10	3	3	10	3	14	6	7
f121-f120	36	13	6	6	13	6	15	3	6
f251-f250	41	20	3	3	20	3	16	5	10
Total	246	93	21	19	93	21	82	27	45
%	58.57	22.14	5	4.524	22.14	5	19.52	6.429	10.71

Wavelet Selection (preserving 70% energy)									
Number of occurrence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
f221-f220	36	17	8	7	17	8	16	8	10
f21-f20	50	17	5	4	17	5	14	6	4
f151-f150	34	17	4	3	17	4	19	6	11
f151-f150	45	13	1	7	13	1	19	10	9
f121-f120	36	23	6	8	23	6	18	6	10
f251-f250	43	19	5	6	19	5	15	9	14
Total	244	106	29	35	106	29	101	45	58
%	58.1	25.24	6.905	8.333	25.24	6.905	24.05	10.71	13.81

* For each block, it is possible to have more than 1 best wavelet.

Preserving 90% energy within the 16x16 block

f221-f220	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	22	22	26	25	22	26	20	19	20	19
{2,1}	28	23	27	30	23	27	21	22	20	20
{3,1}	57	51	53	56	51	53	50	56	51	50
{4,1}	32	41	44	48	41	44	38	42	41	32
{5,1}	48	49	44	43	49	44	42	48	45	42
{6,1}	56	60	65	65	60	65	53	54	52	52
{7,1}	16	14	16	18	14	16	15	18	17	14
{1,2}	10	15	22	28	15	22	17	19	19	10
{2,2}	28	25	22	19	25	22	20	21	19	19
{3,2}	56	50	46	51	50	46	52	57	53	46
{4,2}	57	52	58	59	52	58	56	59	57	52
{5,2}	12	18	19	22	18	19	15	15	15	12
{6,2}	55	55	62	67	55	62	54	55	54	54
{7,2}	66	65	66	66	65	66	63	73	67	63
{1,3}	44	51	65	67	51	65	48	53	51	44
{2,3}	18	16	17	17	16	17	12	12	12	12
{3,3}	34	37	43	44	37	43	40	39	40	34
{4,3}	26	31	33	32	31	33	26	25	25	25
{5,3}	41	41	41	41	41	41	37	35	35	35
{6,3}	59	49	54	59	49	54	64	67	66	49
{7,3}	21	29	31	33	29	31	33	32	33	21
{1,4}	24	42	62	76	42	62	42	49	52	24
{2,4}	68	64	79	85	64	79	65	66	67	64
{3,4}	8	13	24	30	13	24	12	16	18	8
{4,4}	16	21	24	28	21	24	23	25	24	16
{5,4}	23	31	36	43	31	36	33	35	34	23
{6,4}	47	47	44	49	47	44	46	49	49	44
{7,4}	78	67	71	80	67	71	74	72	72	67
{1,5}	46	51	53	51	51	53	53	54	55	46
{2,5}	40	43	50	54	43	50	38	40	40	38
{3,5}	71	70	73	72	70	73	71	68	71	68
{4,5}	24	28	31	34	28	31	29	30	30	24
{5,5}	64	58	55	56	58	55	55	59	56	55
{6,5}	46	42	44	50	42	44	48	46	47	42
{7,5}	33	29	29	35	29	29	35	35	36	29
{1,6}	36	40	55	73	40	55	49	52	55	36
{2,6}	37	41	46	50	41	46	39	41	43	37
{3,6}	59	58	63	66	58	63	57	61	58	57
{4,6}	65	64	72	69	64	72	71	75	75	64
{5,6}	65	60	60	65	60	60	58	61	60	58
{6,6}	37	32	35	38	32	35	33	33	32	32
{7,6}	28	30	33	33	30	33	28	28	28	28
{1,7}	19	48	81	90	48	81	51	70	70	19
{2,7}	72	78	84	83	78	84	75	77	78	72
{3,7}	28	37	59	67	37	59	44	53	53	28
{4,7}	20	23	35	45	23	35	30	31	32	20
{5,7}	13	13	17	23	13	17	17	18	18	13
{6,7}	17	16	20	24	16	20	22	21	24	16
{7,7}	64	66	77	79	66	77	64	69	66	64
{1,8}	1	4	9	15	4	9	5	6	7	1
{2,8}	45	71	99	106	71	99	77	87	85	45
{3,8}	75	69	75	82	69	75	74	72	73	69
{4,8}	21	47	79	97	47	79	59	67	74	21
{5,8}	15	35	61	82	35	61	41	51	52	15
{6,8}	20	44	73	89	44	73	52	65	63	20
{7,8}	51	55	65	70	55	65	55	59	59	51
{1,9}	19	18	22	27	18	22	25	25	27	18
{2,9}	45	51	52	54	51	52	49	52	50	45
{3,9}	49	64	69	65	64	69	51	49	50	49
{4,9}	51	40	52	66	40	52	65	69	69	40
{5,9}	37	49	63	72	49	63	51	54	57	37
{6,9}	40	46	58	57	46	58	45	48	48	40
{7,9}	34	58	92	100	58	92	66	78	80	34
{1,10}	4	12	22	29	12	22	13	16	19	4
{2,10}	14	43	76	95	43	76	45	60	61	14
{3,10}	15	38	68	80	38	68	42	56	58	15
{4,10}	18	51	85	95	51	85	53	73	73	18
{5,10}	24	44	68	76	44	68	47	57	59	24
{6,10}	47	71	87	89	71	87	70	76	77	47
{7,10}	32	48	74	84	48	74	58	70	73	32

Number of occurrence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	39	14	4	1	14	4	11	7	9

f21-f20	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	24	35	46	44	35	46	33	32	36	24
{2,1}	4	6	9	12	6	9	5	6	6	4
{3,1}	29	46	62	81	46	62	45	51	54	29
{4,1}	55	63	73	74	63	73	71	73	71	55
{5,1}	84	73	76	73	73	76	71	71	67	67
{6,1}	25	57	95	105	57	95	65	75	80	25
{7,1}	62	67	79	76	67	79	65	77	75	62
{1,2}	5	7	12	13	7	12	9	10	11	5
{2,2}	19	46	72	86	46	72	47	60	62	19
{3,2}	70	82	89	85	82	89	71	69	71	69
{4,2}	64	72	87	99	72	87	76	80	79	64
{5,2}	65	68	73	82	68	73	72	72	73	65
{6,2}	46	62	86	91	62	86	64	73	72	46
{7,2}	25	50	71	87	50	71	59	68	71	25
{1,3}	21	18	25	34	18	25	30	34	33	18
{2,3}	20	33	50	56	33	50	29	33	35	20
{3,3}	69	72	89	93	72	89	85	87	86	69
{4,3}	3	9	17	27	9	17	10	13	14	3
{5,3}	27	50	72	81	50	72	45	54	58	27
{6,3}	76	86	97	101	86	97	96	94	99	76
{7,3}	84	92	111	111	92	111	86	97	95	84
{1,4}	52	65	71	74	65	71	71	76	76	52
{2,4}	19	44	75	87	44	75	45	59	61	19
{3,4}	22	56	85	98	56	85	60	70	73	22
{4,4}	21	39	52	61	39	52	40	43	45	21
{5,4}	47	54	61	64	54	61	59	59	61	47
{6,4}	56	59	61	63	59	61	62	64	66	56
{7,4}	38	69	102	105	69	102	75	83	85	38
{1,5}	23	26	36	48	26	36	37	39	42	23
{2,5}	49	50	60	63	50	60	52	55	55	49
{3,5}	49	50	60	61	50	60	55	60	58	49
{4,5}	57	61	67	72	61	67	69	68	69	57
{5,5}	35	39	43	48	39	43	39	43	42	35
{6,5}	54	64	67	68	64	67	53	55	53	53
{7,5}	28	58	96	110	58	96	64	83	82	28
{1,6}	30	42	57	61	42	57	44	43	45	30
{2,6}	33	42	57	62	42	57	42	47	47	33
{3,6}	18	22	24	24	22	24	20	21	22	18
{4,6}	27	28	31	34	28	31	29	31	31	27
{5,6}	57	57	61	65	57	61	53	58	58	53
{6,6}	63	64	70	69	64	70	58	55	57	55
{7,6}	53	66	77	81	66	77	65	69	72	53
{1,7}	17	36	63	74	36	63	40	52	53	17
{2,7}	24	59	91	106	59	91	62	80	82	24
{3,7}	19	35	60	85	35	60	36	47	50	19
{4,7}	42	44	60	67	44	60	55	57	58	42
{5,7}	65	74	86	86	74	86	66	67	67	65
{6,7}	54	70	86	94	70	86	71	77	81	54
{7,7}	38	37	46	48	37	46	44	46	45	37
{1,8}	28	26	34	42	26	34	28	30	31	26
{2,8}	17	18	24	30	18	24	21	24	22	17
{3,8}	15	17	24	31	17	24	24	30	28	15
{4,8}	24	24	34	37	24	34	33	39	36	24
{5,8}	27	25	28	30	25	28	26	33	28	25
{6,8}	34	37	38	43	37	38	39	40	39	34
{7,8}	37	34	41	41	34	41	38	38	38	34
{1,9}	34	61	89	92	61	89	68	82	82	34
{2,9}	4	9	13	16	9	13	8	10	10	4
{3,9}	49	49	54	52	49	54	54	54	52	49
{4,9}	21	22	24	33	22	24	29	25	30	21
{5,9}	9	12	16	18	12	16	12	11	12	9
{6,9}	39	40	50	54	40	50	40	45	44	39
{7,9}	53	66	81	88	66	81	74	77	79	53
{1,10}	5	12	18	23	12	18	14	15	17	5
{2,10}	37	47	57	58	47	57	39	40	40	37
{3,10}	46	54	64	61	54	64	50	53	51	46
{4,10}	11	37	68	89	37	68	41	55	57	11
{5,10}	13	33	60	85	33	60	38	49	50	13
{6,10}	23	32	43	43	32	43	29	32	31	23
{7,10}	24	41	59	71	41	59	45	55	57	24

