National Library of Canada

Acquisitions and Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada

Direction des acquisitions et des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Canada

Wavelets for Image Compression

Mrinal Kumar Mandal

A THESIS

submitted to the School of Graduate Studies and Research

in Partial Fulfillment of the Requirements

for the Degree of

MASTER OF APPLIED SCIENCE

in

Electrical Engineering

Ottawa-Carleton Institute of Electrical Engineering

Department of Electrical Engineering

Faculty of Engineering

University of Ottawa

OTTAWA, ONTARIO K1N 6N5

ISBN 0-612-04880-2

Canada

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

To my family

I hereby declare that I am the sole author of this thesis.

I authorize the University of Ottawa to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Mrinal Kumar Mandal

I further authorize the University of Ottawa to reproduce this thesis by photocopying or other means, in total or part, at the request of other institutions or individuals for the purpose of scholarly research.

Mrinal Kumar Mandal

Acknowledgements

It is my pleasure to acknowledge and thank all persons who have influenced me in the course of this research. First, I express my gratitude to both of my advisors Dr. Tyseer Aboulnasr and Dr. Sethuraman Panchanathan for introducing me to the exciting field of wavelets and image compression and for their continued support and encouragement during my thesis work.

Special thanks are due to Dr. R. A. Goubran, Dept. of Systems and Computer Engineering, Carleton University and Dr. J. F. Rivest, Dept. of Electrical Engineering, University of Ottawa, whose valuable comments have enhanced the readability of this thesis.

I would like to thank all the past and present members of the Multimedia Communications Research Laboratory, especially Grant Henderson, M. B. Brahmanandam, Fayez Idris, Nael Hirzulla and Amit Jain for their help and cooperation.

My special thanks are due to all the support staff members of Electrical Engineering for their help, especially Michele Roy and Lucette Lepage.

The generous financial support of Canadian Commonwealth Fellowship Plan and NSERC that made this research possible, is also gratefully acknowledged.

I am truly grateful to my beloved wife Rupashri and my family for their consistent support, without which this work would not have been possible.

Abstract

Wavelets are becoming increasingly important in image compression applications because of its flexibity in representating nonstationary signals. To achieve a high compression ratio, the wavelet has to be adapted to the image. Current techniques use exhaustive search procedures which are computationally intensive to find the optimal basis (type/order/tree) for the image to be coded. In this thesis, we have carried out extensive performance analysis of various wavelets on a wide variety of images. Based on the investigation, we propose some guidelines for searching for the optimal wavelet (type/order) based on the overall activity (measured by the spectral flatness) of the image to be coded. These guidelines will provide the degree of improvement that can be achieved by using the "optimal" over "standard" wavelets. The proposed guidelines can be used to find a good initial guess for faster convergence when searching for optimal wavelet is essential. We propose a wave packet decomposition algorithm based on the local transform gain of the wavelet decomposed bands. The proposed algorithm provides good coding performance at significantly reduced complexity.

Most practical coders are designed to minimize the mean square error (MSE) between the original and reconstructed image. It is known that at high compression ratio, MSE does not correspond well to the subjective quality of the image. In this thesis, we propose an image adaptive coding algorithm which tries to minimize the MSE weighted by the visual importance of various wavelet bands. It has been observed that the proposed algorithm provides a better coding performance for a wide variety of images.

Contents

List of Figures

List of Tables

Abbreviations

ATM	Asynchronous Transfer Mode
CCIR	Comite Consultatif International de Radiodiffusion
CCITT	International Consultative Committee for Telephone and Telegraph
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DST	Discrete Sine Transform
DPCM	Differential Pulse Code Modulation
DTFT	Discrete Time Fourier Transform
DWT	Discrete Wavelet Transform
FFT	Fast Fourier Transform
HDTV	High Definition Television
HVS	Human Visual System
ISDN	Integrated Services Distributed Network
JPEG	Joint Photographic's Expert Group
KLT	Karhunen-Loeve Transform
MPEG	Motion Picture's Expert Group
MSE	Mean Square Error
PCM	Pulse Code Modulation
PR	Perfect Reconstruction
PSNR	Peak Signal to Noise Ratio
QMF	Quadrature Mirror Filter
SAR	Synthetic Aperture Radar
SBC	Subband Coding
SBD	Subband Decomposition
STFT	Short Time Fourier Transform
TCG	Transform Coding Gain
bpp	bit-per-pixel
pdf	probability density function

Chapter 1

Introduction

In the last decade, there has been an enormous increase in the storage and transmission of information. Digital images and video are important tools for communicating visual information. Usually, digital images require large channel bandwidth for transmission and large storage space for archival. In addition, due to the large number of pixels in a high resolution image, manipulation of digital images is feasible only with low complexity algorithms [107]. Because of these reasons, a reliable and fast compression technique is very desirable. Recently, with the advent of broadband networks such as ISDN, ATM *etc*, and low cost VLSI chips, many new application areas such as multimedia communications, high definition television (HDTV) *etc*, are becoming feasible. These new application areas require high compression ratios for storage and transmission.

To meet the growing need for image compression and for ensuring compatibility, the International Standard Organization (ISO) has recently proposed the JPEG and MPEG standards for image and video compression, respectively. These standards are based on discrete cosine transform (DCT) of small image blocks and are very effective in reducing the spatial redundancy in images. However, DCT coding has the drawbacks of blockiness and aliasing distortion in the reconstructed image at high compression ratios.

Recently, wavelet theory has emerged as a powerful technique for nonstationary signal analysis. Discrete wavelet transform (DWT) was first applied to image coding by Mallat [61]. The implementation of DWT is very similar to subband coding. However, subband coding emphasizes on improving the frequency selectivity of the filters whereas wavelets emphasizes the smoothness properties of the basis functions. One of the main contributions of wavelet theory is to relate the discrete time filter with the theory of continuous time function space. Combining the advantages of multiresolution analysis and transform coding, wavelets offer a wide variety of useful features [113].

- Computational complexity of O(N), where N is the number of pixels
- Good time localization of the basis functions
- Efficient VLSI implementation
- Images without blocking artifacts
- Lower aliasing distortion
- Inherent scalability

In addition, it has also been shown that wavelet based methods are near optimal (in a particular mathematical sense) within a large class of stable, transform based nonlinear methods of image compression [27]. As a result, wavelets are being used extensively in the area of the image compression in recent years and have been seen to provide good coding performance. However, their full potential is yet to be explored and requires more investigation concerning decomposition algorithms, quantization schemes and subjective quality of the reconstructed image. Some of the unresolved problems are : What is the best wavelet for a given image ? How to adapt to the changing statistics of an image ? What is the best tree decomposition structure ? How to incorporate HVS models efficiently in an wavelet based coding algorithm ?

In this thesis, some of the above mentioned problems have been addressed. We have investigated the choice of wavelets (type/order) and tree decomposition structures for

image compression. An efficient HVS adapted coding algorithm has also been proposed.

1.1 Outline of Thesis

This thesis discusses the theory of wavelets and its application to image coding. In Chapter 2, a review of image compression techniques is presented. We discuss the concept of entropy, rate-distortion function and distortion measures. Next, we review some lossless coding techniques such as Huffman, arithmetic and run-length coding. This is followed by a review of lossy compression techniques such as predictive, transform, subband, vector quantization and fractal coding. Finally, we briefly describe the JPEG standard for still image compresion.

In Chapter 3, we review the theory and implementation of wavelets. First, we present the wavelet theory from the perspective of short-time Fourier transform. We then relate wavelets with multiresolution analysis. The recursive implementation of discrete wavelet transform is discussed next. This is followed by a review of the parameterization of orthogonal wavelets. We then discuss two important properties of wavelets namely, regularity and vanishing moments. The design techniques for two popular wavelet faimilies - Daubechies and Coiflet families are considered followed by the extension of 1-D DWT to 2-D DWT. Finally, the computational complexity of DWT is reviewed.

In Chapter 4, we describe a wavelet based image coding scheme. We compare various quantization schemes and choose a simple yet efficient scheme for further investigation. The computational complexity and coding performance of the wavelet coder are compared with those of JPEG coder.

In Chapter 5, we briefly review the current research on optimal choice of wavelets. The lists of all the wavelets and images used in this thesis are provided next. This is followed by a comparison of the relative performance of three known wavelet families

based on their filter characteristics, coding gain, objective and subjective coding performance. We relate the coding performance with the filter order and image activity. The coding performance of wavelet families are then compared with the optimal 4, 6 and 8 tap wavelets for various test images. Next, we propose a simple image adaptive tree decomposition algorithm based on the local transform gain. Finally, the performance analysis of complex symmetrical wavelets in image coding is presented.

In Chapter 6, a brief review of the Fourier based HVS model is presented first. This is followed by a discussion of quantization scheme reported in [27]. We discuss in detail the relative importance of wavelet basis functions of various wavelet bands. We then propose an image adaptive wavelet coder which is optimal in the visually weighted mean square sense. This is followed by simulation results and conclusion.

The main body of the thesis is followed by five appendices. Appendix-A tabulates important properties of scaling and wavelet functions discussed in Chapter 3. Appendix-B provides the exact formulae for the parameterization for 6 and 8 tap wavelets. Appendix-C provides the relationship among the zeroes of wavelet filter, the vanishing derivatives and smoothness of the basis functions. Appendix-D tabulates refinement coefficients of various wavelets used in this thesis. Finally, Appendix-E shows the test images used in the simulations.

1.2 Main Contributions

The main contributions of this thesis are summarized below :

- A simple sub-optimal quantization scheme (section 4.2.3)
- Relationship between the choice of wavelets (type/order) and image spectral activity (section 5.3-5.7).
- An efficient irregular tree decomposition algorithm (section 5.8).
- An optimal visually-weighted quantization scheme (section 6.3).

Chapter 2

Review of Image Compression Techniques

The term image compression refers to the process of reducing the amount of data needed to represent an image with an acceptable subjective quality. This is usually done by reducing the statistical redundancy present in an image. In addition, the properties of human visual system can be exploited to further increase the compression ratio. Most of the image compression techniques are based on information theory first formulated by Shannon [92]

In this chapter, we first review the basic concepts of information theory as applied to image compression *i.e.,* its source coding part. We define the important parameters - entropy, rate and distortion measure. We then discuss the distortion measures which are often used to quantify the quality of a reconstructed image. This is followed by a review of lossless image coding techniques such as Huffman coding, run-length coding and arithmetic coding, which preserve all the information present in an image. Next, we discuss some lossy techniques - predictive, transform, subband, vector quantization and fractal coding which provide a better coding performance compared to lossless techniques. We then compare the performance of the various coding techniques. Finally, we briefly describe the JPEG standard for still image compression and conclude the chapter by summarizing the reviews.

2.1 Source Coding Theory

Source coding theory deals with the compression of data generated by an information source that emits a sequence of symbols chosen from a finite alphabet. In source coding theory, *entropy* and *rate-distortion* functions are the two most fundamental concepts [25]. Entropy provides a measure of information contained in the source data and therefore determines the minimum average bit rate required for perfect reconstruction of the source symbols. Rate distortion theory provides a lower bound on the average bit rate for a given distortion in the reconstructed symbols. The concepts of entropy and rate distortion are detailed in the following sections.

2.1.1 Entropy

Let us consider a discrete memoryless information source S, in which successive symbols are being produced. We assume that the source has an alphabet $S = \{s_1, s_2, \ldots\ldots s_K\}$ with associated probabilities of occurence $\{p(s_1), p(s_2), \ldots\ldots p(s_K)\}$ and the symbols are statistically independent. The average information provided by a symbol is related to the reciprocal of its probability and is given by

$$I(s_i) = \log\left(\frac{1}{p(s_i)}\right) \tag{2.1}$$

The *entropy* is defined as the average amount of information per source symbol, which can be expressed as

$$H(S) = \sum_{i=1}^{K} p(s_i) I(s_i) = -\sum_{i=1}^{K} \log_2 p(s_i) \tag{2.2}$$

In the above equation, the base of the logarithm has been taken as 2, so that the information can be expressed in bits per symbol.

It is often useful to deal with large blocks of symbols, rather than individual symbols. We can group the output of the source S into blocks of N symbols and assume that the

symbols are being generated by a source S^N with an alphabet of size K^N. The source S^N is called the N-th extension of the source S [91]. We can consider an information source which is generating images of size $N \times N$ pixels, where each pixel is quantized to K grey levels. A total of $K^{N \times N}$ possible image patterns can be generated by such a source. Let the probability of a specific image pattern s be given by $p(\mathrm{s})$. Here, we are treating each image as a symbol of $K^{N \times N}$ size alphabet. The entropy *i.e.*, the average information of the source can be calculated as

$$H(S) = -\sum_{all\ \mathrm{s}} p(\mathrm{s})\ \log_2 p(\mathrm{s}) \quad bits\ /\ symbol \tag{2.3}$$
$$= -\frac{1}{N \times N} \sum_{all\ \mathrm{s}} p(\mathrm{s})\ \log_2 p(\mathrm{s}) \quad bits\ /\ pixel$$

Let S be an ergodic source with an alphabet of size K and entropy H(S). Consider encoding blocks of N source symbols at a time into binary codewords. Then according to Shannon's noiseless source coding theorem, for any $\delta > 0$, it is possible by choosing N large enough to construct a code in such a way that the average number of bits per original source symbol, $\overline{R}$, satisfies

$$H(S) \leq \overline{R} < H(S) + \delta \tag{2.4}$$

The above theorem states that a source can be coded losslessly with a number of bits arbitrarily close to its entropy, but the number of bits cannot be less than the entropy. However, to achieve a bit-rate close to entropy, one has to encode higher extensions of the source.

It can be shown that the source entropy is bounded by 0 and $\log_2 K$ [49], *i.e.*,

$$0 \leq H(S) \leq \log_2 K \tag{2.5}$$

The left side equality holds if $p(\mathrm{s})$ is zero for all s except one, in which case the source is totally predictable. The right side equality holds when every source symbol s has the same probability. The redundancy of the source is defined as,

$$redundancy = \log_2 K - H(S) \tag{2.6}$$

If every pixel in the image is statistically independent of the others, then the source probability $p(\mathbf{s})$ can be expressed as:

$$p(\mathbf{s}) = \prod_{i,j=0}^{N-1} p_{ij}(s_{ij}) \tag{2.7}$$

Here, p_{ij} represents the probability that the pixel S_{ij} of the image source S, has a value equal to s_{ij}. In this case $H(S)$ can be expressed as,

$$H(S) = -\frac{1}{N \times N} \sum_{i,j=0}^{N-1} \sum_{all\ s_{ij}} p_{ij}(s_{ij}) \log_2 p_{ij}(s_{ij}) \quad bits/pixel \tag{2.8}$$

In practice, the statistical information of an image $p(\mathbf{s})$, cannot be easily measured or modeled and therefore, the true entropy of the image is, in general, very difficult to obtain.

Hence, a simpler measure- the first order entropy $H^1(S)$, which is defined on a pixel-by-pixel basis is oftenly used. $H^1(S)$ is defined as:

$$H^1(S) = -\sum_{k=0}^{N-1} P_k \log_2 P_k \tag{2.9}$$

where P_k is the probability of the occurence of the gray level K. If the pixels of the image are identically and independently distributed (*i.i.d*), *i.e.*,

$$p_{ij}(s_{ij}) = p_{kl}(s_{kl}) , \qquad \text{for } s_{ij} = s_{kl} \tag{2.10}$$

then the entropy $H(S)$ of the image is equal to the first order entropy, $H^1(S)$,

$$H(S) = H^1(S) = -\sum_{k=0}^{N-1} p_k \log_2 p_k \tag{2.11}$$

In this case, P_k is simply the probability of the occurrence of the kth grey level value. The first order entropy is often called *memoryless entropy* as it provides the minimum bit rate required for lossless reproduction of an image without exploiting the interpixel correlation.

2.1.2 Rate Distortion function

Shannon's theorem describes a fundamental limit on error free compression. For most natural images, error free compression cannot provide a compression ratio greater

than 2:1. Hence, in practice, we often use lossy cpmpression techniques to get a higher compression ratio. Rate distortion theory establishes theoretical performance bounds for lossy data compression according to a fidelity criterion.

Given a random source vector S, it is of interest to design an encoder that operates at a given rate and which minimizes the average distortion

$$d_{avg} = E\{d(S,\hat{S})\}, \tag{2.12}$$

where, $d(S,\hat{S})$ represents the distortion between S and $\hat{S}$ (which is the reconstruction of S). The mathematical formulation of the above optimization problem is characterized by i) the probability distribution of S and ii) the distortion criterion $d(\mathrm{s},\hat{\mathrm{s}})$. Here, s is the original image, $\hat{\mathrm{s}}$ is the reconstructed image. Let $p(\mathrm{s})$ denote the source probability density, $p(\hat{s}|s)$ be an arbitrary conditional probability (depending on the encoding and decoding process). Then the rate distortion function can be defined as [91] :

$$R(d_{avg}) = \min_{p(\hat{s}|s);E\{d(S,\hat{S})\}\le d_{avg}} I(S,\hat{S}) \tag{2.13}$$

where, $I(S;\hat{S})$ is the average mutual information defined as

$$\begin{aligned} I(S;\hat{S}) &= H(S) - H(S|\hat{S}) \\ &= \sum_{all\ \mathrm{s},\hat{\mathrm{s}}} p(\mathrm{s})\, p(\hat{\mathrm{s}}/\mathrm{s})\, \log_2 \frac{p(\hat{\mathrm{s}}/\mathrm{s})}{\sum_{all\ \mathrm{s},\hat{\mathrm{s}}} p(\mathrm{s}) p(\hat{\mathrm{s}}/\mathrm{s})} \end{aligned} \tag{2.14}$$

where, $H(S)$ is the average entropy of the source S and $H(S|\hat{S})$ is the average entropy of S knowing $\hat{S}$.

From Eq. (2.13) and (2.14), we note that the average mutual information depends on $p(\hat{s}|s)$, and $R(d_{avg})$ is the minimum value of this information over all transition distributions yielding an average distortion d_{avg}. The significance of this function lies in Shannon's coding theorem, which states that no encoder, regardless of the form or complexity, can yeild an average distortion less than d_{avg} at an average transmission

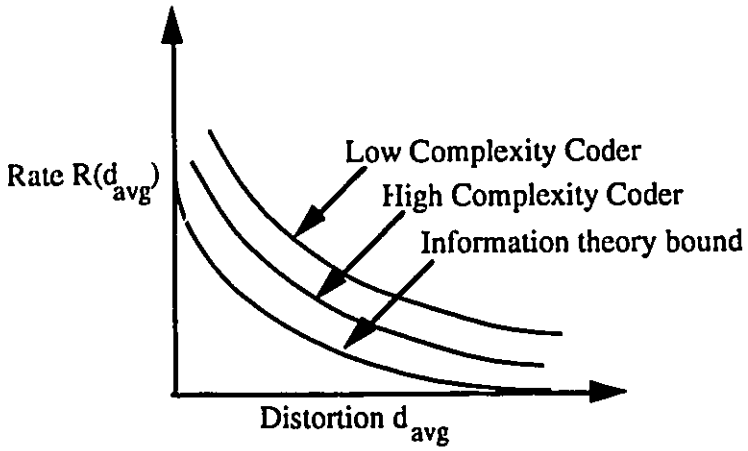


Figure 2.1 : A typical rate-distortion function

rate of $R(d_{avg})$, but it is possible to design an encoder (of sufficient complexity) to yeild an average distortion d_{avg} at a rate arbitrarily close to $R(d_{avg})$. It can be shown that $R(d_{avg})$ is a convex hull, continuous and strictly decreasing function of d_{avg} [79]. A typical rate-distortion curve is shown in Fig. 2.1. This $R(d_{avg})$ is a theoretical limit and can be approached asymptotically by increasing the number of data samples and never achievable in practice. However, it provides a good benchmark against which to compare actual systems.