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	60	7	0	0	7	0	2	2	2

f151-f150	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	41	35	37	44	35	37	35	33	36	33
{2,1}	27	35	42	46	35	42	33	35	35	27
{3,1}	58	53	58	60	53	58	61	64	63	53
{4,1}	38	73	104	115	73	104	76	88	85	38
{5,1}	44	46	50	52	46	50	34	37	35	34
{6,1}	62	53	72	78	53	72	54	57	55	53
{7,1}	63	69	71	70	69	71	67	70	67	63
{1,2}	69	64	77	83	64	77	80	70	71	64
{2,2}	55	50	54	58	50	54	56	59	55	50
{3,2}	64	66	59	58	66	59	61	57	58	57
{4,2}	32	40	48	53	40	48	36	42	42	32
{5,2}	43	47	59	77	47	59	51	55	56	43
{6,2}	80	75	78	76	75	78	73	81	74	73
{7,2}	73	74	91	103	74	91	85	91	90	73
{1,3}	66	80	99	101	80	99	79	84	86	66
{2,3}	17	24	31	35	24	31	20	21	24	17
{3,3}	76	79	85	89	79	85	79	78	80	76
{4,3}	101	89	93	98	89	93	100	103	103	89
{5,3}	83	77	83	87	77	83	84	86	84	77
{6,3}	45	47	59	65	47	59	49	52	53	45
{7,3}	54	56	64	69	56	64	50	57	51	50
{1,4}	65	72	81	80	72	81	71	69	75	65
{2,4}	43	47	48	56	47	48	47	48	50	43
{3,4}	67	69	76	82	69	76	68	76	76	67
{4,4}	10	13	19	23	13	19	15	16	18	10
{5,4}	41	41	42	42	41	42	25	31	27	25
{6,4}	31	34	33	30	34	33	24	28	23	23
{7,4}	49	43	44	45	43	44	42	43	42	42
{1,5}	32	28	29	34	28	29	33	36	36	28
{2,5}	37	46	50	52	46	50	39	41	42	37
{3,5}	78	85	90	87	85	90	79	84	82	78
{4,5}	46	44	50	56	44	50	48	48	47	44
{5,5}	20	22	22	24	22	22	29	30	32	20
{6,5}	28	32	32	34	32	32	28	25	25	25
{7,5}	36	33	36	39	33	36	34	33	32	32
{1,6}	37	53	59	58	53	59	47	51	51	37
{2,6}	22	23	25	30	23	25	21	23	24	21
{3,6}	56	59	63	65	59	63	64	65	66	56
{4,6}	69	63	64	70	63	64	70	76	73	63
{5,6}	70	69	75	77	69	75	65	68	66	65
{6,6}	37	33	37	37	33	37	34	38	34	33
{7,6}	24	34	37	38	34	37	28	28	28	24
{1,7}	36	34	45	46	34	45	36	37	38	34
{2,7}	81	82	85	85	82	85	78	79	80	78
{3,7}	28	64	102	115	64	102	70	85	88	28
{4,7}	48	59	73	79	59	73	58	64	63	48
{5,7}	40	38	47	51	38	47	48	51	50	38
{6,7}	9	12	14	15	12	14	14	14	16	9
{7,7}	25	41	60	64	41	60	43	44	48	25
{1,8}	43	88	114	121	88	114	85	99	99	43
{2,8}	32	39	49	54	39	49	40	44	44	32
{3,8}	50	55	75	78	55	75	60	63	63	50
{4,8}	57	58	69	70	58	69	64	62	64	57
{5,8}	54	55	65	68	55	65	66	65	69	54
{6,8}	69	64	65	70	64	65	54	54	56	54
{7,8}	61	68	76	81	68	76	71	70	74	61
{1,9}	44	52	58	65	52	58	46	51	51	44
{2,9}	48	57	70	72	57	70	53	58	56	48
{3,9}	24	34	40	45	34	40	37	32	36	24
{4,9}	58	64	77	83	64	77	76	77	78	58
{5,9}	40	71	105	106	71	105	79	95	93	40
{6,9}	28	50	91	102	50	91	58	73	76	28
{7,9}	33	63	101	106	63	101	68	86	90	33
{1,10}	17	48	80	100	48	80	52	69	69	17
{2,10}	61	62	79	83	62	79	70	77	76	61
{3,10}	16	39	62	82	39	62	44	54	57	16
{4,10}	55	61	60	60	61	60	56	61	61	55
{5,10}	48	54	67	65	54	67	50	50	50	48
{6,10}	42	59	88	105	59	88	63	72	73	42
{7,10}	47	47	49	51	47	49	55	56	55	47

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	44	13	0	0	13	0	9	4	4

f51-f50	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	18	23	29	33	23	29	27	25	29	18
{2,1}	45	54	61	64	54	61	47	49	50	45
{3,1}	34	49	63	77	49	63	45	48	50	34
{4,1}	52	66	77	79	66	77	59	61	65	52
{5,1}	55	48	59	71	48	59	53	62	61	48
{6,1}	35	49	74	81	49	74	59	66	67	35
{7,1}	37	54	89	107	54	89	73	84	87	37
{1,2}	0	0	0	0	0	0	0	0	0	0
{2,2}	14	21	49	68	21	49	31	42	43	14
{3,2}	10	21	31	36	21	31	23	28	27	10
{4,2}	26	41	79	93	41	79	45	61	59	26
{5,2}	27	62	96	107	62	96	63	80	80	27
{6,2}	49	68	88	94	68	88	67	76	75	49
{7,2}	25	65	108	119	65	108	65	81	83	25
{1,3}	25	33	47	53	33	47	39	44	48	25
{2,3}	79	85	92	102	85	92	86	89	91	79
{3,3}	74	76	88	99	76	88	83	81	82	74
{4,3}	3	9	17	27	9	17	10	13	13	3
{5,3}	29	50	79	92	50	79	49	61	62	29
{6,3}	66	67	79	81	67	79	65	70	70	65
{7,3}	32	69	106	117	69	106	75	88	90	32
{1,4}	13	42	70	86	42	70	52	66	70	13
{2,4}	17	43	68	83	43	68	42	53	54	17
{3,4}	20	35	49	48	35	49	41	43	46	20
{4,4}	18	32	44	59	32	44	33	40	42	18
{5,4}	4	7	9	12	7	9	8	8	8	4
{6,4}	6	8	10	10	8	10	5	4	4	4
{7,4}	55	61	71	72	61	71	57	63	63	55
{1,5}	24	20	25	32	20	25	25	29	28	20
{2,5}	37	43	45	45	43	45	33	34	32	32
{3,5}	42	44	46	49	44	46	41	45	45	41
{4,5}	48	50	53	54	50	53	43	44	45	43
{5,5}	36	38	39	39	39	39	30	32	30	30
{6,5}	25	23	27	26	23	27	27	30	29	23
{7,5}	14	15	19	17	15	19	20	20	21	14
{1,6}	27	25	28	36	25	28	27	30	30	25
{2,6}	20	22	26	27	22	26	23	24	24	20
{3,6}	66	62	63	60	62	63	56	56	54	54
{4,6}	53	57	60	53	57	60	50	56	55	50
{5,6}	54	50	55	57	50	55	43	49	45	43
{6,6}	39	38	37	35	38	37	40	41	40	35
{7,6}	25	23	25	26	23	25	25	27	27	23
{1,7}	39	50	71	76	50	71	50	53	56	39
{2,7}	33	51	74	83	51	74	53	58	60	33
{3,7}	65	74	86	86	74	86	73	81	81	65
{4,7}	21	28	34	41	28	34	25	23	24	21
{5,7}	11	14	21	28	14	21	15	16	18	11
{6,7}	5	6	13	18	6	13	7	7	7	5
{7,7}	39	46	58	65	46	58	37	43	43	37
{1,8}	58	68	79	85	68	79	63	66	65	58
{2,8}	48	57	78	77	57	78	59	65	65	48
{3,8}	30	37	47	53	37	47	37	40	42	30
{4,8}	96	93	97	95	93	97	89	84	87	84
{5,8}	48	70	80	79	70	80	66	75	78	48
{6,8}	37	52	75	89	52	75	55	64	65	37
{7,8}	43	48	59	67	48	59	56	60	60	43
{1,9}	23	31	47	54	31	47	29	33	34	23
{2,9}	64	69	68	71	69	68	71	70	72	64
{3,9}	58	67	71	68	67	71	69	71	69	58
{4,9}	31	31	31	34	31	31	34	35	33	31
{5,9}	30	35	41	47	35	41	31	32	33	30
{6,9}	3	11	25	42	11	25	13	17	19	3
{7,9}	11	16	23	29	16	23	19	21	23	11
{1,10}	5	12	23	34	12	23	14	17	18	5
{2,10}	38	57	77	79	57	77	64	72	73	38
{3,10}	42	53	68	71	53	68	43	47	46	42
{4,10}	49	59	62	58	59	62	51	53	52	49
{5,10}	47	63	73	75	63	73	63	67	68	47
{6,10}	25	56	89	90	56	89	53	69	72	25
{7,10}	25	37	47	51	37	47	38	44	42	25

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	53	7	2	2	7	2	8	3	5

f121-f120	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	31	35	38	41	35	38	33	36	35	31
{2,1}	69	73	72	66	73	72	72	72	71	66
{3,1}	50	43	46	51	43	46	43	47	42	42
{4,1}	26	28	37	44	28	37	39	42	45	26
{5,1}	39	39	43	50	39	43	44	45	45	39
{6,1}	48	48	59	65	48	59	47	52	48	47
{7,1}	91	99	102	93	99	102	84	88	87	84
{1,2}	21	24	23	22	24	23	26	26	27	21
{2,2}	28	24	26	28	24	26	28	30	29	24
{3,2}	75	69	68	73	69	68	69	70	70	68
{4,2}	52	52	52	50	52	52	50	52	49	49
{5,2}	56	63	63	63	63	63	49	52	52	49
{6,2}	64	71	72	67	71	72	74	75	76	64
{7,2}	28	26	29	25	26	29	23	26	22	22
{1,3}	25	38	53	65	38	53	40	46	46	25
{2,3}	66	73	86	89	73	86	76	77	81	66
{3,3}	40	46	55	61	46	55	44	51	49	40
{4,3}	34	34	35	36	34	35	35	36	35	34
{5,3}	49	50	51	56	50	51	50	53	52	49
{6,3}	58	49	49	52	49	49	51	57	52	49
{7,3}	27	24	26	27	24	26	20	23	20	20
{1,4}	30	38	45	50	38	45	45	48	48	30
{2,4}	26	24	34	45	24	34	35	34	37	24
{3,4}	23	25	32	35	25	32	31	28	33	23
{4,4}	50	50	49	52	50	49	52	49	51	49
{5,4}	36	37	42	46	37	42	33	35	35	33
{6,4}	66	70	79	75	70	79	61	65	62	61
{7,4}	56	56	55	53	56	55	54	55	54	53
{1,5}	32	34	43	44	34	43	37	36	38	32
{2,5}	52	58	77	84	58	77	58	64	61	52
{3,5}	54	50	53	58	50	53	56	53	52	50
{4,5}	66	69	81	89	69	81	77	79	79	66
{5,5}	67	68	72	70	68	72	67	66	68	66
{6,5}	44	47	42	37	47	42	40	39	38	37
{7,5}	43	46	55	65	46	55	49	54	54	43
{1,6}	56	52	65	70	52	65	59	60	62	52
{2,6}	36	42	51	53	42	51	33	34	34	33
{3,6}	37	41	55	63	41	65	39	41	42	37
{4,6}	56	56	74	75	56	74	52	59	59	52
{5,6}	47	48	59	64	48	59	45	49	51	45
{6,6}	64	69	73	74	69	73	70	74	73	64
{7,6}	37	39	48	54	39	48	38	46	45	37
{1,7}	28	39	54	55	39	54	39	42	43	28
{2,7}	73	75	72	65	75	72	72	69	70	65
{3,7}	40	45	60	67	45	60	51	56	58	40
{4,7}	32	37	34	34	37	34	39	35	38	32
{5,7}	81	82	87	85	82	87	81	80	82	80
{6,7}	33	58	87	97	58	87	60	71	73	33
{7,7}	3	10	17	27	10	17	13	14	16	3
{1,8}	18	50	88	108	50	88	55	67	71	18
{2,8}	49	58	70	74	58	70	62	64	66	49
{3,8}	58	65	84	95	65	84	66	75	77	58
{4,8}	62	57	69	74	57	69	66	71	70	57
{5,8}	56	74	96	96	74	96	63	72	70	56
{6,8}	47	67	86	80	67	86	62	66	68	47
{7,8}	15	46	82	107	46	82	49	63	67	15
{1,9}	13	24	43	56	24	43	29	35	36	13
{2,9}	40	54	73	78	54	73	55	59	59	40
{3,9}	59	64	78	84	64	78	61	64	64	59
{4,9}	66	57	63	74	57	63	62	60	61	57
{5,9}	19	24	36	43	24	36	30	35	35	19
{6,9}	28	40	44	49	40	44	43	47	47	28
{7,9}	27	67	105	122	67	105	71	88	92	27
{1,10}	1	4	9	15	4	9	5	6	7	1
{2,10}	39	39	57	68	39	57	45	50	52	39
{3,10}	45	64	81	75	64	81	56	62	61	45
{4,10}	44	68	88	77	68	88	70	70	73	44
{5,10}	60	64	74	75	64	74	62	67	66	60
{6,10}	27	57	93	109	57	93	56	71	72	27
{7,10}	47	52	56	59	52	56	58	60	61	47