2.1.3 Distortion Measures

Source coding schemes are generally evaluated by using a distortion measure which is essentially a cost function $d(s,\hat{s})$, for reproducing the input s by an ouput $\hat{s}$. The performance of a coding system can be evaluated using the average distortion introduced by the coding system, $E[d(S,\hat{S})]$,

$$E[d(S,\hat{S})] = \sum_{all\ s,\hat{s}} d(s,\hat{s})\, p(s,\hat{s}) \qquad (2.15)$$

The choice of a distortion measure depends on various factors. First, it should be easily computable. Secondly, it should be analytically tractable. Finally, it should adapt to the human visual system characteristics.

The most widely used distortion measure in image coding is the Peak Signal to Noise Ratio (PSNR), which is defined as [48]:

$$PSNR = 10 \log_{10}\left(\frac{Peak\ Signal\ Value^2}{Mean\ Square\ Error}\right) \tag{2.16}$$

where,

$$Mean\ Square\ Error\ (MSE) = \frac{1}{N \times N}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(s_{ij} - \hat{s}_{ij})^2 \tag{2.17}$$

The other distortion measures used in literature are the Normalized Mean Square Error (NMSE), Signal to Noise Ratio (SNR) and Mean Absolute Error (MAE) which are defined as follows:

$$NMSE = \frac{[\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(s_{ij} - \hat{s}_{ij})^2]}{[\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(s_{ij})^2]} \tag{2.18}$$

$$SNR = 10 \log_{10}\left(\frac{1}{NMSE}\right) \tag{2.19}$$

$$MAE = \frac{1}{N \times N}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(|s_{ij} - \hat{s}_{ij}|) \tag{2.20}$$

where $|s_{ij} - \hat{s}_{ij}|$ is the absolute difference between the original image pixel s_{ij} and the reconstructed image pixel $\hat{s}_{ij}$.

Traditionally, the signal to noise ratio (SNR) has been the most popular error measure in electrical engineering. It provides useful information in most cases and is mathematically tractable. Hence, this is also widely used in image coding. The other measures such as PSNR, NMSE are variants of SNR. It is known that SNR values do not correlate well with the subjective ratings, especially at high compression ratios. Several new distortion measures have been recently proposed for better adaptation to HVS properties [50, 56, 90]. Details are presented in chapter 6.

2.2 Lossless Compression

A high degree of correlation exists between the neighbouring pixels in natural images. In lossless compression techniques, this statistical redundancy is exploited in such a way that the entire process is reversible *i.e.* the original image is exactly recovered.

There is considerable interest in lossless techniques, especially in applications which require very high fidelity reconstructed images (*e.g.* medical imaging). In the following, we present a brief review of the Huffman, run-length and arithmetic coding techniques.

2.2.1 Huffman Coding

According to Shannon's noiseless coding theorem, the average bit rate R for encoding an information source is lower bounded by the entropy of the source. Huffman has provided a practical method for lossless compression [45]. This technique designs a variable length code (VLC) for each source symbol such that the number of bits in the code is approximately inversely proportional to the probability of that symbol's occurence.

To construct a Huffman code, consider a source with an alphabet of size K. By combining the two least probable symbols of the source, we can obtain a new source with (K-1) symbols. This process is repeated until the newest source has only two symbols. These two symbols can be assigned the trivial codewords "0" and "1". Then the codewords for the previous reduced stage can be found by appending a "0" or "1" to the codeword corresponding to the least probable symbols. This process is repeated until the Huffman code for the original source is found.

Huffman code has several limitations. First, it performs optimally only if all the symbol probabilities are integral powers of 1/2 which does not happen in practice. The worst case is realized by a source in which one symbol has a probability approaching unity. Symbols emanating from such a source convey negligible information on average, but require at least one bit to transmit. Secondly, it cannot efficiently adapt to changing source statistics. Though, dynamic Huffman coding schemes have been developed to address these problems, they are very complex to implement.

2.2.2 Arithmetic Coding

Arithmetic coding is an optimal variable length coding scheme. It achieves a compression ratio higher than Huffman coding. This improvement is due to what is known as *fractional bit coding*. For example, if a source has one symbol whose probability approaches unity, *Huffman* coding requires at least one bit to represent each occurrence of the symbol. However, this is not the case with arithmetic coding. It approaches the theoretical entropy bound for any source. In arithmetic coding, there exists a clear separation between the data model and the encoding of data. The performance of the algorithm depends on the effectiveness of the model used.

The basic concept of arithmetic coding was proposed by Elias [2]. Practical techniques for implementing arithmetic coding were first developed by Rissanen [87] and Pasco [76]. An implementation of an arithmetic coder has been provided by Witten *et al.* [109].

To achieve a reasonable coding efficiency with Huffman coding, the message generated by the source is generally divided into blocks and each block is assigned a variable length codeword [79]. In arithmetic coding, a codeword is assigned to the entire input message, say s_m of length m symbols. The codeword length approximately equals $-\log_2 p(s_m)$, where $p(s_m)$ is the probability of the source sequence s_m. Let us assume that the source has an alphabet of size K with a known probability distribution of symbols and also assume a half open interval [0,1). There are K^m different possible messages of length m. Since the sum of $p(s_m)$ over all possible messages is one, it is possible to assign a nonoverlapping sub interval to each s_m in the interval [0,1), where the length of the sub interval is equal to $p(s_m)$. Let us assume that the sub-interval is $[L_l, R_l)$, $(l = 1,2,.....K^m)$. Once, the subinterval is determined, any real number in that sub-interval will uniquely decode the entire input message. Since, the sub-intervals are nonoverlapping, a codeword for s_m can be constructed by expanding any point in the interval in binary form and retaining only the $n_l = \lceil -\log_2 p(s_m) \rceil$ bits

after the decimal point. Hence, the number of bits required to represent the message can differ from the source entropy by a maximum of one bit.

The major problem associated with arithmetic coding is the precision required to carry out the subinterval computation. As the length of the source sequence increases, the width of the subinterval decreases. For example, if a sequence requires 50,000 bits (which is usually the case in image coding), the associated sub-interval will be $2^{-50,000}$. This problem can be resolved using a scaling and rounding strategy [109].

The arithmetic coder is very simple to implement. To start with (before anything is transmitted), the range is the entire interval [0,1). As each symbol is encoded, the range becomes narrower. For example, Table-2.1 shows an alphabet of some source and also the probabilities of the symbol (here a fixed model has been assumed) which will be used to encode the symbols generated by the source.

Table-2.1
Example of a fixed model for
alphabet {*a, b, c, d, e, !*}

Symbol	Probability	Range
a	0.2	[0, 0.2)
b	0.1	[0.2, 0.3)
c	0.2	[0.3, 0.5)
d	0.1	[0.5,0.6)
e	0.3	[0.6,0.9)
!	0.1	[0.9,1.0)

Suppose, we want to transmit the symbol *baad!*. Initially, the range is [0, 1). After seeing the first symbol *b*, the encoder narrows the range to [0.2, 0.3). As soon as it is determined that the interval is [0.0,0.5), the encoder knows that the first symbol is "0". Hence, it can start sending the output bits. After the second symbol, the encoder narrows the range further to [0.2, 0.22). At this moment the encoder knows that the second symbol is "0", since the interval falls within [0,0.25). Similarly, the third symbol will be "1" as the interval falls within [0.125,0.25). A blow up of the process is shown in Fig. 2.2. The encoding process can be illustrated as follows :

		Sub-interval		Bitstream generated
Starting		[0,	1.0)	
after seeing	*b*	[0.2,	0.3)	0
	a	[0.2,	0.22)	0
	a	[0.2,	0.222)	1
	d	[0.211,	0.2132)	1
	!	[0.21298,	0.2132)	0

After the encoding is completed, the transmitter needs to send any number in the interval [0.21298, 0.2132). However, the decoder will face the problem of detecting the end of the message to stop decoding. For example, the single number 0.2 could represent *ba, baa, baaa,* .... To resolve this ambiguity, normally a special EOF (end of file) symbol is used. This EOF marker is known to both encoder and decoder and thus the ambiguity can be resolved. In this case, "!" can be used as the EOF marker.

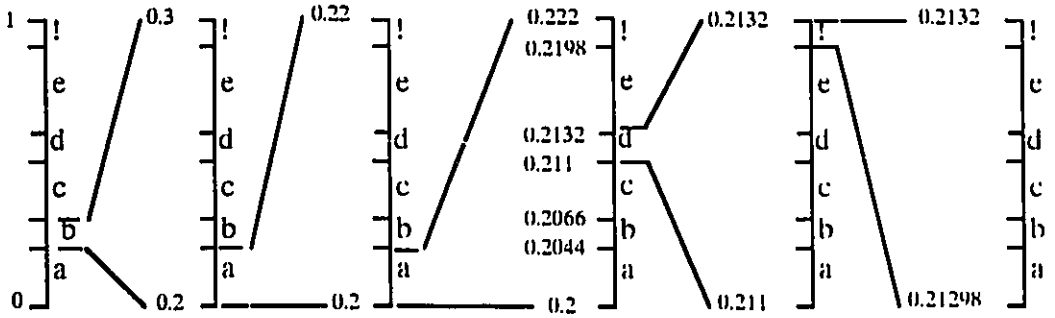


Figure 2.2 : Arithmetic encoding process

The decoding scheme is very simple. Suppose, the number 0.213 is sent. The decoder knows that the first symbol is *b,* as the number lies in [0.2,0.3). The second symbol is *a,* since the number lies in the first one-fifth of the range, *i.e.*, within [0.2, 0.22). Thus, the process continues.

In the above example, a fixed model has been used, which is not suitable for coding nonstationary sources. For wavelet-based coding, the various bands have widely different statistics. In addition, the statistics also changes within a band. Hence, an

adaptive model must be implemented to achieve a good compression ratio. Table 2.2 shows the performance of an adaptive arithmetic coder for a source with an alphabet of {-1, 0, 1}. The frequencies of occurence of the symbols and the associated probabilities are provided by the entries in column 2 and 3, respectively. The quantized coefficients of higher passbands of a wavelet coder represent a source with similar statistics. It is seen that the compression ratio achieved by an adaptive arithmetic coder is very close to the actual entropy.

Table - 2.2
A typical source alphabet (total no. of samples=65536)

Symbol	No. of occurence	prob., p	Average. Entropy	Achieved Bit-rate
- 1	100	0.0015258	0.031 bpp	0.031 bpp
0	65336	0.9969482		
1	100	0.0015258		

2.3 Run-length Coding

Run-length coding (RLC) is also a lossless coding technique and provides a good coding performance for images having identical consecutive symbols. It was developed in the 1950's and has become, along with its two dimensional extensions, the standard approach for facsimile (FAX) coding. RLC defines a *run* as a sequence of consecutive pixels with identical values (usually zeroes or ones) along a fixed direction, for example the horizontal/vertical scan lines. Significant bit rate reductions can be achieved for long *runs*, especially in binary images like text, graphics, *etc.* Although, run-length coding is in itself an effective method of compressing an image, additional compression can usually be achieved by variable length coding the run lengths themselves. The black and white run lengths may be coded separately using variable length codes that are specifically tailored to their own statistics to achieve better compression ratio [34].

RLC cannot be applied to images with high details as the efficiency will be very low. However, significant compression may be achieved by first splitting the images into a set of bit planes which are then individually run-length coded. RLC is also used to provide high compression in transform coders. Most of the high frequency coefficients in a transform coder become zero after quantization and long runs of zeroes will be produced. Run-length coding can then be used very efficiently along with a VLC [104]. Run-length coding is usually extended to two dimensions by defining a *connected area* to be a contiguous group of pixels having identical values. To compress an image using two dimensional RLC, only the values which specify the *connected area* and its intensity are stored/transmitted.

2.3 Lossy Compression

The lossless compression techniques usually result in a low compression ratio (typically 2 to 3). Hence, they are not employed when a high compression ratio is required. In lossy compression, the objective is to reduce the bit rate subject to some constraints on the image quality. Most lossy compression techniques can be classified into following categories - predictive coding, transform coding, wavelet/subband, vector quantization, fractal coding. In the following, we briefly discuss each of these coding techniques.

2.3.1. Predictive Coding

Predictive coding exploits the redundancy related to the predictability and smoothness in the data. For example, an image having a constant gray level can be fully predicted from the gray level value of its first pixel. In images with multiple gray levels, the gray level of an image pixel can be predicted from the values of its neighboring pixels. *Differential pulse code modulation (DPCM)* is the basic compression scheme used in predictive coding techniques.

DPCM

The block diagram of a DPCM system is shown in Fig. 2.3. The difference between the original pixel s_n and its predicted value $\hat{s}_n$, called e_n, is quantized and entropy coded. The quantized difference value is in turn used to predict the next data sample s_{n+1}. In general, both linear and non-linear techniques can be used for prediction. In linear predictive coding (LPC) technique, the predicted value $\hat{s}_n$ is calculated as follows [92]:

$$\hat{s}_n = \sum_{i=0}^{M-1} \alpha_i s_{n-i} \tag{2.21}$$

where, α_i , $i = 0, 1, 2, ..., M-1$, are the coefficients of an M-th order predictor. The predictor can be optimized to minimize the variance or the energy of the prediction error e_n. The optimal set of LPC coefficients $\alpha_{i,opt}$ can be obtained using the following set of simultaneous equations :

$$\sum_i \alpha_{i,opt} R(j-i) = R(j) \tag{2.22}$$

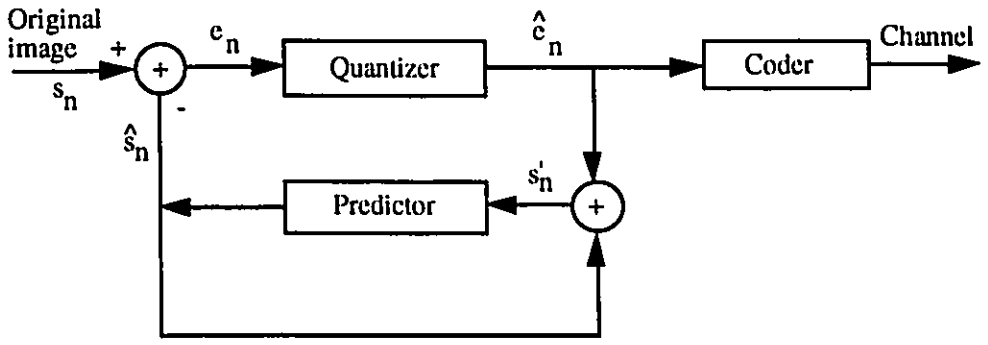
where $R(j)$, $j = 0, \pm 1, \pm 2, ...$ is the autocorrelation function of the input data samples. The step size for quantizing the prediction difference e_n can be adjusted depending on the available data rate and the MSE requirements. The following are some examples of typical predictors [79] :

$$\hat{s}_n = 0.97 s_{n-1} \qquad \text{1st order, 1-D predictor}$$

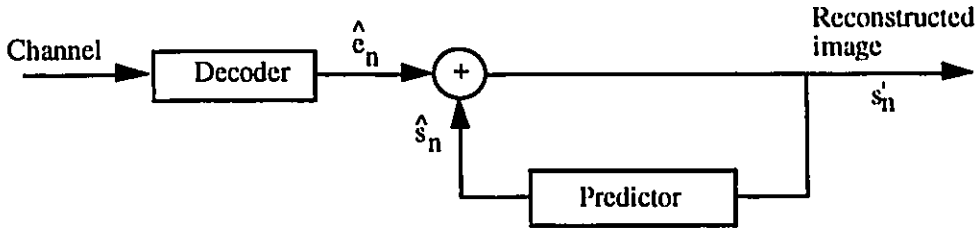
$$\hat{s}_{m,n} = 0.5 s_{m,n-1} + 0.5 s_{m-1,n} \qquad \text{2nd order, 2-D predictor}$$

$$\hat{s}_{m,n} = 0.9 s_{m,n-1} - 0.81 s_{m-1,n-1} + 0.9 s_{m-1,n} \qquad \text{3rd order, 2-D predictor}$$

A special case of DPCM is *Delta Modulation* where the quantizer output can assume only two possible levels. Delta Modulation quantizes the positive and negative differences as $+\Delta$ and $-\Delta$, respectively, where Δ is a predetermined ouput value.



a) Transmitter



b) Receiver

Figure 2.3: Block diagram of a DPCM system
a) Transmitter, b) Receiver

2.3.2. Transform Coding

In transform coding, the image data is first transformed from spatial to frequency domain by a unitary transform and then coded by standard scalar or vector quantization techniques. A unitary transform is a reversible linear transform whose kernel describe a set of complete orthonormal basis functions. The objective of transform coding is to decorrelate the original signal and to repack the energy into fewer coefficients.

Let an $N \times N$ image be denoted by

$$S = [s(m,n)] \qquad \text{for } 0 \le m,n \le N-1$$

The forward transform is defined as

$$\theta(i,j) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} \alpha(i,j;m,n)s(m,n) \qquad 0 \le i,j \le N-1 \qquad (2.23)$$

and the inverse is

$$s(m,n) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} \beta(m,n;i,j)\theta(i,j) \qquad 0 \le m,n \le N-1 \qquad (2.24)$$

where $\alpha(.)$ and $\beta(.)$ are the forward and inverse transform kernels.

In most practical cases, the 2-D kernels are separable and symmetric so that the 2-D kernel can be expressed as the product of the two 1-D orthogonal basis functions. If the 1-D transform operator is denoted by Φ, the forward and inverse transformations can be expressed as

$$\Theta = \Phi^* S \Phi^T$$
$$S = \Phi^T \Theta \Phi^* \qquad (2.25)$$

The above formulations reveal that the image transformation can be done in two stages : by taking the unitary transform Φ of each row of the image array and then by applying transformation Φ^* to each column of the intermediate result. A typical transform coding scheme is shown in Fig. 2.4.

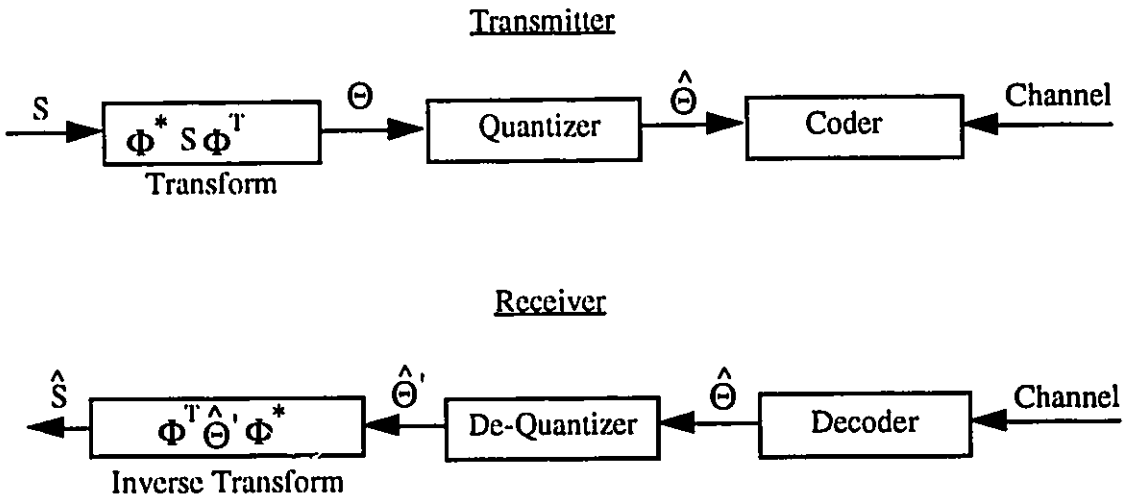


Figure 2.4: Block diagram of a transform coding scheme,
S= Original Image, $\hat{S}$ = Reconstructed image

Optimum Transform

The transformation of an image results in a set of spectral coefficients which are nonstationary in nature. The transform will be statistically *optimal* for image coding if it satisfies two criteria : First, there should not be any correlation among the coefficients *i.e.*, the autocorrelation matrix should be diagonal. Secondly, it should repack the energy in as fewer coefficients as possible. If we calculate the energy of the first L transform coefficients ($0 \leq L \leq N-1$) resulting from various transforms, the energy provided by the optimum transform should be maximum.