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	43	10	3	4	10	3	9	3	4

f251-f250	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	0	0	0	0	0	0	0	0	0	0
{2,1}	4	6	10	12	6	10	7	7	8	4
{3,1}	24	32	48	58	32	48	35	38	41	24
{4,1}	11	14	23	28	14	23	18	20	21	11
{5,1}	28	39	58	63	39	58	47	54	57	28
{6,1}	39	48	52	58	48	52	45	47	48	39
{7,1}	49	56	72	73	56	72	65	67	70	49
{1,2}	22	29	39	44	29	39	31	31	33	22
{2,2}	41	46	54	58	46	54	53	57	56	41
{3,2}	84	76	85	92	76	85	78	86	82	76
{4,2}	12	12	17	28	12	17	16	17	18	12
{5,2}	33	36	40	42	36	40	42	47	46	33
{6,2}	41	61	90	100	61	90	67	74	74	41
{7,2}	49	52	65	70	52	65	54	59	58	49
{1,3}	55	58	69	78	58	69	63	66	67	55
{2,3}	23	25	34	44	25	34	28	30	35	23
{3,3}	44	54	61	60	54	61	46	49	47	44
{4,3}	15	20	33	43	20	33	20	22	24	15
{5,3}	9	13	16	15	13	16	8	7	8	7
{6,3}	35	37	45	52	37	45	41	46	44	35
{7,3}	46	52	64	67	52	64	60	63	66	46
{1,4}	20	21	28	33	21	26	26	27	29	20
{2,4}	37	48	59	61	48	59	43	41	42	37
{3,4}	18	30	31	31	30	31	27	27	28	18
{4,4}	43	44	50	52	44	50	43	46	45	43
{5,4}	44	45	65	73	45	65	58	60	60	44
{6,4}	46	52	65	74	52	65	46	50	49	46
{7,4}	83	85	85	72	85	85	87	84	83	72
{1,5}	40	54	74	83	54	74	52	63	64	40
{2,5}	46	55	71	76	55	71	58	61	63	46
{3,5}	62	68	86	92	68	86	66	70	70	62
{4,5}	53	44	46	48	44	46	44	48	48	44
{5,5}	33	28	28	29	28	28	29	30	27	27
{6,5}	22	19	19	20	19	19	20	21	20	19
{7,5}	32	34	36	37	34	36	34	36	35	32
{1,6}	32	48	73	87	48	73	50	56	58	32
{2,6}	15	17	17	21	17	17	17	20	19	15
{3,6}	45	43	44	43	43	44	33	37	34	33
{4,6}	43	44	46	50	44	46	40	41	39	39
{5,6}	18	17	19	22	17	19	17	18	17	17
{6,6}	26	24	29	35	24	29	27	30	27	24
{7,6}	24	24	28	28	24	28	23	23	23	23
{1,7}	9	10	10	10	10	10	6	6	6	6
{2,7}	25	23	26	30	23	26	25	31	26	23
{3,7}	29	26	34	38	26	34	29	33	32	26
{4,7}	37	33	34	38	33	34	35	38	35	33
{5,7}	39	37	38	38	37	38	30	38	33	30
{6,7}	26	24	26	28	24	26	26	29	27	24
{7,7}	34	41	44	42	41	44	32	34	33	32
{1,8}	33	64	91	96	64	91	72	81	83	33
{2,8}	53	72	96	103	72	96	74	83	84	53
{3,8}	72	79	80	76	79	80	71	71	74	71
{4,8}	66	69	77	78	69	77	70	72	71	66
{5,8}	4	7	11	14	7	11	7	7	9	4
{6,8}	19	19	21	26	19	21	21	23	22	19
{7,8}	40	39	44	45	39	44	39	43	41	39
{1,9}	18	25	32	36	25	32	20	23	24	18
{2,9}	25	27	35	41	27	35	33	37	38	25
{3,9}	47	67	72	69	67	72	60	59	60	47
{4,9}	30	43	62	63	43	62	39	44	47	30
{5,9}	26	50	80	96	50	80	66	77	77	26
{6,9}	30	40	60	67	40	60	47	53	51	30
{7,9}	19	42	69	80	42	69	43	56	55	19
{1,10}	72	76	84	93	76	84	70	72	73	70
{2,10}	26	31	37	39	31	37	34	33	34	26
{3,10}	12	42	78	96	42	78	46	60	63	12
{4,10}	18	50	78	91	50	78	52	64	66	18
{5,10}	46	51	63	62	51	63	58	60	57	46
{6,10}	46	68	97	105	68	97	85	93	93	46
{7,10}	57	77	79	72	77	79	60	64	60	57

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	49	13	2	2	13	2	13	5	6

Preserving 80% energy within the 16x16 block

f221-f220	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	12	15	16	16	15	16	14	13	14	12
{2,1}	15	13	16	18	13	16	11	13	12	11
{3,1}	33	28	29	31	28	29	27	32	29	27
{4,1}	21	25	28	29	25	28	22	25	25	21
{5,1}	24	23	21	17	23	21	22	24	22	17
{6,1}	37	38	40	40	38	40	33	33	32	32
{7,1}	10	9	10	11	9	10	10	11	11	9
{1,2}	7	9	14	17	9	14	9	10	10	7
{2,2}	20	18	15	13	18	15	12	12	11	11
{3,2}	32	26	27	27	26	27	31	34	32	26
{4,2}	34	30	32	34	30	32	31	33	31	30
{5,2}	6	9	12	13	9	12	6	7	8	6
{6,2}	33	31	38	43	31	38	31	32	31	31
{7,2}	43	41	42	44	41	42	41	47	43	41
{1,3}	27	32	39	43	32	39	29	33	31	27
{2,3}	12	12	12	11	12	12	9	8	8	8
{3,3}	18	20	24	26	20	24	21	22	22	18
{4,3}	16	20	22	20	20	22	17	16	15	15
{5,3}	27	24	25	23	24	25	22	20	20	20
{6,3}	32	29	31	34	29	31	36	37	37	29
{7,3}	15	17	19	19	17	19	20	20	21	15
{1,4}	19	26	40	47	26	40	26	27	29	19
{2,4}	40	40	51	53	40	51	40	42	43	40
{3,4}	4	9	17	22	9	17	7	10	12	4
{4,4}	7	10	10	12	10	10	12	13	13	7
{5,4}	13	18	18	20	18	18	19	20	20	13
{6,4}	26	27	23	23	27	23	28	29	28	23
{7,4}	49	39	42	50	39	42	46	46	46	39
{1,5}	29	30	29	30	30	29	31	31	33	29
{2,5}	19	23	28	33	23	28	19	22	21	19
{3,5}	43	43	45	46	43	45	45	43	44	43
{4,5}	14	16	18	18	16	18	17	17	18	14
{5,5}	44	35	32	34	35	32	38	41	38	32
{6,5}	29	25	26	28	25	26	27	28	28	25
{7,5}	21	17	14	16	17	14	21	20	20	14
{1,6}	27	25	33	41	25	33	30	31	33	25
{2,6}	21	24	29	32	24	29	21	24	24	21
{3,6}	33	32	38	39	32	38	31	33	32	31
{4,6}	44	41	49	47	41	49	52	53	53	41
{5,6}	39	36	38	39	38	38	34	36	35	34
{6,6}	21	18	19	20	18	19	17	19	18	17
{7,6}	19	18	20	20	18	20	18	18	18	18
{1,7}	17	23	55	64	23	55	25	41	42	17
{2,7}	44	48	53	53	48	53	48	48	49	44
{3,7}	22	24	34	37	24	34	30	34	34	22
{4,7}	13	14	23	27	14	23	17	19	20	13
{5,7}	7	7	8	10	7	8	8	8	8	7
{6,7}	10	8	10	13	8	10	12	11	12	8
{7,7}	42	47	54	48	47	54	39	43	41	39
{1,8}	1	2	6	11	2	6	3	4	5	1
{2,8}	31	45	68	74	45	68	51	56	56	31
{3,8}	48	41	49	53	41	49	46	45	46	41
{4,8}	17	28	54	72	28	54	35	43	48	17
{5,8}	12	18	41	57	18	41	24	32	33	12
{6,8}	18	22	46	58	22	46	29	39	37	18
{7,8}	33	35	37	38	35	37	36	37	36	33
{1,9}	11	11	14	15	11	14	16	15	17	11
{2,9}	27	27	26	25	27	26	30	28	27	25
{3,9}	29	39	42	36	39	42	28	25	27	25
{4,9}	33	26	33	46	26	33	42	48	48	26
{5,9}	24	30	40	47	30	40	32	34	35	24
{6,9}	28	30	31	30	30	31	28	28	28	28
{7,9}	27	39	62	72	39	62	41	50	50	27
{1,10}	4	5	14	20	5	14	7	11	11	4
{2,10}	13	19	51	67	19	51	23	37	38	13
{3,10}	13	22	46	56	22	46	25	35	37	13
{4,10}	16	25	58	69	25	58	29	47	46	16
{5,10}	18	29	47	52	29	47	29	38	38	18
{6,10}	35	46	58	57	46	58	41	45	45	35
{7,10}	27	31	48	56	31	48	37	44	45	27

Number of occurrence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	39	18	4	3	18	4	14	6	8

f21-f20	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	16	21	31	32	21	31	19	19	21	16
{2,1}	2	4	6	8	4	6	3	4	4	2
{3,1}	21	25	35	49	25	35	26	29	29	21
{4,1}	30	36	42	43	36	42	41	42	41	30
{5,1}	59	50	52	46	50	52	47	47	44	44
{6,1}	19	33	61	71	33	61	38	47	50	19
{7,1}	46	45	47	44	45	47	42	48	47	42
{1,2}	4	5	8	9	5	8	6	7	8	4
{2,2}	15	23	49	59	23	49	23	35	37	15
{3,2}	43	51	56	54	51	56	43	43	44	43
{4,2}	36	46	61	70	46	61	50	52	52	36
{5,2}	45	46	44	47	46	44	46	46	45	44
{6,2}	28	36	52	57	36	52	36	43	41	28
{7,2}	18	29	49	60	29	49	35	46	47	18
{1,3}	15	14	16	20	14	16	16	22	20	14
{2,3}	14	20	33	36	20	33	15	19	20	14
{3,3}	36	48	57	59	48	57	53	55	54	36
{4,3}	3	4	11	18	4	11	5	8	9	3
{5,3}	20	30	49	58	30	49	30	35	38	20
{6,3}	54	58	64	68	58	64	64	62	68	54
{7,3}	52	57	74	81	57	74	57	64	62	52
{1,4}	39	44	46	46	44	46	47	49	50	39
{2,4}	17	24	48	56	24	48	25	33	34	17
{3,4}	17	32	60	71	32	60	37	46	47	17
{4,4}	16	24	37	40	24	37	23	28	29	16
{5,4}	29	28	38	42	28	38	36	37	39	28
{6,4}	34	38	40	39	38	40	37	41	42	34
{7,4}	29	43	68	72	43	68	45	53	55	29
{1,5}	17	16	21	29	16	21	23	24	26	16
{2,5}	31	33	35	37	33	35	32	33	34	31
{3,5}	35	34	34	35	34	34	37	40	37	34
{4,5}	37	42	46	50	42	46	45	47	47	37
{5,5}	17	18	20	22	18	20	20	21	21	17
{6,5}	34	38	41	40	38	41	31	33	32	31
{7,5}	23	38	63	73	38	63	36	50	51	23
{1,6}	19	24	38	40	24	38	25	25	25	19
{2,6}	22	25	30	33	25	30	26	27	27	22
{3,6}	14	13	13	13	13	13	13	14	13	13
{4,6}	19	19	21	23	19	21	19	20	20	19
{5,6}	34	32	35	37	32	35	32	33	33	32
{6,6}	39	40	45	46	40	45	37	36	36	36
{7,6}	33	38	44	49	38	44	38	41	43	33
{1,7}	15	20	43	49	20	43	21	29	31	15
{2,7}	21	33	60	72	33	60	35	48	48	21
{3,7}	13	17	37	55	17	37	18	27	28	13
{4,7}	25	26	38	39	26	38	34	37	37	25
{5,7}	41	47	57	56	47	57	43	43	43	41
{6,7}	38	47	58	61	47	58	42	46	47	38
{7,7}	21	20	23	27	20	23	24	27	28	20
{1,8}	16	14	17	23	14	17	17	19	19	14
{2,8}	9	9	12	17	9	12	11	14	12	9
{3,8}	10	9	13	19	9	13	13	17	16	9
{4,8}	14	15	22	26	15	22	18	23	22	14
{5,8}	19	19	19	19	19	19	17	19	17	17
{6,8}	22	21	24	27	21	24	23	24	23	21
{7,8}	21	19	22	22	19	22	21	21	21	19
{1,9}	30	42	63	66	42	63	45	55	55	30
{2,9}	3	5	8	9	5	8	5	6	6	3
{3,9}	31	30	33	35	30	33	33	31	32	30
{4,9}	11	11	10	15	11	10	14	12	15	10
{5,9}	6	7	10	10	7	10	7	7	8	6
{6,9}	21	20	26	28	20	26	21	23	23	20
{7,9}	37	42	54	58	42	54	49	53	54	37
{1,10}	5	6	12	17	6	12	8	10	11	5
{2,10}	17	23	31	33	23	31	18	21	20	17
{3,10}	31	34	38	32	34	38	29	31	29	29
{4,10}	10	15	46	63	15	46	21	35	37	10
{5,10}	12	14	39	56	14	39	18	30	31	12
{6,10}	16	21	25	21	21	25	19	19	19	16
{7,10}	15	24	37	47	24	37	26	32	34	15

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	49	15	4	1	15	4	8	2	5

f151-f150	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	27	21	23	26	21	23	23	23	22	21
{2,1}	16	20	27	31	20	27	20	21	21	16
{3,1}	37	32	32	31	32	32	36	39	37	31
{4,1}	31	50	74	80	50	74	48	58	55	31
{5,1}	22	24	27	28	24	27	15	18	17	15
{6,1}	40	34	49	54	34	49	32	35	34	32
{7,1}	43	43	43	41	43	43	46	46	46	41
{1,2}	48	40	47	52	40	47	49	44	43	40
{2,2}	31	30	32	33	30	32	35	36	36	30
{3,2}	43	42	35	33	42	35	35	34	33	33
{4,2}	18	23	29	33	23	29	20	22	23	18
{5,2}	28	29	38	53	29	38	32	35	35	28
{6,2}	54	50	49	46	50	49	42	48	43	42
{7,2}	44	46	58	68	46	58	53	59	58	44
{1,3}	43	52	66	67	52	66	52	55	57	43
{2,3}	11	15	20	22	15	20	12	13	14	11
{3,3}	43	48	53	57	48	53	51	53	52	43
{4,3}	66	58	59	65	58	59	69	70	68	58
{5,3}	53	47	54	55	47	54	54	55	54	47
{6,3}	24	31	37	43	31	37	28	30	30	24
{7,3}	35	37	40	40	37	40	32	34	30	30
{1,4}	40	46	55	50	46	55	46	45	47	40
{2,4}	25	27	30	34	27	30	29	29	31	25
{3,4}	41	46	48	52	46	48	39	47	46	39
{4,4}	4	5	8	11	5	8	6	7	8	4
{5,4}	24	24	24	23	24	24	15	17	16	15
{6,4}	16	15	16	15	15	16	11	13	11	11
{7,4}	27	24	25	26	24	25	24	24	24	24
{1,5}	18	14	15	16	14	15	19	21	19	14
{2,5}	20	25	29	27	25	29	20	21	20	20
{3,5}	49	53	57	54	53	57	52	53	52	49
{4,5}	19	22	28	28	22	28	22	24	23	19
{5,5}	9	9	7	8	9	7	12	12	12	7
{6,5}	16	19	19	19	19	19	14	12	12	12
{7,5}	21	16	19	21	16	19	19	19	18	16
{1,6}	27	35	40	36	35	40	29	33	31	27
{2,6}	11	10	12	14	10	12	12	12	12	10
{3,6}	32	36	37	40	36	37	38	40	41	32
{4,6}	41	38	40	43	38	40	43	46	46	38
{5,6}	44	42	45	48	42	45	40	43	41	40
{6,6}	22	17	19	19	17	19	20	22	19	17
{7,6}	13	19	22	23	19	22	14	14	14	13
{1,7}	26	24	28	28	24	28	21	23	23	21
{2,7}	52	54	55	54	54	55	52	50	52	50
{3,7}	20	36	69	79	36	69	39	53	55	20
{4,7}	29	39	49	51	39	49	36	39	39	29
{5,7}	21	21	25	29	21	25	26	27	27	21
{6,7}	8	7	10	9	7	10	10	9	11	7
{7,7}	19	26	38	44	26	38	26	28	29	19
{1,8}	29	54	80	85	54	80	54	67	67	29
{2,8}	20	24	26	27	24	26	22	23	24	20
{3,8}	37	36	43	43	36	43	36	38	36	36
{4,8}	42	40	43	40	40	43	42	39	39	39
{5,8}	32	33	37	41	33	37	42	43	44	32
{6,8}	34	35	35	35	35	35	29	30	29	29
{7,8}	36	39	45	49	39	45	41	42	43	36
{1,9}	24	29	34	36	29	34	23	26	26	23
{2,9}	32	37	47	47	37	47	31	35	34	31
{3,9}	16	23	28	32	23	28	22	21	23	16
{4,9}	38	37	47	56	37	47	46	49	49	37
{5,9}	33	49	72	72	49	72	51	63	62	33
{6,9}	24	32	58	69	32	58	35	44	46	24
{7,9}	27	37	69	75	37	69	41	52	55	27
{1,10}	15	20	55	68	20	55	26	42	42	15
{2,10}	43	38	49	51	38	49	42	47	46	38
{3,10}	12	22	43	58	22	43	25	35	36	12
{4,10}	36	40	36	35	40	36	34	39	37	34
{5,10}	33	34	41	37	34	41	29	28	30	28
{6,10}	34	38	58	72	38	58	41	47	47	34
{7,10}	26	23	25	28	23	25	32	34	33	23