The unitary transform that satisfies both the criteria is Karhunen-Loeve transform (KLT) [4]. However, KLT is image dependent and has higher computational complexity. Therefore, in practice, image independent sub-optimal transforms such as discrete sine, cosine, Fourier, Hadamard, Walsh transforms are used which have a lower computational complexity compared to KLT. Among all the sub-optimal transforms, the rate distortion performance of discrete cosine transform (DCT) is closest to KLT [81]. Hence, DCT has been adopted as the compression technique in image and video coding standards, such as JPEG, MPEG and H.261. It can be shown that for a first order Markov source model, the DCT basis functions become identical to the KLT basis functions as the adjacent pixel correlation coefficient approaches unity [16]. Natural images generally exhibit high pixel-to-pixel correlation and therefore DCT provides a compression performance virtually indistinguishable from KLT. In addition, DCT has a fast implementation like DFT, with a computational complexity of $O(N \log N)$ for an N-point transform. Unlike DFT, DCT avoids generation of spurious spectral components resulting in a higher compression efficiency.

Orthonormal transforms have the following two properties which are very useful in image coding applications [49] :

1. It satisfies Parsevals relation *i.e.* the total energy in the frequency domain is equal to that of the spatial domain.

$$\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}|s(m,n)|^2 = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}|\theta(i,j)|^2 \tag{2.26}$$

2. The mean square reconstruction error is equal to the mean square quantization error *i.e.*

$$\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}|s(m,n)-\hat{s}(m,n)|^2 = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|\theta(i,j)-\hat{\theta}(i,j)\right|^2 \tag{2.27}$$

These two properties are very helpful in designing an MSE quantizer. If we have a distortion budget, we know that the same amount of distortion can be introduced in the frequency domain. The noise power in the frequency domain is additive. Hence, we can distribute the noise among the coefficients according to some optimum criteria.

Bit Allocation

In order to obtain significant compression, the bit allocation procedure in a transform coding technique generally requires optimization. One of the popular criteria for optimization is to minimize the mean square error. In other words, the transform coefficients should be assigned bits depending on their contribution to error variance in the spatial domain. If the signal has $N \times N$ coefficients and the total number of available bits is R, then the optimization problem is to find various $R_{i,j}$, so that

$$D = \sum_{i,j=0}^{N-1} D_{i,j} \text{ is minimized with the constraint}$$

$$R = \sum_{i,j=0}^{N-1} R_{i,j} \tag{2.28}$$

where $D_{i,j}$ is the distortion produced by the (i,j)*th* coefficient when $R_{i,j}$ number of bits are assigned to it. If we use the *pdf* optimized Lloyd-Max quantizer, the total distortion becomes [49],

$$D = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \gamma_{i,j} 2^{-2R_{ij}} \sigma_{i,j}^2 \tag{2.29}$$

where, $\sigma_{i,j}^2$ is the variance of the (i,j)-th transform coefficient and $\gamma_{i,j}$ is a performance factor which depends on the *pdf* of the coefficient.

For orthonormal transform, the above problem can be solved using *Lagrangian optimization* [4]. The solution becomes,

$$R_{i,j} = R_{avg} + \frac{1}{2}\log_2 \frac{\sigma_{i,j}^2}{[\prod_{k,l=0}^{N-1} \sigma_{k,l}^2]^{1/(N\times N)}}; \quad i,j = 0,1,\ldots,N-1 \tag{2.30}$$

where, R_{avg} is the average number of bits per coefficient, *i.e.*, $R_{avg} = R/(N\times N)$. It can also be shown that, with this optimal bit allocation, the quantization noise of all coefficients are equal [49].

Transform Coding Gain

A good measure for comparing coding performance of various transforms is the transform coding gain [49]. For the orthogonal transform, this is defined as

$$G_{TC} = \frac{\frac{1}{N\times N}\sum_{i,j=0}^{N-1} \sigma_{i,j}^2}{\left(\prod_{i,j=0}^{N-1} \sigma_{i,j}^2\right)^{1/(N\times N)}} \tag{2.31}$$

where, N is the total number of coefficients to be coded and $\sigma_{i,j}$ is the variance of the (i,j)*th* coefficients.

Fourier-based transforms (*e.g.* DCT, DST, DFT) are efficient in exploiting the low frequency aspects of the image, but clearly do so at the expense of the edge information, which is spread out across the frequency spectrum. A sharp edge is represented by many cancellation terms which must be preserved intact and in the same relationship to one another to achieve good fidelity of the reconstructed image. An uncompensated quantization error in one band will be present throughout the spatial domain in which it has been constrained to operate [113].

Typical implementations of spectral techniques involves first dividing the original image into 8x8 or 16x16 subblocks. This serves two purposes - i) it decreases the computational complexity, ii) it limits the damage done by run-away cancellation terms. This is the drawback of large or infinite basis functions. Blocking of data introduces a number of undesirable side effects. Most objectionable is the blocking effect which shows a quilted appearence of the image. In addition, Gibbs phenomenon of the spectral method causes a loss of contrast in the image when high frequency coefficients have quantization errors. The blocking method also imposes an upper limit to the actual compression ratio achieved, since one must save a high resolution DC term and the coefficients of at least the lowest frequencies for each block.

2.3.3 Subband Coding

Subband coding was first developed for speech compression and later extended for image coding [99, 110]. In subband coding, an image is first filtered to create a set of sub-images or *subbands*, each of which contains a limited range of spatial frequencies. The different subbands can be downsampled due to their lower bandwidths as compared to the original image, keeping the data rate same. The subbands are then quantized and encoded using one or more coders. Different bit-rates or coding techniques can be used for each subband, thus taking advantage of the properties of the subbands. The coding errors can be distributed over the subbands in a visually optimal manner. The image is reconstructed by upsampling the decoded subbands, applying appropriate filters and adding the reconstructed subbands together. Subband coding is generally implemented using quadrature mirror filters (QMFs) for reducing aliasing effect in the reconstructed image.

Subband coding coding is motivated by the idea that subbands can be coded more efficiently than the entire full band image. This is because most of the energy in the subband domain is represented by few lowpass coefficients. The idea is very similar to

that of transform coding. In fact, transform coding and subband coding are two special cases of multirate filterbanks. In practice, DCT, DFT *etc.*, are used in a block coding approach with a blocksize of 8×8 or 16×16. This can be viewed as a filterbank with the decimation factor being the same as the filter length. In subband coding, the subband filter length is generally much larger than the decimation factor, resulting in lesser blocking artifacts in the reconstructed image [79].

Wavelets

Recently, wavelets have become very popular in image compression. It provides a multiresolution / multi-frequency representation of nonstationary signals, such as images. It has good energy compaction capability resulting in high compression ratio. Wavelets can also be viewed as a special case of multirate filterbanks with a dyadic tree decomposition. It provides good signal localization in both spatial and frequency domains. A detailed review of wavelets and its application in image coding is provided in later chapters.

2.3.4 Vector Quantization

A fundamental result of Shannon's rate-distortion theory is that better performance can always be achieved by coding vectors instead of scalars. A vector quantizer (VQ) can be defined as a mapping Q of K-dimensional Euclidean space R^K into a finite subset Y of R^K [57], *i.e.*,

$$Q: R^K \rightarrow Y$$

where $Y = (x_i'; i = 1,2,.....N)$

Y is the set of reproduction vectors, and is called a VQ codebook or VQ table. N is the number of vectors in Y. At the transmitter and receiver, an identical codebook exists whose entries contain combinations of pixels in a block. At the encoder, each data vector is matched or approximated with a codeword in the codebook, and the address or index of that codeword is transmitted instead of the data vector itself. At the

decoder, the index is mapped back to the codeword, and the codeword is used to represent the original data vector. If the image block size is $m \times n$ pixels, and each pixel is represented by p bits; theoretically $(2^p)^{mn}$ combinations are possible. In practice, however, there are only a limited number of combinations that occur most often, which reduces the size of the codebook considerably. This is the basis for vector quantization.

The major drawback of vector quantization is that it is highly image dependent and its computational complexity grows exponentially with vector dimension. Also, there is a problem in designing a good codebook that is representative of all the possible occurrences of pixel combinations in a block.

2.3.5 Fractal Image Compression

In simplest terms, a fractal is an image of a texture or shape, expressed as one or more mathematical formulas. In terms of fractal geometry, a fractal is a geometric form whose irregular details recur at different scales and angles which can be described by fractal transformations (*e.g.* an affine transformation). Fractal transformations can be used to describe most real world pictures.

Fractal image compression is the inverse of fractal image generation, *i.e.* instead of generating an image from a given formula, fractal image compression searches for sets of fractals in a digitized image which describe and represent the entire image. Once the appropriate sets of fractals are determined, they are reduced to very compact fractal transform codes or formulas. The codes are rules for reproducing the various sets of fractals which, in turn, regenerate the entire image. Since fractal transform codes require a very small amount of data to be expressed and stored as formulas, fractal compression results in very high compression ratios [103].

The major drawback of fractal coding is that it is image dependent. For each image, a distinct set of rules must be specified. Obviously, it is easier to specify a set of images

showing a repeated pattern, than it is for a picture with a number of distinct features. In addition to being image specific, fractal coding is also a computationally intensive technique. However, the computations required are iterative and make possible high speed hardware implementations. Also, fractal coding is highly asymmetric as the computational complexity of the decoder is much less than that of the encoder.

2.3.6 Future Trends

A number of high performance coding techniques has been discussed. Usually a combination of these techniques is used in practice to achieve a high compression ratio. For example, vector quantization of wavelet transform coefficients provides a very good coding performance [7, 105]. The choice of a compression technique depends on its computational complexity and coding performance.

Predictive coding techniques have lower computational complexity and memory requirements. However, their coding performance is relatively poor resulting in a loss of subjective image quality at high compression ratios. Transform coding usually achieves compression ratios higher than the predictive coding for a given distortion value. It also achieves a better subjective image quality by distributing the reconstruction error over a large image region. However, it has higher computational complexity and memory requirements. Wavelet/subband coding uses a filterbank structure and achieves a good coding performance. The subjective quality of the reconstructed images is better than that achieved using block transform coding. Vector quantization and fractal coding achieves high compression, but their computational complexity is very high.

The present and future trends in the performance of image coding techiques for still imagery and video are depicted in Fig. 2.5 [39, 48]. The developments are summarized in terms of number of bits required to have a still frame picture or video with little or no noticeable degradation. Although, at very low bit-rate, the reconstructed image may

not acceptable for home or studio quality television, they may be acceptable for teleconferencing and in image recognition and detection applications where the emphasis is on information content rather than cosmetic quality.

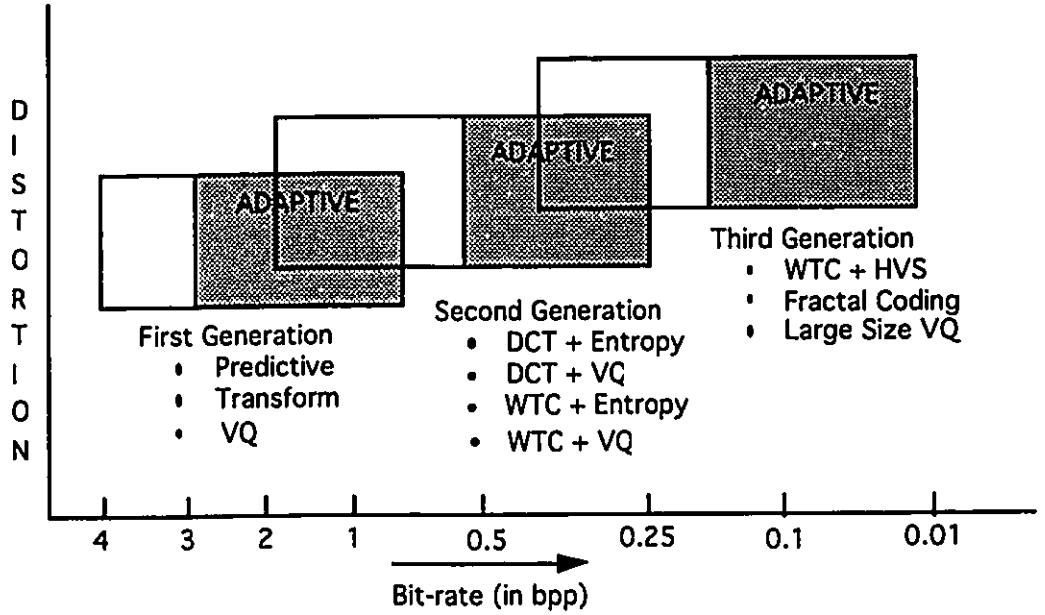


Figure 2.5 : Comparison of Image Compression Techniques

In Fig. 2.5, the first generation techniques include predictive coding, transform coding and simple vector quantization techniques. These techniques were developed in the 60's and 70's. Second generation techniques use DCT-based coding (with entropy coding or VQ). It also includes wavelet (also subband) coding. The present industry standards *e.g.* JPEG fall in this cateory. The third generation techniques refer to the techniques which are under development *e.g.* wavelet-based coding adapted to human visual system properties, fractal coding, large size VQ, *etc.*

2.4 JPEG Standard for Still Image Compression

The CCITT and ISO collaborated to develop the most popular and comprehensive JPEG (Joint Photographic Experts Group) standard for still image compression. The JPEG standard provides a framework for high quality compression and reconstruction

of continuous-tone grayscale and color images for a wide range of applications[51, 104]. The standard specifies details of the compression and decompression algorithms for various application environments. It has four modes of operation, which usually covers most image compression environments. These are [104] :

(a) Baseline Sequential

(b) Progressive coding

(c) Hierarchical coding

(d) Lossless Compression

The baseline sequential mode provides a simple and efficient algorithm that is adequate for most image coding applications. In progressive coding, the image is encoded in multiple scans for applications in which transmission time is long and the user prefers to view the image building up in multiple coarse-to-clear passes. In heirarchical coding, the image is coded at multiple resolutions, so that lower resolutions versions may be accessed without first having to decompress the image at its full resolution. In lossless coding, the image is encoded to guarantee the exact recovery of every source image sample value (of course, if the channel is error free!). The progressive and hierarchical modes usually use a modified version of the baseline algorithm. We now present a brief description of the baseline sequential mode.

Baseline Sequential Mode

In the baseline system, the input and output data precision is limited to 8 bits, whereas the quantized DCT values are restricted to 11 bits. The compression is performed in three steps : DCT computation, quantization and variable-length coding. The original image is first partitioned into non-overlapping blocks of 8x8 pixels as shown in Fig.2.6. In order to decrease the average energy of the image pixels, each pixel is level-shifted by subtracting the quantity 2^{n-1} , where n is the number of bits required to represent each pixel value (*i.e.* a value of 128 is subtracted for images with

256 gray level). The 2-D DCT of the block is then computed and quantized using a visually adapted quantization table suggested by JPEG. Each 8x8 block generates one DC coefficient and 63 AC or, high frequency coefficients. The quantized coefficients are reordered using zigzag pattern to form a 1-D sequence of quantized coefficients. This scheme generates long runs of zero value coefficients (corresponding to the higher frequency AC cofficients) in most of the images. The DC coefficients from each block is DPCM coded and all other coefficients, *i.e.*, the AC coefficients are compressed using a combination of Huffman and run-length coding. The decoding scheme is just the inverse of the encoding scheme and has not been shown here.

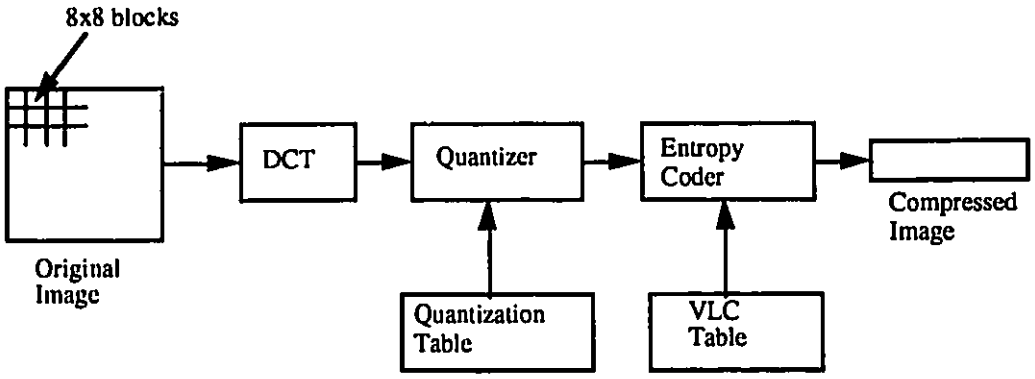


Figure 2.6 : Baseline JPEG Encoder

The computational complexity of JPEG Baseline algorithm can be calculated as follows. The computation of two-dimensional DCT of 8x8 blocks is usually done by a separable approach. Several well known algorithms require 29 additions and 13 multiplications to perform an 8-point DCT [79]. A total of 16 1-D DCT will be required to process each 8x8 block. It has been shown that a JPEG encoder requires about 12 FLOP per input pixel, ignoring the complexity of Huffman encoding [79]. For a 512x512 image, the computational complexity is about 3 MFLOP. The computational complexity of the decoder is similar to that of encoder.

2.5 Summary

In this chapter, we first reviewed the fundamental concepts of source coding theory, namely *entropy* and *rate-distortion*, as applied to image compression. This was followed by a review of lossless and lossy compression techniques, such as Huffman, arithmetic, run-length, transform and subband coding. The concept of vector quantization and fractal coding was also introduced briefly. This was followed by a discussion of the future trends in image compression. We then presented a brief review of the JPEG standard for still image compression.

Chapter 3

Wavelet Theory and Implementation

In recent years, wavelet transform has emerged as a powerful mathematical tool in many areas of science and engineering [15, 37, 68, 84, 102, 108], specifically for data compression [17, 27, 55]. In this chapter, we will briefly describe the theory and implementation of wavelet transform. First, we discuss the continuous wavelet transform and its usefulness in nonstationary signal analysis [20, 44, 88]. We discuss the multiresolution capability of wavelets. We then derive the recursion formulas for calculating discrete wavelet transform (DWT). This is followed by a discussion about the implementation of DWT for finite length data. The parameterization of wavelets is considered next. We also review two properties of wavelets namely, regularity and vanishing moments which are very important for image coding. We next provide the design procedure of two wavelet families : *Daubechies* and *Coiflet* families. The extension of 1-D wavelet transform to 2-D case is presented next. This is followed by the comparison of the computational complexity of different algorithms for calculating DWT. Finally, we conclude the chapter by summarizing the reviews.