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	33	17	1	3	17	1	15	5	9

f51-f50	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	13	15	18	19	15	18	18	17	19	13
{2,1}	22	30	35	39	30	35	25	28	28	22
{3,1}	19	27	42	51	27	42	25	30	30	19
{4,1}	33	38	48	50	38	48	35	37	39	33
{5,1}	35	28	36	46	28	36	34	40	38	28
{6,1}	17	27	42	46	27	42	34	39	38	17
{7,1}	26	33	56	70	33	56	44	53	53	26
{1,2}	0	0	0	0	0	0	0	0	0	0
{2,2}	11	14	27	44	14	27	19	24	24	11
{3,2}	9	10	20	25	10	20	11	15	15	9
{4,2}	23	27	50	63	27	50	27	34	34	23
{5,2}	23	36	65	71	36	65	36	49	49	23
{6,2}	33	43	55	58	43	55	40	46	45	33
{7,2}	22	34	71	82	34	71	36	50	51	22
{1,3}	17	21	32	37	21	32	22	26	28	17
{2,3}	48	53	62	66	53	62	58	61	62	48
{3,3}	41	46	58	67	46	58	52	53	53	41
{4,3}	3	4	11	18	4	11	5	8	8	3
{5,3}	20	30	52	65	30	52	28	35	36	20
{6,3}	44	42	51	46	42	51	42	44	45	42
{7,3}	25	38	73	84	38	73	41	54	56	25
{1,4}	12	21	47	61	21	47	29	43	45	12
{2,4}	15	25	46	53	25	46	22	31	32	15
{3,4}	15	21	27	26	21	27	22	23	24	15
{4,4}	12	20	29	37	20	29	20	25	26	12
{5,4}	2	4	6	8	4	6	4	5	6	2
{6,4}	4	5	7	6	5	7	3	3	3	3
{7,4}	32	35	41	42	35	41	32	35	34	32
{1,5}	11	8	9	14	8	9	12	16	14	8
{2,5}	21	24	25	24	24	25	16	17	17	16
{3,5}	19	21	23	24	21	23	18	19	19	18
{4,5}	29	31	33	35	31	33	25	24	24	24
{5,5}	21	22	22	23	22	22	19	19	19	19
{6,5}	17	13	16	17	13	16	17	18	17	13
{7,5}	12	10	13	13	10	13	16	15	16	10
{1,6}	14	13	13	17	13	13	15	18	17	13
{2,6}	12	13	15	16	13	15	13	14	15	12
{3,6}	39	39	38	37	39	38	33	34	33	33
{4,6}	37	37	37	35	37	37	33	37	36	33
{5,6}	36	35	36	37	35	36	29	32	31	29
{6,6}	29	25	23	22	25	23	27	24	25	22
{7,6}	17	13	16	18	13	16	16	17	17	13
{1,7}	27	32	44	40	32	44	30	33	33	27
{2,7}	23	31	49	56	31	49	34	40	41	23
{3,7}	43	49	58	57	49	58	48	55	53	43
{4,7}	10	14	20	22	14	20	12	12	13	10
{5,7}	5	7	12	15	7	12	7	7	8	5
{6,7}	3	4	8	11	4	8	4	4	5	3
{7,7}	19	28	33	37	28	33	19	22	23	19
{1,8}	35	41	48	51	41	48	35	38	38	35
{2,8}	30	35	48	42	35	48	39	43	41	30
{3,8}	15	19	26	29	19	26	20	21	23	15
{4,8}	66	64	64	63	64	64	60	54	56	54
{5,8}	38	48	52	46	48	52	44	47	49	38
{6,8}	24	32	47	56	32	47	33	38	39	24
{7,8}	25	27	34	38	27	34	32	36	35	25
{1,9}	15	18	23	26	18	23	16	17	17	15
{2,9}	41	41	40	38	41	40	43	43	43	38
{3,9}	35	41	43	39	41	43	45	44	44	35
{4,9}	19	16	14	15	16	14	17	17	16	14
{5,9}	20	22	22	24	22	22	17	18	18	17
{6,9}	3	6	16	32	6	16	7	11	13	3
{7,9}	10	10	16	20	10	16	13	14	15	10
{1,10}	5	5	14	21	5	14	6	10	10	5
{2,10}	29	38	47	48	38	47	39	45	45	29
{3,10}	30	34	41	40	34	41	26	27	25	25
{4,10}	34	37	35	31	37	35	30	30	30	30
{5,10}	29	39	44	47	39	44	40	43	43	29
{6,10}	22	36	62	63	36	62	32	42	42	22
{7,10}	16	22	29	29	22	29	21	26	24	16

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	48	10	3	3	10	3	14	6	7

f121-f120	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	16	19	19	19	19	19	20	20	20	16
{2,1}	43	46	45	41	46	45	44	44	43	41
{3,1}	26	20	24	27	20	24	18	21	19	18
{4,1}	8	11	17	23	11	17	19	21	22	8
{5,1}	22	21	21	26	21	21	25	26	26	21
{6,1}	25	26	34	41	26	34	25	27	27	25
{7,1}	63	70	70	65	70	70	58	60	58	58
{1,2}	12	15	13	13	15	13	13	14	13	12
{2,2}	17	15	15	16	15	15	16	17	16	15
{3,2}	51	46	47	49	46	47	45	47	46	45
{4,2}	29	29	28	28	29	28	30	31	29	28
{5,2}	31	38	38	38	38	38	29	30	31	29
{6,2}	39	47	45	43	47	45	47	47	48	39
{7,2}	13	11	12	11	11	12	8	9	7	7
{1,3}	19	25	35	39	25	35	24	28	28	19
{2,3}	37	44	53	57	44	53	44	46	49	37
{3,3}	23	24	34	40	24	34	26	31	30	23
{4,3}	22	20	22	22	20	22	22	22	23	20
{5,3}	31	32	35	37	32	35	29	30	31	29
{6,3}	34	29	29	30	29	29	30	34	31	29
{7,3}	10	10	12	12	10	12	8	9	8	8
{1,4}	17	22	27	30	22	27	25	28	27	17
{2,4}	14	12	17	23	12	17	21	20	21	12
{3,4}	17	16	21	25	16	21	20	18	21	16
{4,4}	27	27	26	27	27	26	29	30	29	26
{5,4}	19	19	22	23	19	22	18	20	19	18
{6,4}	45	47	52	48	47	52	42	45	43	42
{7,4}	40	39	38	38	39	38	40	41	41	38
{1,5}	21	22	27	27	22	27	23	22	23	21
{2,5}	29	34	49	55	34	49	35	38	37	29
{3,5}	33	28	28	29	28	28	29	28	27	27
{4,5}	42	41	51	55	41	51	48	51	50	41
{5,5}	43	44	46	46	44	46	44	42	44	42
{6,5}	26	25	21	18	25	21	25	23	23	18
{7,5}	26	27	35	42	27	35	34	37	37	26
{1,6}	35	35	41	42	35	41	36	35	35	35
{2,6}	24	28	34	36	28	34	21	20	21	20
{3,6}	17	20	30	33	20	30	16	18	18	16
{4,6}	34	34	48	49	34	48	28	31	32	28
{5,6}	29	27	35	39	27	35	26	30	30	26
{6,6}	39	41	45	43	41	45	45	47	48	39
{7,6}	20	20	25	27	20	25	18	21	21	18
{1,7}	18	24	35	33	24	35	25	26	27	18
{2,7}	45	48	47	44	48	47	48	45	45	44
{3,7}	27	28	31	34	28	31	32	33	34	27
{4,7}	19	20	18	17	20	18	21	20	21	17
{5,7}	55	50	54	53	50	54	50	52	50	50
{6,7}	23	32	61	67	32	61	35	45	47	23
{7,7}	3	5	12	17	5	12	7	10	10	3
{1,8}	13	27	61	78	27	61	31	42	45	13
{2,8}	33	38	45	46	38	45	38	40	41	33
{3,8}	36	40	56	60	40	56	43	50	50	36
{4,8}	36	32	44	46	32	44	41	45	44	32
{5,8}	36	47	61	64	47	61	36	44	42	36
{6,8}	35	45	56	53	45	56	38	39	39	35
{7,8}	13	21	55	75	21	55	26	40	41	13
{1,9}	9	12	27	36	12	27	15	20	21	9
{2,9}	26	32	43	45	32	43	31	32	33	26
{3,9}	34	40	49	51	40	49	35	39	37	34
{4,9}	34	31	37	45	31	37	35	33	34	31
{5,9}	7	12	22	26	12	22	15	18	19	7
{6,9}	18	22	22	24	22	22	23	26	25	18
{7,9}	24	38	72	86	38	72	41	57	58	24
{1,10}	1	2	6	11	2	6	3	4	5	1
{2,10}	27	22	33	41	22	33	27	29	31	22
{3,10}	26	39	50	48	39	50	36	41	39	26
{4,10}	33	47	53	48	47	53	48	45	47	33
{5,10}	40	42	47	46	42	47	42	43	42	40
{6,10}	20	33	64	78	33	64	32	43	43	20
{7,10}	29	29	33	36	29	33	35	35	36	29