3.1 Continuous Wavelet Transform :

The objective of signal analysis is to extract relevant information from the input signal. A generic way of signal analysis is to represent the signal by a set of basis functions

whose properties are well understood. For stationary signal analysis, the best known transform is the Fourier transform which is defined as [29],

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt \tag{3.1}$$

Fourier transform has been successful in wide range of applications; however, it works well only if the signal consists of a few stationary components. Any abrupt change in time is reflected in the whole spectrum. Thus, if there is a sharp discontinuity, it requires many transform coefficients to represent the signal discontinuity. Also, the amplitude spectrum doesn't provide any idea of how the frequency components evolve with time. To determine the time evolution, one has to analyze the phase spectrum as well, which is a difficult task.

This drawback can be overcome by applying a time window on the data and then taking the Fourier transform. This is known as short-time Fourier transform (STFT). STFT of $x(t)$ can be calculated as [73]

$$STFT_x(\omega,b) = \int_{-\infty}^{\infty} x(t)g^*(t-b)e^{-j\omega t}dt \tag{3.2}$$

The window $g(t-b)$ is a time-localized function which is shifted over the time axis to calculate the transform at several time locations b. The STFT can be viewed as a transform with a set of basis functions that are windowed complex exponentials. Uncertainty principle dictates that the joint time-frequency localization cannot be improved beyond a certain limit. That means, one can't have good time and frequency resolution simultaneously. One of the commonly used compromizes is the Gaussian time window with a corresponding Gaussian frequency window. This transform is called Gabor transform [30]. It should be noted that Gaussian functions have the best time-frequency localization.

Ths STFT has two major drawbacks. First, both the time and frequency windows are fixed. Hence, STFT cannot adapt to the signal statistics. The need to adapt to the

signal arises because very low and very high frequency components may be simultaneously present in the signal. A short time window will be able to detect the high frequency components, but will provide poor response to low frequency components and vice versa. Thus, we need a system which can tackle wide range of input signals simultaneously. Secondly, it is desirable to have a window function that has some decay in the time domain as well as in the frequency domain. According to Ballian-Low theorem, the short-time Fourier transform cannot simultaneously have an orthonormal basis and a window function with good localization both in time and frequency domain [24]. Continuous wavelet transform (CWT) overcomes both of these drawbacks. CWT is defined as

$$CWT_x(a,b) = \int_{-\infty}^{\infty} x(t)\psi^*_{a,b}(t)\,dt \tag{3.3}$$

where the basis functions $\psi_{a,b}(t) \in L^2(R)$, where $L^2(R)$ is the space of square summable functions[14], can be expressed as

$$\psi_{a,b}(t) = a^{-1/2}\psi(\frac{t-b}{a}), \qquad a \in R^+, b \in R \tag{3.4}$$

These basis functions are called *wavelets* (*i.e.* small waves) and can be viewed as dilated and translated versions of the mother wavelet $\psi(t)$. These functions can have both compact or infinite support in time. The arguments a and b denote *scale* and *location* parameters respectively. Smaller a represents rapidly oscillating basis functions. The factor $a^{-1/2}$ on the right hand side of Eq. (3.3) maintains the norm of the wavelet in different scales.

The tiling of the time-frequency plane for both STFT and CWT is shown in Fig. 3.1. While the time and frequency windows are fixed for STFT, the window shape changes for CWT. For higher scale, the time window narrows and for lower scale, it widens. This type of adaptation is very useful in most practical situations. The high frequency components are normally short-lived and hence need a narrow time window to

improve the time resolution. On the other hand, low frequency components change slowly over time, thus requiring a wider time window to detect the slow changes.

The signal can be reconstructed by the following *resolution of identity* formula [14].

$$x(t) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{0}^{\infty} CWT_x(a,b) a^{-1/2} \psi\left(\frac{t-b}{a}\right) \frac{da\, db}{a^2} \tag{3.5}$$

The constant C_ψ depends only on $\psi(t)$ and is given by

$$C_\psi = \int_{0}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} < \infty \tag{3.6}$$

where $\Psi(\omega)$is the Fourier transform of $\psi(t)$. The constant C_ψ must not be equal to zero or ∞, otherwise Eq. (3.5) can not be evaluated.

Eq. (3.5) can be viewed as describing x(t) as a summation of basis functions with weights $CWT_x(a,b)$.

It is easy to see from Eq. (3.6) that for $C_\psi$to be finite, $\Psi(\omega)$ must be zero at $\omega = 0$.

This means

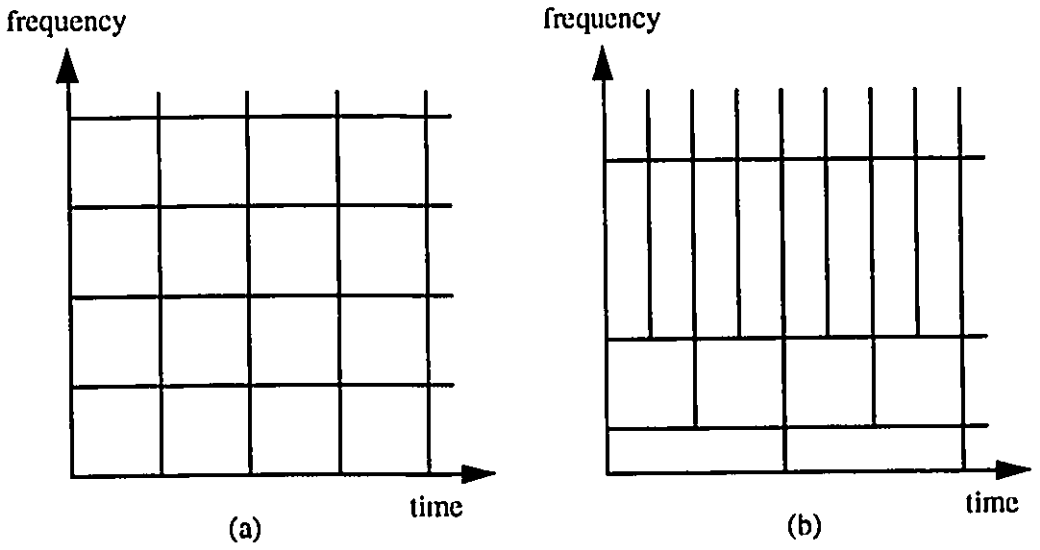


Figure 3.1 : Time-frequency plane showing resolution cells
a) STFT, b) CWT

$$\int_{-\infty}^{\infty} \psi(t)dt = 0 \tag{3.7}$$

This is partially the reason for the name wavelet : small oscillating functions with zero mean. The parameters a and b are free to vary continuously in R (the space of real numbers) with the constraint $a \neq 0$. The wavelet transform defined as above represents a function of one variable using functions of two continuous variables (namely a and b). Thus, it is natural that the transform is highly redundant. Though in some cases redundancy may be useful, for the image compression application, it is not desirable. In addition, when the transform is redundant, there is no unique way for reconstructing the original signal. It is possible to reduce the redundancy by sampling the parameters a and b on a rectangular grid in the following way :

$$a = a_0^j, \quad b = k\, b_0\, a_0^j, \qquad\qquad j,k \in Z,\ a_0 > 1,\ b_0 \neq 0 \tag{3.8}$$

where Z is the space of integer numbers. b_0 can be chosen as any real number but for convenience it is generally chosen as b_0=1. Thus, Eq. (3.4) can be written as

$$\psi_{j,k}(t) = a_0^{-j/2}\psi(a_0^{-j}t - k) \tag{3.9}$$

Thus, the discretized version of Eq. (3.3) becomes

$$CWT_x(j,k) = \int_{-\infty}^{\infty} a_0^{-j/2}\psi(a_0^{-j}t - k)x(t)dt \tag{3.10}$$

Though, a and b are discrete, the transform is still for continuous signals. By properly choosing a and b, we can have an orthonormal set of basis functions (*i.e.* a tight and exact frame). In many applications of signal analysis, it is required to partition the frequency axis into disjoint frequency bands possibly with some overlap. The simplest case is the binary partition, in which case a_0 should be equal to 2. The wavelet with a_0=2, is known as *dyadic wavelet.*

$\psi_{j,k}$ forms an orthonormal set of basis functions, if it satisfies the following orthonormality condition

$$\int \psi_{j,k}(t)\psi^*_{l,m}(t)dt = \delta_{j-l,k-m} \tag{3.11}$$

where $\delta_{i,j}$ is the Kronecker delta function. In this case, the transform satisfies Parseval's theorem (*i.e.* conservation of power or energy) along with other interesting properties of an orthonormal transform.

3.2 Multiresolution Representation of Functions

The power of the wavelet theory lies in its ability to represent a signal at different resolutions with excellent time-scale localization [4, 23, 61]. The *scale* in wavelet literature, is analogous to the frequency in Fourier domain. The only difference is that in wavelet theory the width of the frequency bands vary logarithmically. Multiresolution analysis enables one to view a signal or image in different resolutions and has interesting applications in image analysis, pattern recognition *etc.*

A multiresolution analysis is possible if we can find a sequence of closed subspaces $\{V_j | j \in Z\}$ of $L^2(R)$, which has the following properties [4].

- Containment

$$\ldots V_2 \subset V_1 \subset V_0 \subset V_{-1} \ldots. \tag{3.12}$$

<--- Coarser Finer--->

- Completeness

$$\bigcap_{j \in Z} V_j = \{0\} \text{ and } \bigcup_{j \in Z} V_j = L^2(R) \tag{3.13}$$

- Scaling property

$$x(t) \in V_j \Leftrightarrow x(2t) \in V_{j-1} \tag{3.14}$$

for any function $x \in L^2(R)$.

- Orthogonality

There exists a scaling function such that the set

$$\{\phi_{j,k}(t) = 2^{-j/2}\phi(2^{-j}t - k)\} \tag{3.15}$$

is an orthonormal basis for V_j, *i.e.*,

$$\int \phi_{j,k}(t)\phi^*_{j,k'}(t)dt = \delta_{k-k'} \tag{3.16}$$

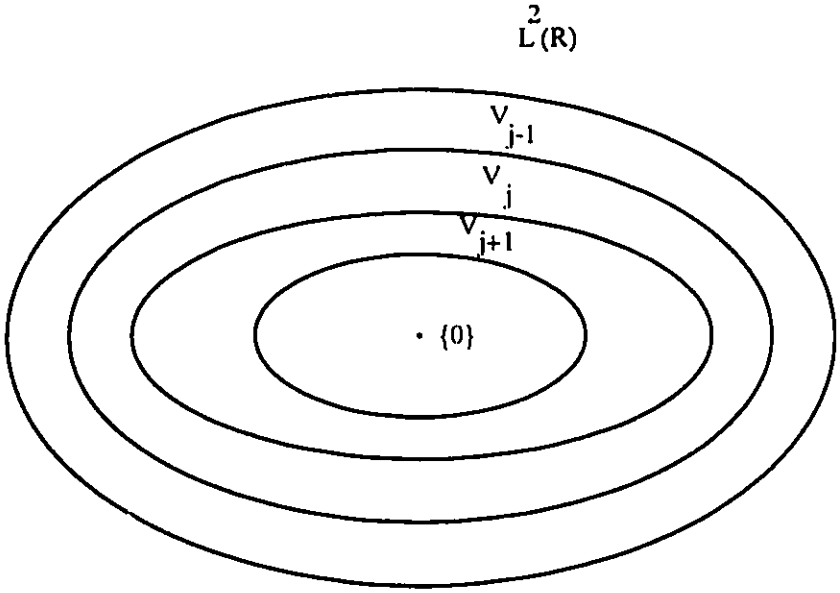


Figure 3.2 : Multiresolution representation of $L^2(R)$

The functions $\{\phi_{j,k}\}$ are known as *scaling* functions since they build up scaled versions of functions in $L^2(R)$. Any function $x(t)$ can be expressed as the limit of the approximation $x_j(t) \in V_j$ for j tends to $-\infty$, *i.e.*

$$x(t) = \lim_{j \to -\infty} x_j(t) \tag{3.17}$$

Multiresolution analysis enables one to examine the function at several resolutions or scales. The embedded space of V_j is shown in Fig. 3.2 [10]. The variable j indicates the *scale* and is called the *scale factor*. If the *scale factor* is high, the *scale* is large and

the function in V_j will be a coarse approximation of the original, *i.e.*, the details will be neglected. The function $x_j(t)$ can also be thought of as an orthogonal projection of x(t) onto V_j,

$$x_j(t) = \sum_k \langle \phi_{j,k}(t), x(t) \rangle \phi_{j,k}(t)$$
$$= \sum_k c_{j,k} \phi_{j,k}(t) \tag{3.18}$$

Since, $\phi(t) = \phi_{0,0}(t) \in V_0 \subset V_{-1}$, we can write,

$$\phi_{0,0}(t) = \sum_n h[n] \phi_{-1,n}(t) = 2^{1/2} \sum_n h[n] \phi(2t - n) \tag{3.19}$$

for some sequence $h[n]$, known as *refinement coefficients*. $\phi_{0,0}(t)$ is thus the solution of a *two scale difference equation* [95].

3.2.1 Properties of Scaling Functions

From the above discussion, we can derive some useful relationships concerning $h[n]$ and $\phi(t)$.

Integrating Eq. (3.19) over the whole time axis, we get

$$\sum_n h[n] = \sqrt{2} \tag{3.20}$$

Taking Fourier transform of both sides of Eq. (3.19), we get

$$\Phi(\omega) = \sqrt{2} \sum_n h[n] \int \phi(2t - n) e^{-j\omega t} dt$$
$$= \frac{1}{\sqrt{2}} \sum_n h[n] \int \phi(y) e^{-j\omega(y+n)/2} dy \qquad \text{putting } y = 2t - n$$
$$= \frac{1}{\sqrt{2}} \sum_n h[n] e^{-j\omega n/2} \int \phi(y) e^{-j\omega y/2} dy$$
$$= \frac{1}{\sqrt{2}} H(\omega/2) \Phi(\omega/2) \tag{3.21}$$

Using this relation recursively, we get

$$\Phi(\omega) = \prod_{j=1}^{\infty} \{ \frac{1}{\sqrt{2}} H(\frac{\omega}{2^j}) \} \tag{3.22}$$

Since, $\{\phi_{j,k}(t)\}$ is a set of orthogonal basis functions,

$$\langle \phi_{0,0}(t), \phi_{0,k}(t) \rangle = \delta_{0,k}$$

or, $$\langle \sum_n h[n]\phi_{-1,n}(t), \sum_m h[m]\phi_{-1,m+2k}(t) \rangle = \delta_{0,k}$$

or, $$\sum_n h[n]h[n-2k] = \delta_{0,k} \tag{3.23}$$

Since, $\{\phi_{j,k}(t)\}$ form an orthonormal base, their Fourier transform must satisfy

$$\sum_k |\Phi(\omega + 2k\pi)|^2 = 1 \tag{3.24}$$

From Eq. (3.21) and (3.24), it can be easily shown that

$$|H(\omega)|^2 + |H(\omega + \pi)|^2 = 2 \tag{3.25}$$

Also, from Eq. (3.20) and (3.25), it is obvious that,

$$H(\pi) = 0 \tag{3.26}$$

3.2.2 Wavelet Function

In Fig. 3.2, we see the embedded spaces of V_j for various j. Here, V_j contains V_k if $j < k$. We can also view the representation as a set of nonoverlapping rings, representing $L^2(R)$. Each ring will be the difference between two consecutive spaces. A difference space can be denoted by W_j and is defined as the orthogonal complement of V_j with V_{j-1}, *i.e.*,

$$V_j \oplus W_j = V_{j-1}, \qquad V_j \perp W_j \tag{3.27}$$

The containment and completeness property of $\{V_j\}$ together with Eq. (3.27) imply that the spaces $\{W_j\}$ are mutually orthogonal and that their direct sum is $L^2(R)$ [4]. Now, we can write

$$\bigcap_{j \in Z} W_j = \{\phi\}, \qquad \bigcup_{j \in Z} W_j = L^2(R) \tag{3.28}$$

Recall that,

$$V_0 = span\{\phi(t-k)\}, \; V_{-1} = span\{\phi(2t-k)\} \text{ and } V_0 \oplus W_0 = V_{-1}$$

Thus, it is reasonable to expect the existence of a function $\psi(t) \in W_0$ such that $W_0 = span\{\psi(t-k)\}$. This function $\psi(t)$ is the wavelet function associated with multiscale analysis. Let $\psi(t) = \psi_{0,0}(t)$ be a basis function of W_0.

Now, $\psi_{0,0}(t) \in W_0 \in V_{-1}$. Hence we can write

$$\psi_{0,0}(t) = \sum_n g[k]\phi_{-1,n}(t) \tag{3.29}$$

for a certain sequence $g[n]$. Since, $\{\phi_{j,k}(t)\}$ are translated and dilated versions of a mother function $\phi(t)$, we can also define functions that are dilated and translated versions of a mother function $\psi(t)$, *i.e.*,

$$\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}t - k) \tag{3.30}$$

These functions are identical to the wavelets in section 3.1. The parameter a_0 in Eq. (3.9) has been fixed at 2. $\psi_{j,k}(t)$ form an orthogonal basis for $L^2(R)$. The $2^{-j/2}$ factor maintains the equality of norm of the function in different scales.

Relation between $\phi(t)$, $\psi(t)$, $h[n]$ and $g[n]$

Taking the Fourier transform of both sides of Eq. (3.29) we get,

$$\psi(\omega) = \frac{1}{\sqrt{2}} G(\omega/2)\phi(\omega/2) \tag{3.31}$$

Replacing Eq. (3.22) in the above equation, we get

$$\psi(\omega) = G(\omega/2)\prod_{j=2}^{\infty}\{\frac{1}{\sqrt{2}}H(\frac{\omega}{2^j})\} \tag{3.32}$$

Now, since $\phi(t)$ and $\psi(t)$ are orthogonal, their inner product must be zero, *i.e.*,

$$\langle \phi_{0,0}(t), \psi_{0,0}(t) \rangle = 0$$

$$= \langle \sum_n h[n]\phi_{-1,n}(t), \sum_m g[m]\phi_{-1,m}(t) \rangle$$

$$= \sum_{m,n} h[n]g[m]\langle \phi_{-1,n}, \phi_{-1,m} \rangle$$

$$= \sum_n h[n]g[n] \tag{3.33}$$

The above relation is satisfied by the following choice

$$g[n] = (-1)^n h[N-1-n] \tag{3.34}$$

In Fourier domain this becomes,

$$G(\omega) = -e^{-j\omega(N-1)} H(-\omega + \pi) \tag{3.35}$$

In the above equations, N is the number of refinement coefficients representing the scaling equations. For compactly supported wavelets, N must be an even number. Taking the Fourier transform of both the sides and using Eq. (3.32), we get

$$\Psi(\omega) = G(\omega/2) \prod_{j=2}^{\infty} \{ \frac{1}{\sqrt{2}} H(\frac{\omega}{2^j}) \}$$

$$= -e^{-\frac{j\omega(N-1)}{2}} H(-\frac{\omega}{2} + \pi) \Phi(\frac{\omega}{2}) \tag{3.36}$$

Substituting for ω=0 in Eq. (3.35) and (3.36), we can conclude that $\psi(t)$ and $g[n]$ have zero mean, *i.e.*,

$$\int_{-\infty}^{\infty} \psi(t)dt = 0 \tag{3.37}$$

$$\sum_n g[n] = 0 \tag{3.38}$$

From the above discussion, we can conclude that the multiresolution analysis of wavelets has enabled us to write any function $x(t)$ in $L^2(R)$ as a sum of projections on W_j, $j \in R$,

$$x(t) = \sum_{-\infty}^{\infty} p_j(t) \tag{3.39}$$

where the projections $p_j(t)$ are calculated as

$$p_j(t) = \sum_k \langle \psi_{j,k}(t), x(t) \rangle \psi_{j,k}(t) \tag{3.40}$$

We can also express the function as the sum of a coarse resolution component and several detail components, *i.e.*,

$$\begin{aligned} x(t) &= x_j(t) + \sum_{m=-\infty}^{j} p_m(t) \\ &= \sum_k \langle \phi_{j,k}(t), x(t) \rangle \phi_{j,k}(t) + \sum_{m=-\infty}^{j} \sum_n \langle \psi_{m,n}(t), x(t) \rangle \psi_{m,n}(t) \\ &= \sum_k c_{j,k} \phi_{j,k}(t) + \sum_{m=-\infty}^{j} \sum_n d_{m,n} \psi_{m,n}(t) \end{aligned} \tag{3.41}$$

3.3. Discrete Wavelet Transform :

In practice, wavelet transform is applied on discrete data. Thus, one is interested in the discrete versions of wavelet transform. Unfortunately, there is no discrete basis function whose translated and dilated versions form a base of $l^2(R)$, the space of square summable infinite length sequence.