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	36	13	6	6	13	6	15	3	6

f251-f250	Number of DWT coefficients (preserving 90% energy)									Minimum number of coefficients
16x16 block	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2	
{1,1}	0	0	0	0	0	0	0	0	0	0
{2,1}	3	4	6	7	4	6	4	5	5	3
{3,1}	17	20	30	39	20	30	23	26	27	17
{4,1}	8	10	15	21	10	15	11	12	14	8
{5,1}	20	26	39	42	26	39	29	37	39	20
{6,1}	24	29	32	33	29	32	26	26	29	24
{7,1}	35	40	48	43	40	48	43	45	45	35
{1,2}	17	18	26	32	18	26	21	20	22	17
{2,2}	23	25	30	33	25	30	27	29	29	23
{3,2}	48	46	53	59	46	53	49	53	50	46
{4,2}	9	8	11	18	8	11	11	11	12	8
{5,2}	21	25	27	25	25	27	25	29	28	21
{6,2}	27	38	59	66	38	59	40	45	45	27
{7,2}	28	30	36	40	30	36	31	34	31	28
{1,3}	34	34	44	52	34	44	40	45	44	34
{2,3}	17	15	22	28	15	22	18	20	22	15
{3,3}	26	33	37	37	33	37	24	26	25	24
{4,3}	9	12	22	31	12	22	13	14	15	9
{5,3}	6	8	11	8	8	11	5	5	5	5
{6,3}	22	21	25	29	21	25	23	25	24	21
{7,3}	35	31	39	39	31	39	37	38	40	31
{1,4}	13	13	18	23	13	18	18	19	20	13
{2,4}	18	29	39	41	29	39	28	25	25	18
{3,4}	13	17	18	17	17	18	17	17	18	13
{4,4}	23	23	26	27	23	26	23	25	23	23
{5,4}	29	28	39	43	28	39	36	37	37	28
{6,4}	31	34	41	48	34	41	29	28	28	28
{7,4}	52	54	55	50	54	55	59	53	53	50
{1,5}	26	34	46	52	34	46	34	40	40	26
{2,5}	29	34	42	46	34	42	34	37	37	29
{3,5}	39	40	57	63	40	57	40	40	42	39
{4,5}	31	26	28	31	26	28	25	27	27	25
{5,5}	22	20	19	18	20	19	16	17	16	16
{6,5}	15	14	14	13	14	14	14	14	14	13
{7,5}	21	22	24	24	22	24	21	22	21	21
{1,6}	22	32	45	55	32	45	31	34	35	22
{2,6}	8	8	11	12	8	11	11	12	12	8
{3,6}	31	30	30	29	30	30	25	26	25	25
{4,6}	29	29	32	33	29	32	28	28	27	27
{5,6}	15	14	14	15	14	14	14	14	14	14
{6,6}	15	14	16	19	14	16	16	19	18	14
{7,6}	13	11	15	17	11	15	14	15	15	11
{1,7}	5	5	7	7	5	7	3	4	4	3
{2,7}	12	12	14	19	12	14	12	16	14	12
{3,7}	15	14	17	20	14	17	18	20	19	14
{4,7}	23	19	19	21	19	19	21	23	22	19
{5,7}	25	23	25	27	23	25	20	24	22	20
{6,7}	18	17	18	18	17	18	18	19	19	17
{7,7}	19	23	27	24	23	27	17	19	18	17
{1,8}	24	39	63	65	39	63	45	53	56	24
{2,8}	31	45	65	70	45	65	43	53	53	31
{3,8}	50	47	48	45	47	48	45	44	45	44
{4,8}	44	45	49	52	45	49	46	47	48	44
{5,8}	2	3	7	9	3	7	3	4	4	2
{6,8}	13	12	14	17	12	14	15	16	16	12
{7,8}	25	27	31	31	27	31	26	28	27	25
{1,9}	15	16	21	23	16	21	12	14	14	12
{2,9}	19	19	23	25	19	23	21	23	23	19
{3,9}	29	39	46	43	39	46	34	33	34	29
{4,9}	19	24	36	37	24	36	21	23	23	19
{5,9}	23	29	54	66	29	54	37	47	49	23
{6,9}	20	24	33	34	24	33	27	29	28	20
{7,9}	17	24	47	55	24	47	26	34	34	17
{1,10}	45	50	56	62	50	56	44	47	47	44
{2,10}	16	19	24	23	19	24	20	20	20	16
{3,10}	11	19	51	69	19	51	23	37	40	11
{4,10}	16	23	54	64	23	54	27	40	41	16
{5,10}	30	35	42	39	35	42	36	36	33	30
{6,10}	33	44	66	71	44	66	57	62	63	33
{7,10}	33	46	48	44	46	48	32	36	32	32

Number of occurence as the best wavelet	db1	db2	db3	db4	sym2	sym3	sym4	coif1	coif2
	41	20	3	3	20	3	16	5	10

Appendix C. MATLAB Code for Wavelet-DCT

dwtmtx.m

```
function DWTmtx=dwtmtx(N,wavelet)
J=1;

[h0 h1 g0 g1]=wfilters(wavelet);
limitH = length(h0);

if N < limitH
    DWTcoeffsN=x;
    return
end

if mod(N,2^J)>0
    remainder = rem(N,2^J);
    L=2^J-remainder;
    L=N+L;
    x(L)=0;
    N=length(x);
end;

%x=wshift('1',x,limitH-1);
h0mtx = convmtx(fliplr(h0),N);
ppdh0 = h0mtx(1:end,end - limitH+2:end);
h0mtx = h0mtx(1:end,1:end-limitH+1);
h0mtx(1:end,1:limitH-1) =
h0mtx(1:end,1:limitH-1)+ppdh0;

h1mtx = convmtx(fliplr(h1),N);
ppdh1 = h1mtx(1:end,end - limitH+2:end);
h1mtx = h1mtx(1:end,1:end-limitH+1);
h1mtx(1:end,1:limitH-1) =
h1mtx(1:end,1:limitH-1)+ppdh1;

DSMPLmtx = eye(N);
DSMPLmtx = DSMPLmtx(1:2:end, 1:end);

DWTmtx = [DSMPLmtx*h0mtx;DSMPLmtx*h1mtx];
```

rle.m

```
function Ldata = rle(x)
% data = rle(x) (de)compresses the data with
the RLE-Algorithm
%   Compression:
%      if x is a numbervector data{1} contains
the values
%      and data{2} contains the run lenths
%
%   Decompression:
%      if x is a cell array, data contains the
uncompressed values
%
%      Version 1.0 by Stefan Eireiner (<a
%href="mailto:stefan-
e@web.de?subject=rle">stefan-e@web.de</a>)
%      based on Code by Peter J. Acklam
%      last change 14.05.2004

if iscell(x) % decoding
       i = cumsum([ 1 x{2} ]);
       j = zeros(1, i(end)-1);
       j(i(1:end-1)) = 1;
       data = x{1}(cumsum(j));
else % encoding
       if size(x,1) > size(x,2), x = x'; end
% if x is a column vector, tronspose
    i = [ find(x(1:end-1) ~= x(2:end))
length(x) ];
       data{2} = diff([ 0 i ]);
       data{1} = x(i);
    Ldata = length(data{1}) + length(data{2});
End
```

energy.m

```
function y = energy(x);

yt = 0;
for i = 1:( size(x,1)*size(x,2) )
    yt = yt + x(i)*x(i);
end
y = yt;
```