This situation can be tackled by observing that every discrete signal can be viewed as the sequence of weights of a set of scaling functions, $\phi_{-J,k}(t)$, representing a continuous function $x_{-J}(t) \in V_{-J}$ at the finest scale of interest, say -J. That means, with $c_{-J,k} = c_k$ we can express $x_{-J}(t)$as,

$$x_{-J}(t) = \sum_k c_{-J,k} \phi_{-J,k}(t)$$

This is a reasonable assumption because for high values of J, the support of $\phi_{-J,k}(t)$ will be very small and can be treated as *delta* functions. If the samples of x(t) are above Nyquist rate, they are good approximations to the DWT coefficients at that scale [71].

In other words, we can interpret the sequence c_k as if it were the representation $c_{-J,k}$ of some continuous function $x_{-J}(t) \in V_{-J}$ for a certain multiresolution space of $L^2(R)$.

From the function $x_{-J}(t)$ in V_{-J}, we compute its smoother versions, lying in the spaces $V_{-J+1}, V_{-J+2}, \ldots\ldots$. The spaces $V_{-J-1}, V_{-J-2}, \ldots\ldots$ are neglected in this interpretation.

Notwithstanding the values of the theory of scaling and wavelet functions, in practice one doesn't need them to calculate the transform. In this section, we will show how to calculate the transform coefficients recursively from one stage to another using the refinement coefficients $h[n]$ and $g[n]$.

3.3.1 Analysis :

In order to work directly with the coefficients $h[n]$ and $g[n]$, we need to derive a relationship between the expansion coefficients at a lower scale level, in terms of those at higher scale.

Let the function to be decomposed be $x_j(t) \in V_j$ represented by

$$\begin{aligned} x_j(t) &= \sum_m c_{j,m}\phi_{j,m}(t) \\ &= \sum_k c_{j+1,k}\phi_{j+1,k}(t) + \sum_l d_{j+1,l}\psi_{j+1,l}(t) \end{aligned}$$

(coarse + details)

where

$$\begin{aligned} c_{j+1,k} &= \langle \phi_{j+1,k}(t), x_j(t) \rangle \\ &= \langle \phi_{j+1,k}(t), \sum_m c_{j,m}\phi_{j,m}(t) \rangle \\ &= \sum_m c_{j,m} \langle \phi_{j+1,k}(t), \phi_{j,m}(t) \rangle \end{aligned} \tag{3.42}$$

But,

$$\begin{aligned} \langle \phi_{j+1,k}(t), \phi_{j,m}(t) \rangle &= \langle \sum_n h[n]\phi_{j,n+2k}(t), \phi_{j,m}(t) \rangle \\ &= \langle \sum_n h[n]\phi_{j,n+2k}(t), \phi_{j,m}(t) \rangle \\ &= \sum_n h[n] \langle \phi_{j,n+2k}(t), \phi_{j,m}(t) \rangle \\ &= h[m-2k] \end{aligned} \tag{3.43}$$

Then substituting the above result in Eq. (3.42), we get

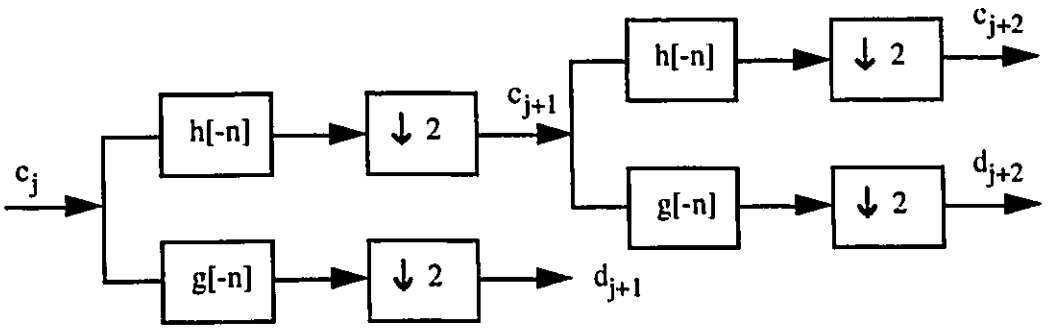


Figure 3.3 : Decomposition of signal using analysis filters

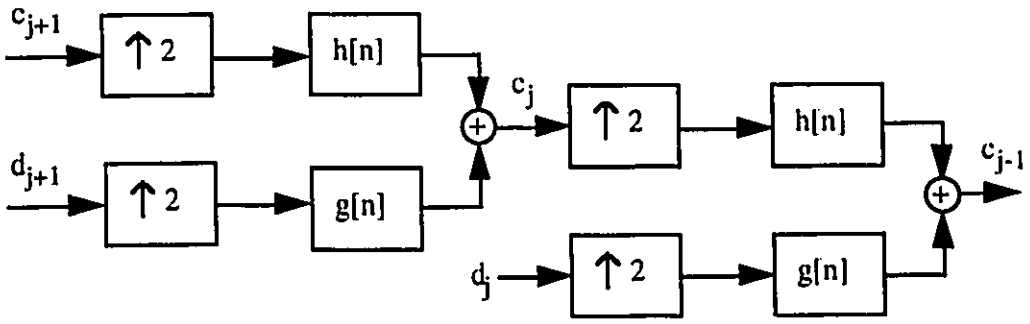


Figure 3.4 : Reconstruction of signal using synthesis filters

$$c_{j+1,k} = \sum_m c_{j,m} h[m-2k] \tag{3.44}$$

Similarly, we get

$$d_{j+1,k} = \sum_m c_{j,m} g[m-2k] \tag{3.45}$$

This shows that the scaling and wavelet coefficients $c_{j+1,k}$ at scale (j+1) can be obtained by convolving the coefficients at scale j by h[-n] and g[-n] (*i.e.*, the time reversed sequence of h[n] and g[n]) respectively, followed by downsampling or decimating (keeping every other term) by two. In other words, the scale (j+1) coefficients are filtered by two FIR digital filters with coefficients h[-n] and g[-n], which after downsampling give the next coarser scaling and wavelet coefficients. This is shown schematically in Fig. 3.3. This has been extensively studied in filterbank

theory and first applied to calculate wavelet coefficient by Mallat [61]. In the wavelet literature, this type of decomposition is known as *Mallat's tree algorithm.*

3.3.2 Synthesis

Similar to the analysis stage, the reconstruction of the original fine scale coefficients can be made from the coarser coefficients.

Again, assume $x_j(t) \in V_j$, so it can be represented by

$$\begin{aligned} x_j(t) &= \sum_m c_{j,m}\phi_{j,m}(t) \\ &= \sum_k c_{j+1,k}\phi_{j+1,k}(t) + \sum_l d_{j+1,l}\psi_{j+1,l}(t) \end{aligned}$$

Now, we need to calculate $c_{j,m}$ from $c_{j+1,k}$ and $d_{j+1,l}$.

$$\begin{aligned} c_{j,m} &= \langle \phi_{j,m}(t), x_j(t) \rangle \\ &= \langle \phi_{j,m}(t), \sum_k c_{j+1,k}\phi_{j+1,k}(t) + \sum_l d_{j+1,l}\psi_{j+1,l}(t) \rangle \\ &= \langle \phi_{j,m}(t), \sum_k c_{j+1,k}\phi_{j+1,k}(t) \rangle + \langle \phi_{j,m}(t), \sum_l d_{j+1,l}\psi_{j+1,l}(t) \rangle \\ &= \sum_k c_{j+1,k}\langle \phi_{j,m}(t), \phi_{j+1,k}(t) \rangle + \sum_l d_{j+1,l}\langle \phi_{j,m}(t), \psi_{j+1,l}(t) \rangle \\ &= \sum_k c_{j+1,k}h[m-2k] + \sum_l d_{j+1,l}g[m-2l] \end{aligned} \tag{3.46}$$

The above equation is evaluated by upsampling the (j+1) *s*cale coefficient sequence $c_{j+1}(k)$ by two (*i.e.* inserting zero between each coefficient), convolving it with the *refinement* coefficients $h[n]$. The same is done to the (j+1) level wavelet coefficient (but convolved with $g[n]$) sequence and both results are added to give the j level scaling function coefficients. This is schematically shown in Fig. 3.4.

The analysis and synthesis procedure derived above enables one to calculate recursively the transform coefficients. If one level of coefficients is available, the next level coefficients can be calculated.

3.3.3 Wavelet Transform versus Filter Banks

In this section, we describe the requirements for designing a two channel paraunitary FIR filterbank. It will be seen that the requirements for paraunitary FIR filterbanks are

very similar to that of wavelets, derived in the previous sections. An FIR filterbank will be called paraunitary if the impulse response of the filters in each bank satisfies orthonormality constraints. The paraunitary filterbanks with dyadic tree structure are identical to wavelets if the low pass filter has at least one vanishing moments *i.e.*, H(z) has at least one zero at z=-1.

The schematic diagram of one stage of a two channel FIR filterbanks is shown in Fig. 3.5. $h[n]$ and $g[n]$ are, respectively, the impulse response of the lowpass and highpass synthesis filters. $\tilde{h}[n]$ and $\tilde{g}[n]$ are the impulse response of the lowpass and highpass analysis filters, respectively. Here, we have defined $\tilde{h}[n]$ as $h[-n]$, *i.e.*, the time-reversed sequence of $h[n]$.

The output at various points of the lowpass section can be expressed in the z-transform domain as [100]

$$X_0(z) = X(z)\tilde{H}(z) \tag{3.47}$$

$$V_0(z) = \frac{1}{2}\left[X_0(z^{1/2}) + X_0(-z^{1/2})\right] \tag{3.48}$$

$$\hat{X}_0(z) = V_0(z^2) \tag{3.49}$$

$$Y_0(z) = H(z)\hat{X}_0(z)$$

Combining the above equations along with the corresponding outputs of the high pass sections, we get

$$\begin{aligned}\hat{X}(z) &= \frac{1}{2}\left[\tilde{H}(z)H(z) + \tilde{G}(z)G(z)\right]X(z) \\ &\quad + \frac{1}{2}\left[\tilde{H}(-z)H(z) + \tilde{G}(-z)G(z)\right]X(-z) \\ &= B(z)X(z) + A(z)X(-z)\end{aligned} \tag{3.50}$$

For perfect reconstruction (PR), the following conditions must be satisfied

i) Alias component, A(z) =0

ii) B(z) = cz^{-k}, *i.e.*, it should only represent a delay (c and k are constants)

The above conditions will ensure the perfect reconstruction. However, the conditions for paraunitary filterbanks are more stringent. It can be shown that a two band

paraunitary filter bank must satisfy the conditions in Table 3.1, which have been provided both in time and z-transform domain for convenience [4].

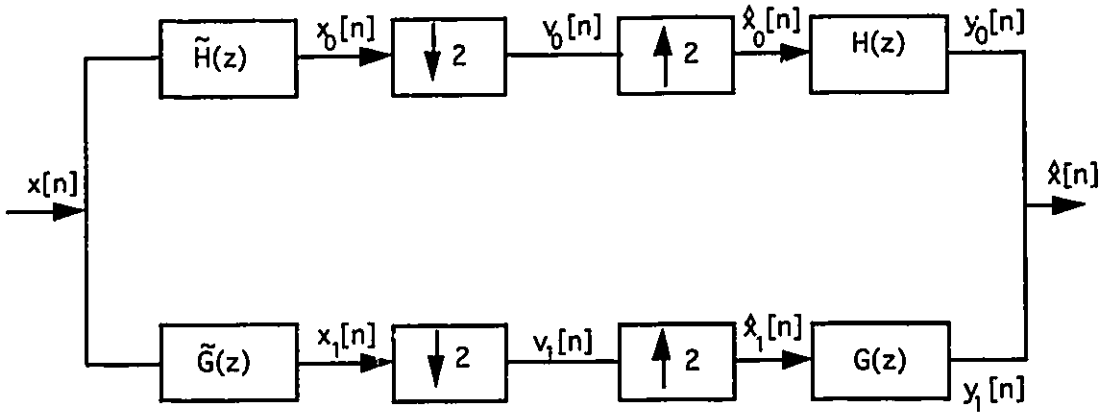


Figure 3.5 : Two channel filter bank

Table- 3.1
Conditions for paraunitary filterbanks

Time Domain	Frequency Domain
$h[n]=(-1)^{n+1}\tilde{g}[n]$	$H(z)=-\tilde{G}_1(-z)$
$g[n]=(-1)^n\tilde{h}[n]$	$G(z)=\tilde{H}(-z)$
$\tilde{g}[n]=(-1)^{n+1}\tilde{h}[N-1-n]$	$\tilde{G}(z)=z^{-(N-1)}\tilde{H}(-z^{-1})$
$\sum_{n=0}^{N-1}h[n]h[n-2k]=\delta[k]$	$H(z)H(z^{-1})+H(-z)+H(-z^{-1})=2$

It is clear that the conditions in Table 3.1 for paraunitary filterbanks have identical counterparts in the constraints imposed on the *refinement coefficients* $h[n]$ in section 3.2 (see Eq. 3.23, 3.34). However, the refinement coefficients satisfies another constraint ensuring at least one vanishing moment of the *wavelet* function. This additional constraint of vanishing moment forces the basis functions to have some degree of smoothness. It should also be noted that the output of the above system is identical to the input within a delay of (N-1) samples, where N is the number of filter coefficients.

3.4 Implementation of the Discrete Wavelet Transform

In section 3.3, we described the calculation of wavelet transform of a data sequence using Mallat's *tree* algorithm. We observed that the sampled data could be considered as the coefficients of -*J* level scaling functions, for high enough *J*. In this section, calculation of the wavelet transform will be considered for finite length data. First, we describe discrete wavelet transform as a series of matrix operations. This approach enables us to view DWT as any other orthogonal transform. Then we describe it as a filterbank implementation. This will enable us to view DWT as a special case of filterbanks. For describing the implementation, we have assumed compactly supported orthonormal wavelets (*i.e.* a finite number of *refinement* coefficients). For other types of wavelets (*e.g.* bi-orthogonal, IIR wavelets), the diagram has to be modified accordingly.

In section 3.3.3, we have seen the requirements for a perfect reconstruction (PR) system. However, both cases assumed an infinite length sequence. For finite length data, PR is not automatically guaranteed. The problem arises at both ends of the data sequence during each level of decomposition. When the system starts operation, it does not have sufficient input data (at least as many as the filter coefficients) for convolution. As a result, to achieve PR, one has to store extra (L-1) coefficients (arising out of linear convolution), where L is the number of filter coefficients. This increase in data-rate is not desirable in image compression application. However, PR can be achieved even with the same data-rate if we extend the data periodically. The logical reason for this can be as follows.

Let us assume, we have a periodic input, with period L. The output of the two-band PR system will also be periodic with period L (since PR is assured for signals with infinite duration) with a delay of (N-1) sample. Thus there are only L different consecutive output samples that are being repeated to produce the periodic output. From any L number of consecutive output samples, we will be able to generate the

whole output sequence. Thus, with periodic extension of data, we keep only L output samples. This L output sample is nothing but the circularly shifted version of the input. With proper shift, we will be able to get the original data.

Other type of extensions *e.g., zero padding, data repetition, symmetrical extension,* do not provide perfect reconstruction. However, for the special case of *symmetric* filters, symmetrical extension of data can provide perfect reconstruction. Symmetrical extension methods, in general provide results better than the periodic extension methods as it can efficiently handle the image borders [93]. This method is generally applied to *bi-orthogonal* [18] cases where the filters are symmetric and also can be applied to complex symmetrical orthonormal wavelets. Unfortunately, for real orthonormal wavelets, it can not be used as the filters are always asymmetric.

3.4.1 Matrix Multiplication

Any orthonormal transform is associated with a unitary transformation matrix. The calculation of forward transform coefficient can be done by multiplying the transformation matrix with the data vector. Let us, assume that we have a wavelet with four *refinement* coefficients, (h[n], n = 0,1,2,3), the data vector of length 8 (associated with j level scaling functions), is $\{c_{j,k}, 0 \le k \le 7\}$. The scaling and wavelet coefficients for the (j+1) level can be calculated as follows [78].

$$\begin{bmatrix} c_{j+1,0} \\ d_{j+1,0} \\ c_{j+1,1} \\ d_{j+1,1} \\ c_{j+1,2} \\ d_{j+1,2} \\ c_{j+1,3} \\ d_{j+1,3} \end{bmatrix} = \begin{bmatrix} h[0] & h[1] & h[2] & h[3] & & & & \\ h[3] & -h[2] & h[1] & -h[0] & & & & \\ & & h[0] & h[1] & h[2] & h[3] & & \\ & & h[3] & -h[2] & h[1] & -h[0] & & \\ & & & & h[0] & h[1] & h[2] & h[3] \\ & & & & h[3] & -h[2] & h[1] & -h[0] \\ h[2] & h[3] & & & & & h[0] & h[1] \\ h[1] & -h[0] & & & & & h[3] & -h[2] \end{bmatrix} \begin{bmatrix} c_{j,0} \\ c_{j,1} \\ c_{j,2} \\ c_{j,3} \\ c_{j,4} \\ c_{j,5} \\ c_{j,6} \\ c_{j,7} \end{bmatrix} \quad (3.51)$$

The first row of the transformation matrix is nothing but the *refinement* coefficients of the scaling functions. The *blank entries* are zeroes. The third, fifth and seventh rows of the transformation matrix are identical except for an even shift. The second, fourth, sixth and eighth rows correspond to coefficients of the *wavelet* function. The last two rows of the matrix, wrap around as in circular convolutions to make the matrix unitary. We note that there are two types of output data - $\{c_{j+1,k}\}$ (the coarse component) and $\{d_{j+1,k}\}$ (the detail component). After each stage of decomposition, the data is sorted out in the above two categories. The detail components are kept as they are. However, the coarse components are further decomposed recursively with a similar approach.

The inverse transformation matrix will be the transpose of the forward transformation matrix as the matrix is unitary. Thus, the inverse transformation will be as follows.

$$\begin{bmatrix} c_{j,0} \\ c_{j,1} \\ c_{j,2} \\ c_{j,3} \\ c_{j,4} \\ c_{j,5} \\ c_{j,6} \\ c_{j,7} \end{bmatrix} = \begin{bmatrix} h[0] & h[3] & & & & & h[2] & h[1] \\ h[1] & -h[2] & & & & & h[3] & -h[0] \\ h[2] & h[1] & h[0] & h[3] & & & & \\ h[3] & h[0] & h[1] & -h[2] & & & & \\ & & h[2] & h[1] & h[0] & h[3] & & \\ & & h[3] & -h[0] & h[1] & -h[2] & & \\ & & & & h[2] & h[1] & h[0] & h[3] \\ & & & & h[3] & -h[0] & h[1] & -h[2] \end{bmatrix} \begin{bmatrix} c_{j+1,0} \\ d_{j+1,0} \\ c_{j+1,1} \\ d_{j+1,1} \\ c_{j+1,2} \\ d_{j+1,2} \\ c_{j+1,3} \\ d_{j+1,3} \end{bmatrix} \tag{3.52}$$

3.4.2 Filterbank approach

In this approach also, we use periodic extension to ensure perfect reconstruction. Filterbanks are implemented as described in Fig. 3.5 with little modification. Recall that in image compression applications, the wavelet decomposed images (*i.e.* transform coefficients) are often used in progressive image transmission, image analysis *etc.* Hence, the coarse resolution images should appear as a smaller version of the original, if displayed on a monitor. However, with periodic extension of data, there is a possibility that the image may be circularly shifted with a maximum of L-1

pixels in both dimensions, where L is the number of refinement coefficients. This can be rectified by adding appropriate shift while doing the convolution. This is shown in Fig. 3.6.

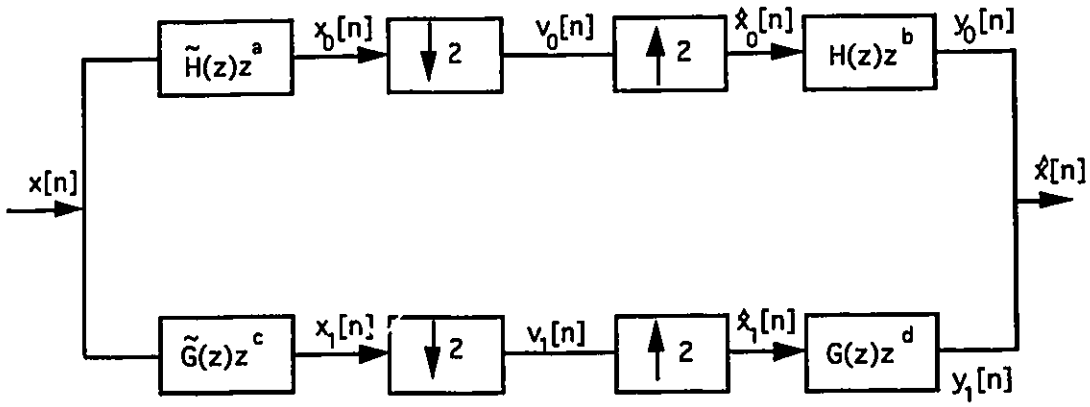


Figure 3.6 : Modified two channel filter bank

For perfect reconstruction, we require that a+b = c+d, *i.e.* the total amount of shift in both channels should be equal. Moreover, we have seen in section 3.3.3 that the filterbank produces the output with a delay of N-1, where N is the filter length. Hence, if we make a+b = N-1 (maintaining a=c, b=d), both delays will be cancelled and perfect reconstruction will be guaranteed. In our implementation, we have chosen, a = c = N/2-1 and b = d = N/2. The complete operation is shown in Fig. 3.7. In the following discussion, we have assumed an input data of length 6, extended periodically. We have also assumed a wavelet with 6 coefficients.

Original data, X = { a, b, c, d, e, f}

Refinement Coeff. = { P, Q, R, S, T, U}

3.5 Parameterization of Wavelets

Unlike the Fourier transform which has a unique exponential base, wavelet bases are not unique [77, 106, 114]. In this section, we will describe the generation of various wavelet bases in a systematic way using parameterization. This parameterization of

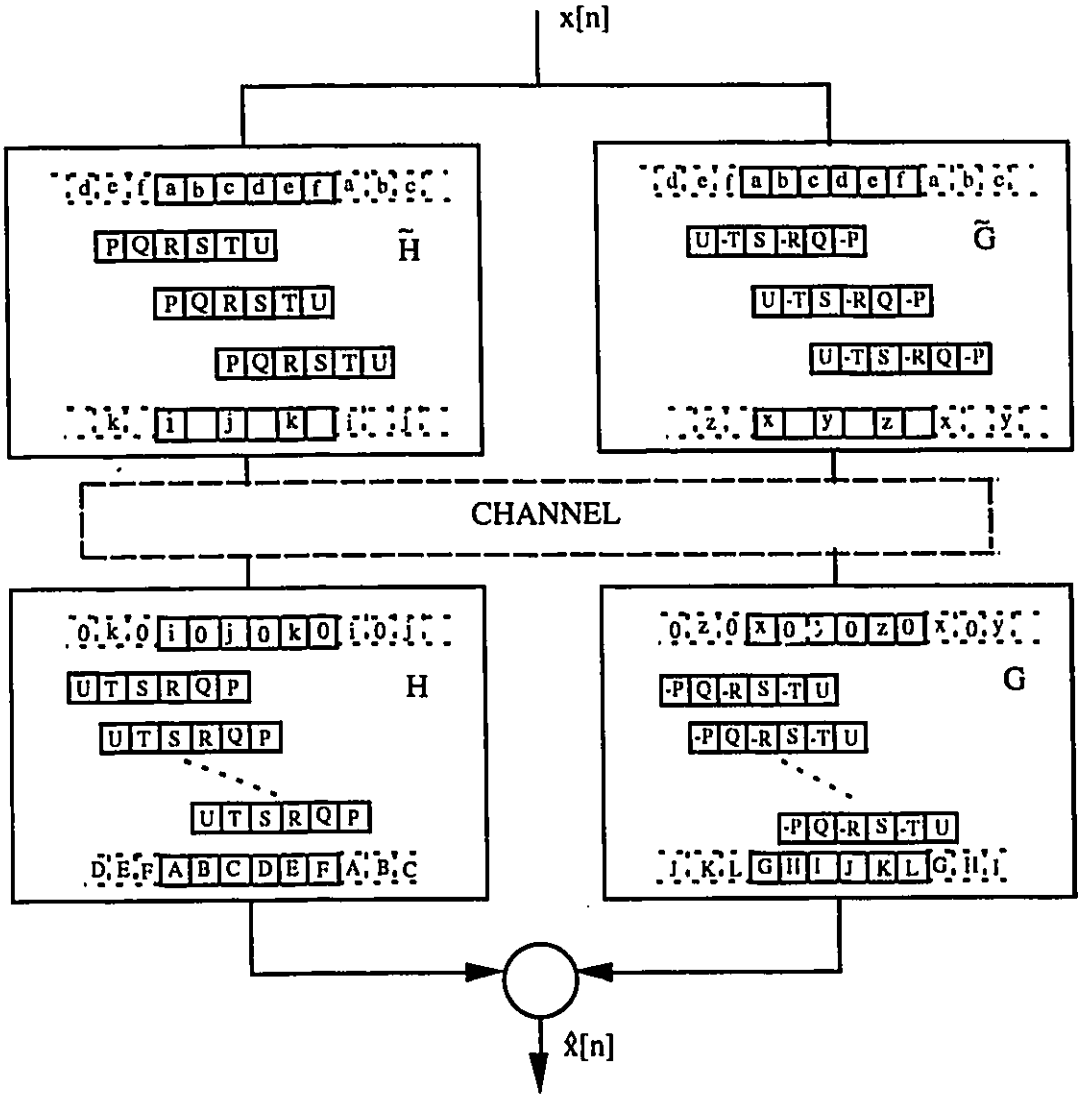


Figure 3.7 : Periodic extension of finite length data

wavelets is very useful when searching for the optimal wavelet for some applications. Here, we will limit our discussion to real orthonormal compactly supported wavelets. In section 3.2.1, it was shown that to qualify as wavelets, the refinement coefficients should satisfy the following conditions (see Eqs. 3.20 and 3.23)

$$\sum_n h[n] = \sqrt{2}$$

$$\sum_n h[n]h[n-2k] = \delta_{0,k}$$

Thus, for an N coefficient wavelet, the coefficients should satisfy one linear and N/2 bi-linear conditions. Since to determine N coefficients uniquely, one needs N conditions; we are left with (N/2-1) degrees of freedom which can be used to generate a wide variety of wavelet functions.

Pollen's Parameterization [77]

A general parameterization for constructing FIR wavelet has been reported by Pollen [77]. Here, we will provide the construction formula for 4 and 6 tap wavelets. For 4 coefficient wavelets, we have only one degree of freedom left. The filter parameters can be expressed in terms of one parameter α $(0 \le \alpha \le 2\pi)$ as follows.

$$
\begin{aligned}
h[0] &= \frac{1}{2\sqrt{2}}[1-\cos(\alpha)+\sin(\alpha)] \\
h[1] &= \frac{1}{2\sqrt{2}}[1+\cos(\alpha)+\sin(\alpha)] \\
h[2] &= \frac{1}{2\sqrt{2}}[1+\cos(\alpha)-\sin(\alpha)] \\
h[3] &= \frac{1}{2\sqrt{2}}[1-\cos(\alpha)-\sin(\alpha)]
\end{aligned} \tag{3.53}
$$

For 6 tap filters, we have two degrees of freedom. The filter coefficients are then expressed in terms of two parameters α and β. The parameterization becomes :

$$
\begin{aligned}
h[0] &= \frac{1}{4\sqrt{2}}[(1+\cos(\alpha)+\sin(\alpha))(1-\cos(\beta)-\sin(\beta))+2\sin(\beta)\cos(\alpha)] \\
h[1] &= \frac{1}{4\sqrt{2}}[(1-\cos(\alpha)+\sin(\alpha))(1+\cos(\beta)-\sin(\beta))-2\sin(\beta)\cos(\alpha)] \\
h[2] &= \frac{1}{2\sqrt{2}}[1+\cos(\alpha-\beta)+\sin(\alpha-\beta)] \\
h[3] &= \frac{1}{2\sqrt{2}}[1+\cos(\alpha-\beta)-\sin(\alpha-\beta)] \\
h[4] &= \frac{1}{2\sqrt{2}}[1-h[0]-h[2]] \\
h[5] &= \frac{1}{2\sqrt{2}}[1-h[1]-h[3]]
\end{aligned} \tag{3.54}
$$

By changing α and β over $[0,2\pi]$, one can generate numerous 6-tap wavelet bases. For 4 and 6-tap wavelets, this parameterization technique is quite simple to use.

However, higher order wavelets are difficult to parameterize with the technique of Pollen. Recently, Zou *et al.* [114] have developed a simpler technique which we will describe briefly in the following.

Zou's Parameterization

General parameterization for paraunitary filterbanks has been studied by Vaidyanathan [99]. Zou *et al.* have used the same parameterization for generating wavelets by adding an extra constraint to ensure the presence of at least one vanishing moment [114].

Let H(z) and G(z) be the z-transform of the lowpass and highpass sequences, $h[n]$ and $g[n]$, respectively, of a given paraunitary filterbank. Then

$$H(z) = \sum_k h_k z^{-k}$$
$$G(z) = \sum_k g_k z^{-k}$$

where $z = e^{j\omega}$. We can write the polyphase representation of the above as follows [99]:

$$\begin{bmatrix} H(z) \\ z^{-2(N-1)}G(z) \end{bmatrix} = E(z^2)\begin{bmatrix} 1 \\ z^{-1} \end{bmatrix} \tag{3.55}$$

where,

$$E(z) = V_{N-1}(z)V_{N-2}(z).....V_1(z)V_0$$
$$V_k(z) = I - (1 - z^{-1})v_k v_k^T$$
$$v_k = \begin{bmatrix} \cos(\theta_k) \\ \sin(\theta_k) \end{bmatrix}$$
$$V_0 = \begin{bmatrix} \cos(\theta_0) & -\sin(\theta_0) \\ \sin(\theta_0) & \cos(\theta) \end{bmatrix} \tag{3.55a}$$

where $k = 1,.....,L/2$, where L is the order of the filter.

The above parameterization is for general paraunitary filterbank [99]. The construction of a filterbank is as follows : For N tap general filterbank, there will be N/2 free parameters. Let the free parameters be $\{\theta_k, k = 0,...N/2-1\}$. Then, from Eqs.

(3.55a), we can express H(z) as a polynomial of z^{-1}. The coefficients of the polynomial are the functions of $\{\theta_k, k = 0,...N/2-1\}$. Now, changing θ_k's, we can generate various filterbanks. The construction of wavelet is simple. It can be shown that H(z) will have a zero at $\omega = \pi$, by making $\theta_0 = \frac{7\pi}{4}$ [114]. With this constraint, the parameterization becomes valid for wavelet. Other (N/2-1) free parameters can be chosen to construct a wavelet with various desirable properties. We note that θ is in the range $[0, \pi]$. For 4 coefficient wavelets, if we define the parameter as θ, the coefficients become

$$\begin{aligned} h[0] &= \sin(\theta)\sin(\theta + \pi/4) \\ h[1] &= \sin(\theta)\sin(\theta - \pi/4) \\ h[2] &= \cos(\theta)\sin(\pi/4 - \theta) \\ h[3] &= \cos(\theta)\sin(\pi/4 + \theta) \end{aligned} \tag{3.56}$$

Similar formula for 6-tap and 8-tap cases have been provided in Appendix B.

3.6 Regularity and Vanishing Moments

In this section, we will show how to generate the basis functions of a wavelet transform, *i.e.*, the scaling and the wavelet function. This is followed by a discussion of the regularity and vanishing moment of wavelets. It will be seen that the number of vanishing moments increases the regularity of the wavelet. It will also be observed that higher number of vanishing moments of a wavelet provides better energy compaction and hence increases the coding performance.

3.6.1 Generation of Scaling and Wavelet Function

The simplest way for generating a basis function of any orthonormal transform is to take the inverse transform of a single impulse function. Depending on the position of the impulse, it will generate basis function with different frequency or scale. Fig. 3.4

shows the synthesis scheme of a wavelet transform for only two stages. In the z-transform domain it can be written as,

$$C_{j-1}(z) = C_j(z^2)H(z)$$

Iterating this for m stages, we can write

$$C_{j-m}(z) = C_j(z^{2^m})H^m(z)$$

$$\text{where, } H^m(z) = H(z^{2^{m-1}})H(z^{2^{m-2}}).....H(z^{2^1})H(z)$$

We would like to see the graphs of impulse response $h^m[n]$ (the inverse z-transform of $H^m(z)$) as m increases. We note that $h^m[n]$ is of length $(2^m - 1)(L-1)$ where L is the number of refinement coefficients. Since, the number of impulse response increases exponentially, the graphs are normally plotted against $n2^{-m}$ and may converge to a limit function of a continuous variable $\phi(t)$. This has been described as scaling function in section 3.2. Fig. 3.8 shows how the limit function converge to a smooth function for h = {0.482962913144, 0.836516303737, 0.224143868042. -0.129409522551} (later, we will see that this is *Daubechies* 4 tap filter). Fig. 3.9 shows one example, where the limit function diverges. This FIR lowpass filter with 8 tap, was designed with Smith-Barnwell method, having a stopband attenuation of 40 dB. Since, the filter does not have any vanishing moment, we see that the limit function does not exist and hence cannot produce a wavelet. Once, the scaling function has been generated, the wavelet function can be generated by using Eq. (3.29). Fig. 3.8(g) shows the wavelet corresponding to the scaling function.

3.6.2 Regularity

Regularity is a new filter property introduced by wavelet theory. Traditional subband coding attempted to improve the coding performance by optimizing various filter properties *e.g.* frequency selectivity, phase linearity, *etc.* Wavelet theory requires another constraint - the basis function should be regular. In simple terms, regularity is a smoothness requirement on a continuous function and can be defined as the

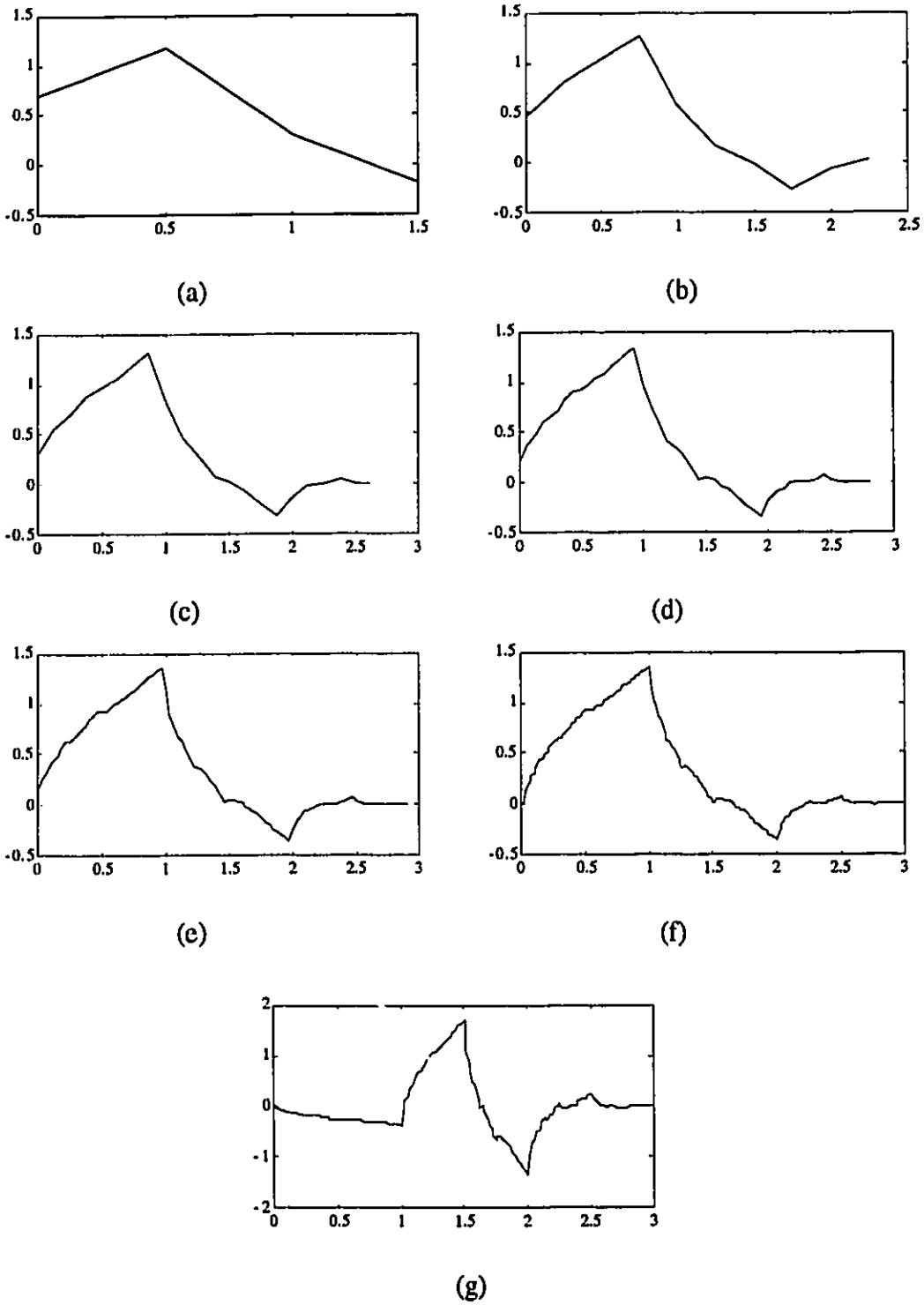


Figure 3.8 : Limit function generated by successive iterations of lowpass filters (in this case Daub-4). (a) i=1, (b) i=2, (c) i=3, (d) i=4, (e) i=5, (f) i=10. (g) corresponding wavelet function.