fwdDWTN.m

```
function DWTcoeffsN=fwdDWTN(x,wavelet,J)
[row col] = size(x);
if row ==1 |col ==1
    if col>1
        x=x';
    end
end

[h0 h1 g0 g1]=wfilters(wavelet);
limitH = length(h0);
N = length(x);
if N < limitH
    DWTcoeffsN=x;
    return
end

if mod(N,2^J)>0
    remainder = rem(N,2^J);
    L=2^J-remainder;
    L=N+L;
    x(L)=0;
    N=length(x);
end;
%x=wshift('1',x,limitH-1);
h0mtx = convmtx(fliplr(h0),N);
ppdh0 = h0mtx(1:end,end - limitH+2:end);
h0mtx = h0mtx(1:end,1:end-limitH+1);
h0mtx(1:end,1:limitH-1) =
h0mtx(1:end,1:limitH-1)+ppdh0;

h1mtx = convmtx(fliplr(h1),N);
ppdh1 = h1mtx(1:end,end - limitH+2:end);
h1mtx = h1mtx(1:end,1:end-limitH+1);
h1mtx(1:end,1:limitH-1) =
h1mtx(1:end,1:limitH-1)+ppdh1;

DSMPLmtx = eye(N);
DSMPLmtx = DSMPLmtx(1:2:end, 1:end);

DWTmtx = [DSMPLmtx*h0mtx;DSMPLmtx*h1mtx];
DWTcoeffsN = DWTmtx*x;

if J>1
input = DWTcoeffsN(1:ceil(N/2));
DWTcoeffsN(1:ceil(N/2))=
fwdDWTN(input,wavelet,J-1);
end
```

fwdDWTN2.m

```
function DWTcoeffsN=fwdDWTN2(x,wavelet,J)

[row col] = size(x);
if row ==1 |col ==1
    if col>1
        x=x';
    end
end

[h0 h1 g0 g1]=wfilters(wavelet);
limitH = length(h0);
N = length(x);
if N < limitH
    DWTcoeffsN=x;
    return
end

if mod(N,2^J)>0
    remainder = rem(N,2^J);
    L=2^J-remainder;
    L=N+L;
    x(L)=0;                 %x(N+1:L)=0; (regina
edit)
    N=length(x);
end;
%x=wshift('1',x,limitH-1);

h0mtx = convmtx(fliplr(h0),N);
ppdh0 = h0mtx(1:end,end - limitH+2:end);
h0mtx = h0mtx(1:end,1:end-limitH+1);
h0mtx(1:end,1:limitH-1) =
h0mtx(1:end,1:limitH-1)+ppdh0;

h1mtx = convmtx(fliplr(h1),N);
ppdh1 = h1mtx(1:end,end - limitH+2:end);
h1mtx = h1mtx(1:end,1:end-limitH+1);
h1mtx(1:end,1:limitH-1) =
h1mtx(1:end,1:limitH-1)+ppdh1;

DSMPLmtx = eye(N);
DSMPLmtx = DSMPLmtx(1:2:end, 1:end);

DWTmtx = [DSMPLmtx*h0mtx;DSMPLmtx*h1mtx];
DWTcoeffsN = DWTmtx*x*DWTmtx';

if J>1
input = DWTcoeffsN(1:ceil(N/2),1:ceil(N/2));
DWTcoeffsN(1:ceil(N/2),1:ceil(N/2))=
fwdDWTN2(input,wavelet,J-1);
end
```

recombine.m

```
function Y = recombine(Di,wavelet,J)
% maxlevels = 3;
N=length(Di);
% J=1;
[row col] = size(Di);
if row==1|col ==1
    if col>1
        Di=Di';
    end
end

if J>1
for levels=J:(-1):2
WA = dwtmtx(N/(2^(levels-1)),wavelet);
input = Di(1:ceil(N/(2^(levels-
1))),1:ceil(N/(2^(levels-1))));
Di(1:ceil(N/(2^(levels-
1))),1:ceil(N/(2^(levels-1))))= WA'*input*WA;
end
end
```

histA.m

```
function TL = histA(d,drop)

Wbrshp = reshape(d,1,size(d,1)*size(d,2));
E = energy(abs(Wbrshp));
sWb = sort(abs(Wbrshp));

cDrop = 0;
EDrop = 0;

    for j = 1:size(sWb,1)*size(sWb,2)
        EDrop = EDrop + (sWb(j)^2);
        if ( (EDrop/E) < drop )
            cDrop = cDrop + 1;
            TL = sWb(cDrop);

    end
```

fwdDWPTN2.m

```
function DWTcoeffsN = fwdDWPTN2(x,wavelet,J)

[row col] = size(x);
if row ==1 |col ==1
    if col>1
        x=x';
    end
end

N = length(x);

DWTmtx = dwtmtx(N,wavelet);
DWTcoeffsN = DWTmtx*x*DWTmtx';

if J>1

LL = DWTcoeffsN(1:ceil(N/2),1:ceil(N/2));
LH = DWTcoeffsN(1:ceil(N/2),ceil(N/2)+1:end);
HL = DWTcoeffsN(ceil(N/2)+1:end,1:ceil(N/2));
HH =
DWTcoeffsN(ceil(N/2)+1:end,ceil(N/2)+1:end);

DWTcoeffsN(1:ceil(N/2),1:ceil(N/2))=
fwdDWPTN2(LL,wavelet,J-1);
DWTcoeffsN(1:ceil(N/2),ceil(N/2)+1:end)=
fwdDWPTN2(LH,wavelet,J-1);
DWTcoeffsN(ceil(N/2)+1:end,1:ceil(N/2))=
fwdDWPTN2(HL,wavelet,J-1);
DWTcoeffsN(ceil(N/2)+1:end,ceil(N/2)+1:end)=
fwdDWPTN2(HH,wavelet,J-1);

End
```

threshold.m

```
function Di = threshold(d,TL,J)

Di = d;

for k=1:size(d,1)*size(d,2)
    if abs(Di(k))<TL
        Di(k)=0;
    end
end
% Di(1:(size(d,1)/2)*(size(d,2)/2)) =
d(1:(size(d,1)/2)*(size(d,2)/2));
% Di(1:size(d,1)/4,1:size(d,2)/4) =
d(1:size(d,1)/4,1:size(d,2)/4);
Di(1:size(d,1)/(2^J),1:size(d,2)/(2^J)) =
d(1:size(d,1)/(2^J),1:size(d,2)/(2^J));

R = rcbmtx(N);
Y = R*Di*R';
```

Appendix D. MATLAB Code for Data Generation

frame_reordered.m

```
aviobj = avifile('C:\Program
Files\MATLAB\R2006a\work\regina\fadct_simulink_model_16by16\frame_ordered3.avi','fps',30)
% I-frame
for k=1:9:283
    frame{k} = aviread('vipmen.avi',k);
    yuv{k} = (frame{k}.cdata);
    y{k} = yuv{k}(1:112,1:160,:);
end

% P-frame
p1 = 4:9:283;
p2 = 7:9:283;
p = [p1;p2 0];
p = reshape(p,1,size(p,1)*size(p,2));
p = p(1:63);

for k=p
        refframe{k} = aviread('vipmen.avi',k-3);
    nxtframe{k} = aviread('vipmen.avi',k);
    yuv{k} = (nxtframe{k}.cdata)-(refframe{k}.cdata);
    y{k} = yuv{k}(1:112,1:160,:);
end

% B-frame
b1 = 2:3:283;
b2 = 3:3:283;
b = [b1;b2];
b = reshape(b,1,size(b,1)*size(b,2));
b = b(1:size(b,1)*size(b,2));

for k=b
        refframe{k} = aviread('vipmen.avi',k-1);
    nxtframe{k} = aviread('vipmen.avi',k);
    yuv{k} = (nxtframe{k}.cdata)-(refframe{k}.cdata);
    y{k} = yuv{k}(1:112,1:160,:);
end

for k=1:283
aviobj = addframe(aviobj,y{k});
end
aviobj = close(aviobj);
```

DTI.m

```
frame220 = aviread('vipmen.avi',220); %reference frame
frame221 = aviread('vipmen.avi',221);
yuv220 = rgb2ycbcr(frame220.cdata);
yuv221 = rgb2ycbcr(frame221.cdata);
y220 = yuv220(:,:,1);
y221 = yuv221(:,:,1);
y220 = y220(1:112,1:160);
y221 = y221(1:112,1:160);
fileref = 'D:\Test_Images\y220.bmp';
filenex = 'D:\Test_Images\y221.bmp';
imwrite(y220,fileref,'bmp');
imwrite(y221,filenex,'bmp');

% d = y221 - y220;
% image = double(d);
```