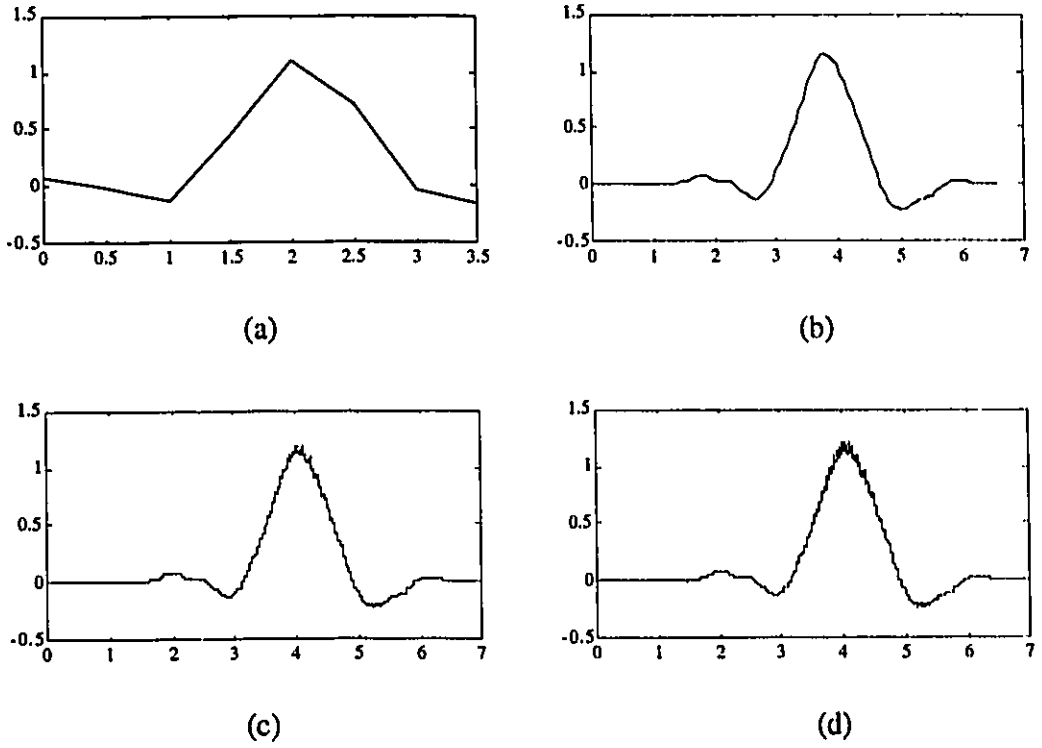


Figure 3.9 : Limit function generated by successive iterations of lowpass filters. (a) i=1, (b) i=4, (c) i=8, (d) i=9. Here 8 tap Smith-Barnwell filter (with 40 dB attenuation) has been used. We see that the limit diverges since, $H(-1) \neq 0$

continuity of the function and its derivatives. A more regular function has a larger number of continuous derivatives.

Regularity has been shown to be of significant importance in image coding applications. When we represent a signal in terms of transform coefficients, the signal is basically represented as a superposition of basis functions weighted by the transform coefficients. As a result, a smooth signal can be represented by a few transform coefficients if the basis functions are smooth. However, if the basis functions are irregular or fractal in nature, more transform coefficients will be required to represent the signal. Also, there will be a discontinuity in the transform coefficients (they will be somewhat random in nature). On the other hand, if the signal is fractal in nature, a fractal-like basis function will be more efficient in representing the function.

Since most natural images are smooth, a regular basis function will be very helpful in image processing applications in general and image compression in particular.

Note that when a transform coefficient is quantized, the reconstruction error is proportional to the basis function corresponding to that transform coefficient. Now, if the basis function is irregular, the reconstruction error will also be irregular in nature and will result in a lower subjective quality [61]. It is thus expected that a regular wavelet will provide a better subjective performance compared to an irregular wavelet given otherwise identical compression systems.

Estimation of Regularity

The regularity of *Meyer* (shifted Gaussian) or *Shanon* (sinc basis) wavelets is easy to determine. Since, they are infinitely differentiable, they have infinite regularity. However, the regularity of compactly supported wavelets is hard to determine. Techniques to estimate the regularity of a wavelet fall into two categories : those based on Fourier transform of $\phi(t)$ and those that work directly with $\phi(t)$ [24]. The Fourier-based technique are better suited for asymptotic estimates. However, the time domain method gives better local estimates, but is harder to use. Here, we will review one simple Fourier-based defintion of regularity [86].

The regularity of the scaling function is defined as the maximum value of r such that

$$|\Phi(\omega)| \leq \frac{c}{(1+|\omega|)^{r+1}}, \qquad \omega \in R \tag{3.57}$$

From the above equation, it is anticipated that faster decay of $|\Phi(\omega)|$ implies higher regularity. One simple way to ensure the fast decay of $|\Phi(\omega)|$ is to put as many zeros as possible at z=-1 of the lowpass wavelet filter. It is shown in Appendix-C that if H(z) has L zeroes at z=-1, $|\Phi(\omega)|$ decays as fast as $\left[\mathrm{sinc}\left(\frac{\omega}{2}\right)\right]^L$. More generally, H(z) must have at least N+1 zeroes to achieve a regularity order of N. This constructive result provides a simple rule for designing regular filters. Accordingly,

Daubechies [24] designed orthonormal wavelet filters by imposing as many zeroes of H(z) as possible at z=-1 with the orthonormality constraints (will be discussed in more detail in section 3.7.1). It turns out that these filters are maximally flat at $\omega = \pi$. However, they are not maximally regular for a given filter length. This is mainly because, much of the smoothness earned by the zeroes at z=-1 is destroyed by the zeroes at other locations.

How much regularity is useful ?

Intuitively, regularity is a useful property in image coding. However, there is no quantification of the optimum regularity of a wavelet for image coding application. Traditional subband coding filters did not pay any attention to the smoothness properties of the basis functions. Yet, many of them provide very good performance, inspite of their low regularity (or, some times negative regularity) [42]. One more point to be noted, the scaling or wavelet function are obtained after infinite iteration. However, in practice, one never goes beyond 5 levels of decomposition (though, the shape of the scaling function is obtained after 5 or 6 stages). Hence, optimizing the regularity of wavelets (since we are not using them) may not give optimal coding performance.

Given that most natural images can be well represented with a piecewise polynomial of order 2, a regularity of 2 is believed to be enough for representing an image [113]. Some of the free parameters for designing a wavelet should thus be utilized to improve other filter properties more useful in image coding applications.

3.6.3 Vanishing Moments

A wavelet is said to have M vanishing moments if its wavelet function satisfies the following condition

$$\int_{-\infty}^{\infty} t^m \psi(t) dt = 0 \qquad \text{for m} = 0, 1, \qquad \text{M-1}$$

We will show shortly that the above condition will be satisfied only if the lowpass filter H(z) has at least M zeros at z=-1. In the previous sections, we have seen that the zeros of H(z) at z=-1, leads to higher regularity. Hence, a function with more vanishing moments is expected to have higher regularity. However, this is not always true. Daubechies and Lagarias have proven that the highest vanishing moment solution does not lead to the highest regularity wavelet [24]. They devised counter examples of wavelets with higher regularity for the same support width, but with a reduced number of zeros at z=-1. In the previous section, we explained why regularity is an important property in image processing applications. Here, we will show the importance of vanishing moments in image coding applications [10].

We recall from Eq. (3.36) that

$$\Psi(\omega) = -e^{-\frac{j\omega(N-1)}{2}} H(-\frac{\omega}{2}+\pi)\Phi(\frac{\omega}{2})$$

Taking derivatives of $\Psi(\omega)$ with respect to ω at $\omega = 0$ and since $H(\omega)$ has M zeros at $\omega = \pi$, we get

$$\left.\frac{d^m\Psi(\omega)}{d\omega^m}\right|_{\omega=0} = \left.\frac{d^m H(-\omega/2+\pi)}{d\omega^m}\right|_{\omega=0} = 0$$

This in turn implies that

$$\begin{aligned}\left.\frac{d^m\Psi(\omega)}{d\omega^m}\right|_{\omega=0} &= \left.\frac{d^m}{d\omega^m}\int_{-\infty}^{\infty}\psi(t)e^{-j\omega t}dt\right|_{\omega=0} \\ &= (-j)^m\int_{-\infty}^{\infty}t^m\psi(t)dt \\ &= 0\end{aligned}$$

Or, in others words,

$$\int_{-\infty}^{\infty}t^m\psi(t)dt = 0 \qquad (3.58)$$

Thus the wavelet function has M zero moments if H(z) has M zeroes at z=-1. The corresponding formula with filter coefficients is obtained in (C.7) (see appendix C) as :

$$\sum_n n^m g[n] = 0$$

The above result is very useful in signal coding applications. We know that a signal can be represented as a piecewise polynomial. If the degree of the polynomial is less than M (the vanishing moment of the wavelet), the highest-pass transform coefficients will all be zeroes and hence can be discarded, resulting in a high compression ratio.

It is also advantageous to have more vanishing moments when a wavelet decomposition scheme (often a dyadic structure) is cascaded a number of times. For a non-ideal lowpass filter there will be some leakage in the highpass region. It is important to contain this leakage, especially when the filters are cascaded, otherwise it can lead to audible or visible aliasing, once quantization is introduced. This leakage can be reduced by increasing the vanishing moments of the wavelet function [24].

It is argued that vanishing moment properties of a regular wavelet (a highly regular wavelet has also a high number of vanishing moments) is more important than its smoothness for image coding application [35]. Strictly speaking, the basis functions of a wavelet decomposition scheme are always smooth and can be expressed as a linear combination of *sinc* functions. This is because, we can interpret the samples $x\left(\frac{n}{T}\right)$ as the Nyquist rate samples of a bandlimited signal. This means that the function is being represented by a set of sinc basis function with the sample value as the weights. These turn out to be the expansion coefficients with respect to the *sinc* basis. Now, the *sinc* function is infinitely differentiable. When we decompose the function, recursively, the basis functions at any level are linear combinations of the *sinc* functions (if the filter is FIR), which are infinitely smooth, regardless of whether the

filters are regular or not. With this interpretation, the basis function of wavelet transform is not the wavelet function, rather the convolution of the wavelet function and sinc function, which are always smooth. Thus the smoothness of the underlying functions, may not be a valid argument for using a highly regular filter.

3.7 Design of Wavelet Filters

In this section, we will describe the design procedure for two popular wavelet families, namely *Daubechies* and *Coiflet* family.

3.7.1 Daubechies wavelets

In the previous section, we have discussed wavelets with maximum vanishing moments. Daubechies [22, 24] was the first to construct these wavelets. Even though she constructed many other wavelets [18, 22], we generally refer to this particular class as *Daubechies* wavelet. These wavelets with maximum number of vanishing moments have corresponding filters that are maximally flat at $\omega = 0$ and π. It turns out that the coefficient sequence of Daubechies wavelet is identical to the unit sample response of the Binomial QMF proposed by Akansu *et al.* [4]. In the following, we present a design procedure of *Daubechies* wavelet with various support length.

A lowpass filter with L zeroes at $\omega = 0$ can be expressed as (see appendix C)

$$\left|H(e^{j\omega})\right| = \left|\cos\left(\frac{\omega}{2}\right)\right|^{L} \left|P(e^{j\omega})\right| \tag{3.59}$$

Since, P(z) is a polynomial in z^{-1} (where $z = e^{j\omega}$) with real coefficients, $Q(z) = P(z)P(z^{-1})$ is a symmetric polynomial, *i.e.*,

$$Q(z) = \sum_{-M}^{M} q_n z^{-n}, \quad M = N - 1 - L \text{ and } q_n = q_{-n}$$

Thus,

$$Q(z) = \left|P(e^{j\omega})\right|^2 = q_0 + 2\sum_{n=1}^{M} q_n \cos(n\omega)$$

Thus, $\cos(n\omega)$ can be expressed as a polynomial of $\cos(\omega)$, which in turn can be represented in terms of $\sin^2(\omega/2)$. Therefore, $|P(e^{j\omega})|^2$ can be expressed as some function of $\sin^2(\omega/2)$ of degree (N-1-L).

$$|H(e^{j\omega})|^2 = \left|\cos^2\left(\frac{\omega}{2}\right)\right|^L f\left(\sin^2(\omega/2)\right) \tag{3.60}$$

$$= (1-y)^L f(y)$$

where $y=\sin^2(\omega/2)$. Substituting the above in Eq. (3.25) (with a normalization of 2) gives

$$(1-y)^L f(y) + y^L f(1-y) = 1 \tag{3.61}$$

The solution of this equation is of the form [24]

$$f(y) = \sum_{k=0}^{L-1}\binom{L-1+k}{k} y^k + y^L R\left(\frac{1}{2}-y\right) \tag{3.62}$$

where, $\binom{L-1+k}{k}$ is a binomial sequence and R(y) is an odd polynomial chosen so that $f(y) \geq 0$ for $0 \leq y \leq 1$.

There are many choices for R(y) and L each leading to a different solution. From section 3.2.1, we note that to qualify as N tap FIR wavelets, N/2 quadratic conditions and one linear condition must be satisfied. These conditions imply a zero of $H(\omega)$ at $\omega = \pi$. Thus, the available (N/2-1) free conditions can be chosen to put (N/2-1) more zeroes of $H(\omega)$ at $\omega = \pi$. For an N-tap wavelet, there can be a maximum of N/2 zeroes. With this condition, and R(y) = 0, the frequency response becomes :

$$|H(e^{j\omega})|^2 = |\cos(\omega/2)|^N \sum_{k=0}^{N/2-1}\binom{N/2-1+k}{k}\sin^{2k}(\omega/2) \tag{3.63}$$

The design procedure of an N tap wavelet is then given by :

i) For a specific value of N in Eq. (3.63), find the expression of $|H(e^{j\omega})|^2$.

ii) Find the roots of $|H(e^{j\omega})|^2$ using spectral factorization.

iii) Select the roots in complex pairs such that the coefficients will be real.

Daubechies chose to put the zeroes inside the unit circle (*i.e.* minimum phase) when selecting $|H(e^{j\omega})|$ from $|H(e^{j\omega})|^2$. An alternation of zeroes inside and outside the unit circle can be chosen to get a more symmetric sequence, h[n]. However, a completely symmetric real orthogonal wavelets (*i.e.* linear phase filter) is not possible. The above steps are usually carried out in z-transform domain for convenience.

Example

As an example, we now show the design procedure of a 4-tap Daubechies Wavelet. Let N in Eq. (3.63) be 4. Then

$$|H(e^{j\omega})|^2 = |\cos(\omega/2)|^4 \sum_{k=0}^{1} \binom{1+k}{k} \sin^{2k}(\omega/2)$$

$$= 0.5 + 0.75\cos(\omega) - 0.25\cos^3(\omega)$$

Solving the above polynomial, we find three zeroes at $\omega = \pi, \pi, j1.317$ The corresponding zeroes in the z domain are at z = -1, -1 and 0.2679 or at its inverse 3.7322. The zero at 0.2679 is inside the unit circle and the zero at 3.7322 is outside the unit circle. Based on these zero selection, we can construct two wavelets :

i) $H(z) = 0.48296 + 0.83651z^{-1} + 0.22414z^{-2} - 0.12941z^{-3}$, (minimum phase)

ii) $H(z) = -0.12941 + 0.22414z^{-1} + 0.83651z^{-2} + 0.48296z^{-3}$ (maximum phase)

The refinement coefficients of Daubechies wavelets (with minimum phase) for various number of taps are given in Table D.1 (Appendix D). Corresponding scaling and wavelet functions are shown in Figs. 3.10 and 3.11, respectively. The scaling and wavelet functions of least asymmetric Daubechies wavelets (for 8, 10 and 12 taps) are shown in Fig. 3.12. The corresponding refinement coefficients are provided in Table D.2.

3.7.2 Coiflet wavelets

In section 3.6.3 we discussed the advantages of having higher vanishing moments for ψ: it led to high compression because the highpass transform coefficients of a function would be essentially zero where the function was smooth. Given that $\int \phi(t)dt = 1$, this

can never happen for the lowpass transform coefficients. Still, if $\int t^l \phi(t)dt = 0$, for l=1, 2,...L, then we can conclude that the low pass transform coefficients will also be small where the function was smooth. For this reason, R. Coifman suggested that for some applications, it might be worthwhile to construct wavelets whose scaling and wavelet functions - both have vanishing moments. Subsequently, Daubechies constructed such a wavelet family [24]. The goal is to find ϕ and ψ so that

$$\int t^l \psi(t)dt = 0\,, \qquad l = 0,\ 1,\ldots\ldots,\ L-1 \tag{3.64}$$

$$\int \phi(t)dt = 1, \quad \int t^l \phi(t)dt = 0\,, \quad l = 1,\ 2,\ \ldots\ldots,\ L-1 \tag{3.65}$$

where, L is the order of the coiflet wavelet.

In the frequency domain, these conditions can be stated as,

$$\Phi(0) = 1$$

$$\frac{d^l}{d\omega^l}\phi(0) = 0, \qquad l = 1,\ 2,\ \ldots\ldots,\ L-1$$

$$\frac{d^l}{d\omega^l}\Psi(0) = 0, \qquad l = 0,\ 1,\ldots\ldots,\ L-1$$

In terms of $H(e^{j\omega})$, these conditions can be stated as,

$$\left.\frac{d^l}{d\omega^l}H(e^{j\omega})\right|_{\omega=0} = 0, \qquad l = 1,\ 2,\ \ldots\ldots,\ L-1$$

$$\left.\frac{d^l}{d\omega^l}H(e^{j\omega})\right|_{\omega=\pi} = 0, \qquad l = 0,\ 1,\ldots\ldots,\ L-1$$

From the above two conditions, $H(e^{j\omega})$ can be written as

$$H(e^{j\omega}) = 1 + (1 - e^{-j\omega})^L Q(e^{j\omega}) \tag{3.66}$$

$$H(e^{j\omega}) = \left[\frac{1}{2}(1 + e^{-j\omega})\right]^L P(e^{j\omega}) \tag{3.67}$$

The above two equations are difficult to solve. Daubechies proposed an indirect approach [24]. For even L, she showed that $H(e^{j\omega})$ can be expressed as

$$H(e^{j\omega}) = 1 + \sin^L(\omega/2)\left[-\sum_{k=0}^{K-1}\binom{K-1+k}{k}\cos^L(\omega/2) + \cos^L(\omega/2)f(e^{j\omega})\right] \tag{3.68}$$

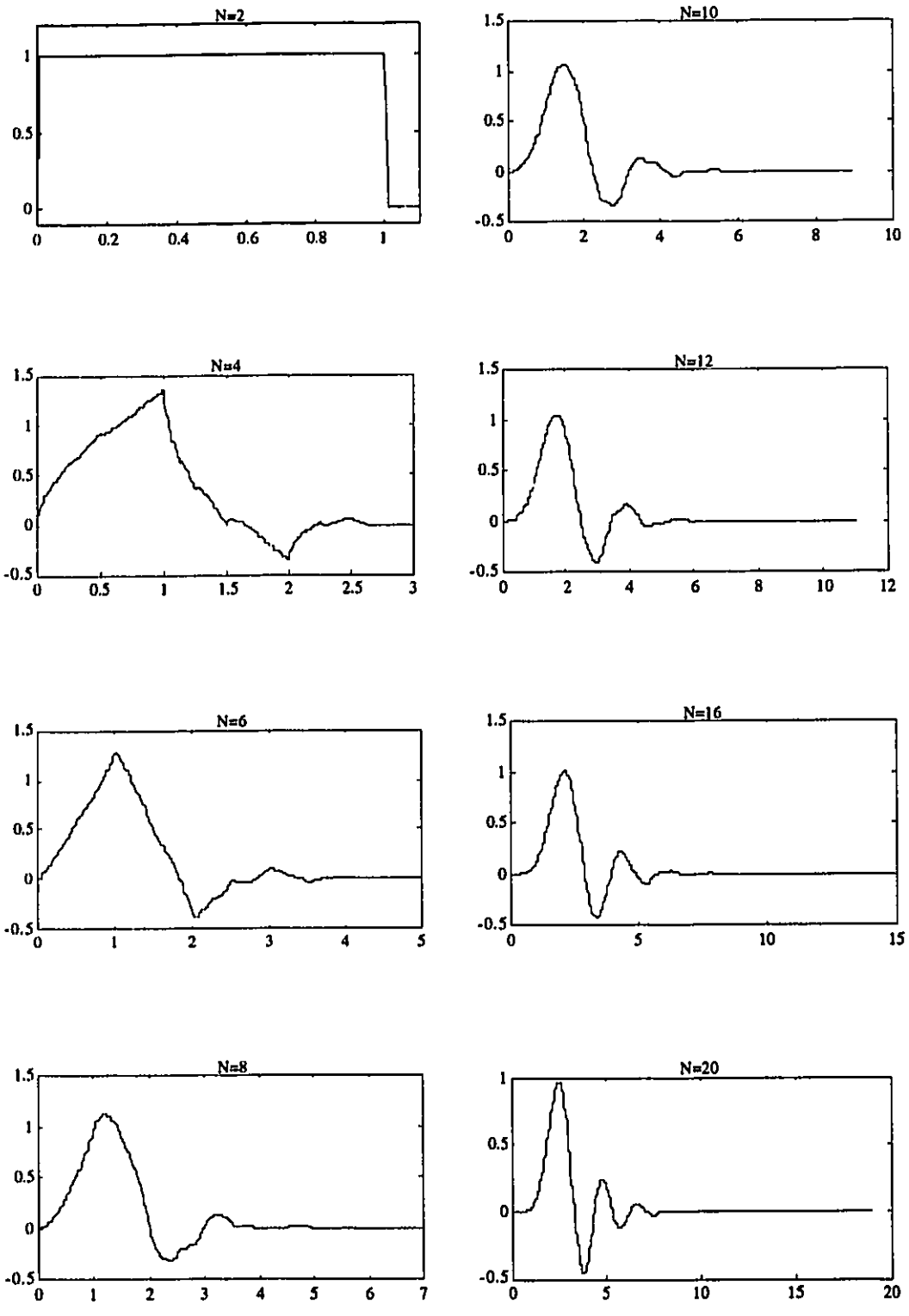


Figure 3.10 : Scaling Functions of Daubechies Wavelet
for N = 2, 4, 6, ...20.

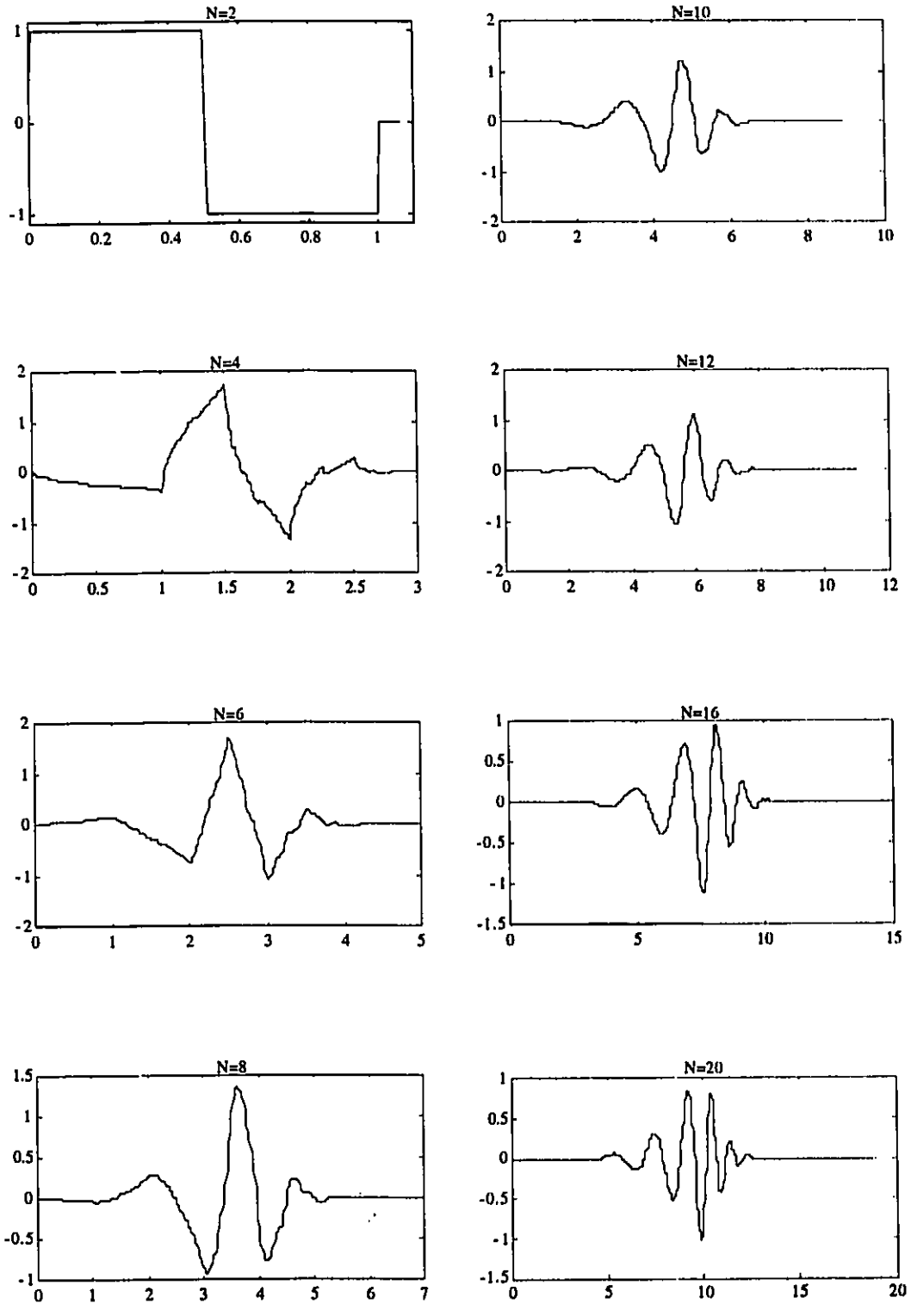


Figure 3.11 : Wavelet Functions of Daubechies Wavelet for N = 2, 4, 6, ...20.

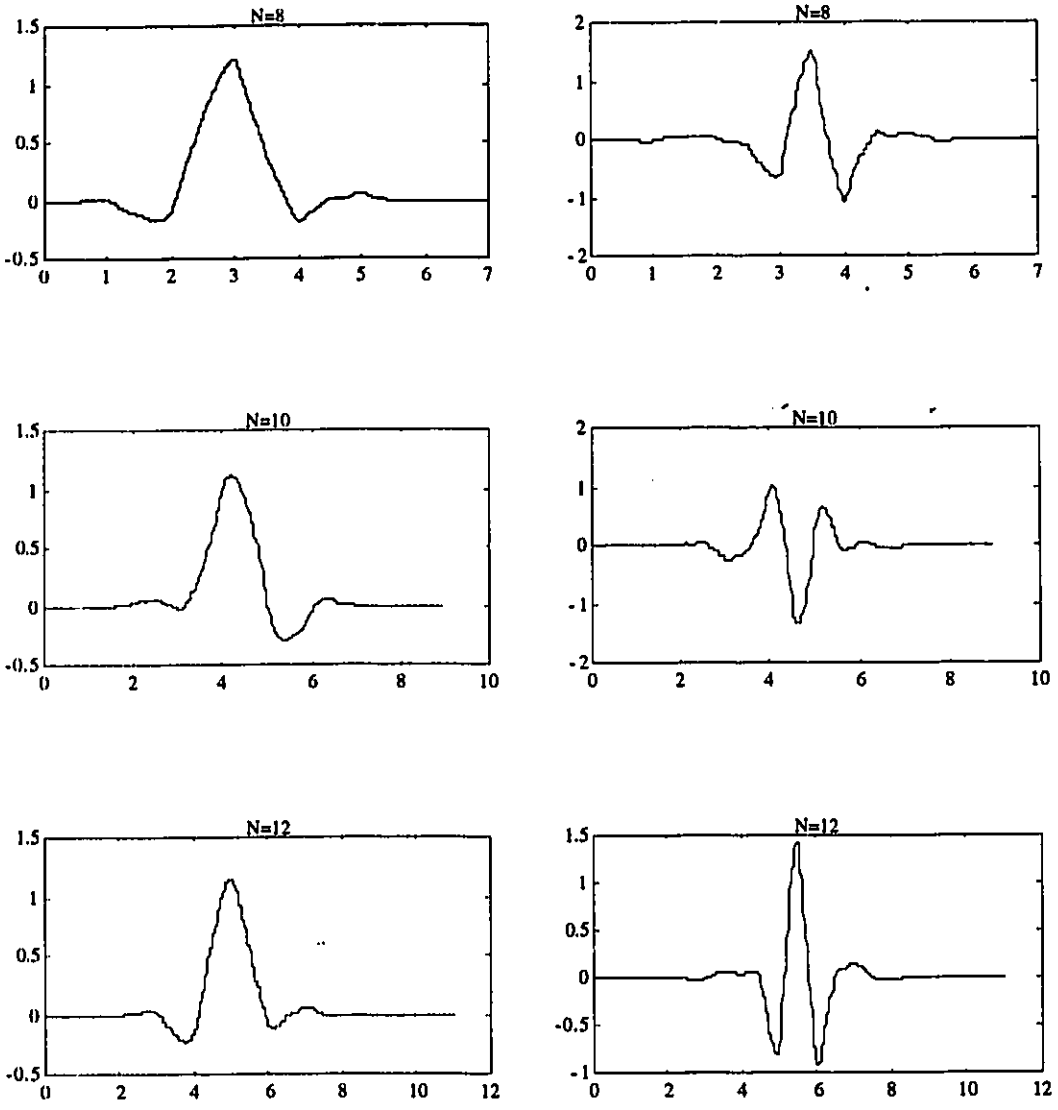


Figure 3.12 : Least Asymmetric Daubechies Wavelet for N = 8, 10 and 12
a) Scaling Function, b) Wavelet Function

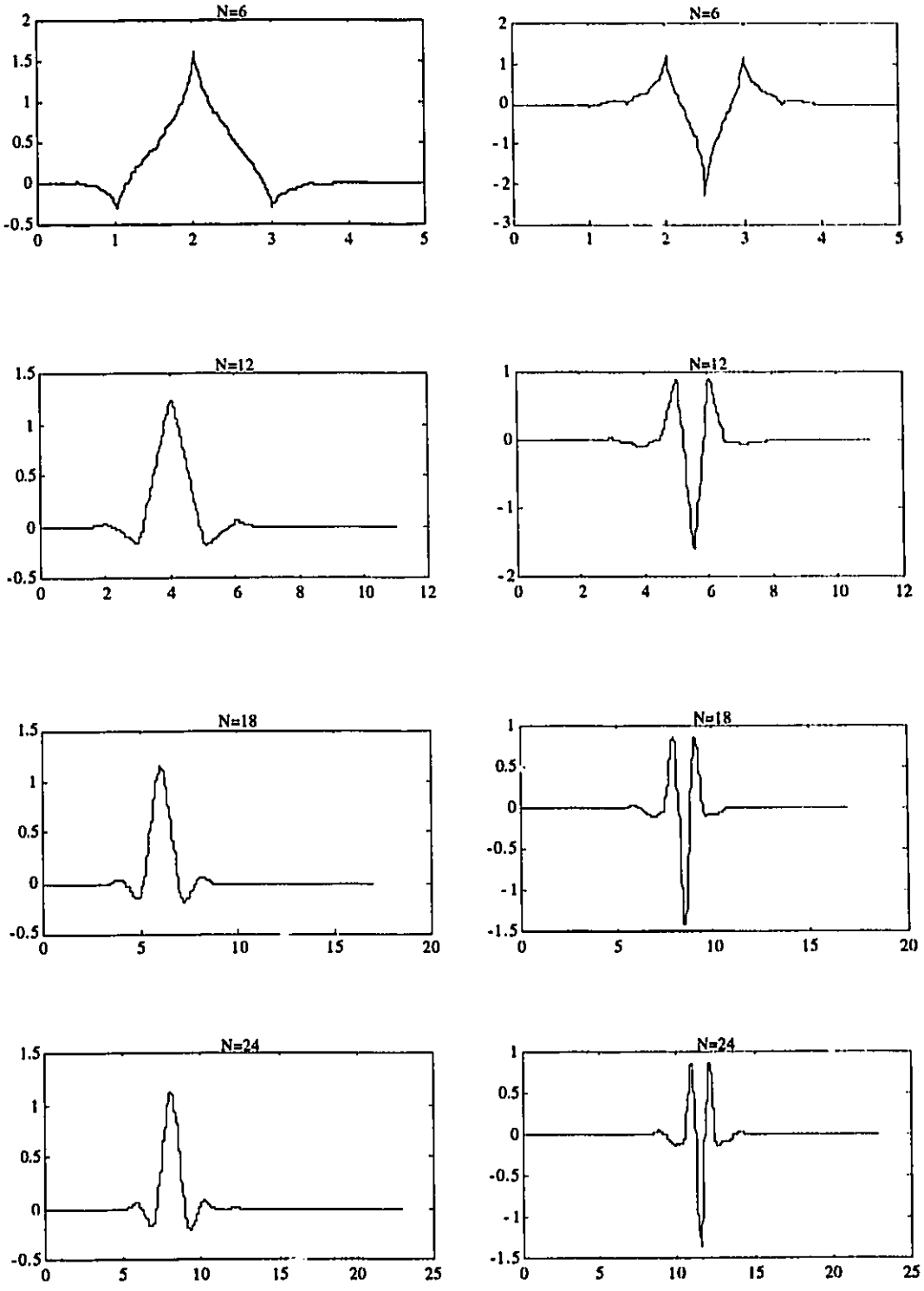


Figure 3.13 : Coiflet Wavelet for N = 6, 12, 18 and 24
a) Scaling Function, b) Wavelet Function

where $f(e^{j\omega})$ must be chosen to satisfy the perfect reconstruction requirements. Refinement coefficients of these Coiflet bases are provided in Appendix-D for N=6, 12, 18, 24 (*i.e.* for L = 2, 4, 6 and 8). The corresponding scaling and wavelet functions are provided in Fig. 3.13.

3.8 Two-Dimensional Wavelet Transform

We have developed 1-D wavelet transform from the concept of multiresolution analysis in section 3.2. In a similar way, an orthonormal basis of $L^2(R^2)$ can be constructed by using 2-D multiresolution analysis scheme. However, the construction of such wavelets are more difficult. A simpler approach is to construct a 2-D orthonormal basis by taking tensor product of two 1-D orthonormal wavelet basis. This scheme was proposed by Mallat [61]. We will briefly describe this scheme in the following [24].

Let us define spaces $\mathbf{V}_j, j \in \mathbf{Z}$, by

$$\mathbf{V}_0 = V_0 \otimes V_0 = Span\{F(x,y) = f(x)g(y); f,g \in V_0\},$$

$$F \in \mathbf{V}_j \Leftrightarrow F(2^j.,2^j.) \in \mathbf{V}_0$$

where, "$\otimes$" indicates "tensor product" of two function spaces.

Then $\mathbf{V}_j$ (note the difference between $\mathbf{V}$ and V) forms a multiresolution ladder in $L^2(R^2)$ satisfying

$$\ldots\ldots \mathbf{V}_1 \subset \mathbf{V}_0 \subset \mathbf{V}_{-1} \subset \mathbf{V}_{-2} \ldots\ldots,$$

$$\bigcap_{j\in\mathbf{Z}} \mathbf{V}_j = \{\varnothing\}, \quad \bigcup_{j\in\mathbf{Z}} \mathbf{V}_j = L^2(R^2).$$

Since, $\phi(.-n), n \in \mathbf{Z}$, constitute an orthonormal basis for V_0, the product functions

$$\Phi_{0;k_1,k_2}(x,y) = \phi(x-k_1)\phi(y-k_2), \qquad k_1,k_2 \in \mathbf{Z},$$

constitute an orthonormal basis for $\mathbf{V}_0$, generated by the $\mathbf{Z}^2$-translations of a single function Φ. Similarly, the

$$\Phi_{j;k_1,k_2}(x,y) = \phi_{j,k_1}(x)\phi_{j,k_2}(y)$$

$$= 2^{-j}\Phi(2^{-j}x-k_1, 2^{-j}y-k_2), \quad k_1,k_2 \in \mathbf{Z},$$

constitute an orthonormal basis for $\mathbf{V}_j$. As in the 1-D case, we can define $\mathbf{W}_j$ to be the orthogonal complement in $\mathbf{V}_{j-1}$ of $\mathbf{V}_j$. Thus, we have

$$\begin{aligned}\mathbf{V}_{j-1} &= V_{j-1} \otimes V_{j-1} = (V_j \oplus W_j) \otimes (V_j \oplus W_j) \\ &= V_j \otimes V_j \oplus \left[(W_j \otimes V_j) \oplus (V_j \otimes W_j) \oplus (W_j \otimes W_j)\right] \\ &= \mathbf{V}_j \oplus \mathbf{W}_j\end{aligned}$$

where, $\mathbf{W}_j = (W_j \otimes V_j) \oplus (V_j \otimes W_j) \oplus (W_j \otimes W_j)$ and "$\oplus$" indicates "orthogonal sum". We observe that, $\mathbf{W}_j$ consists of three components. Hence, we can define three wavelets as follows,

$$\Psi^h(x,y) = \phi(x)\psi(y)$$

$$\Psi^v(x,y) = \psi(x)\phi(y)$$

$$\Psi^d(x,y) = \psi(x)\psi(y)$$

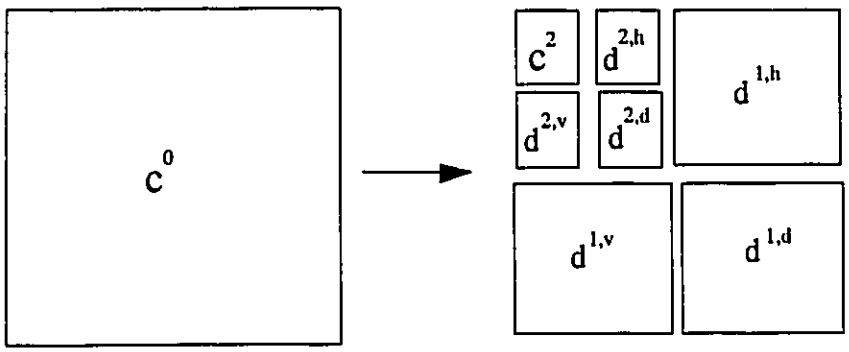
In the above expressions, h, v, d stand for *horizontal*, *vertical* and *diagonal*, respectively.

In 1-D case, we have seen that each level of decomposition produces two bands (or scales) corresponding to low and high resolution data. In case of 2-D wavelet transform, each level of decomposition will produce four bands of data, one corresponding to scaling functions and three corresponding to *horizontal*, *vertical* and *diagonal* wavelets. If the original 1-D $\phi(x)$ and $\psi(x)$ have compact support, then the corresponding 2-D scaling and wavelet functions will also have compact support. The filtering can be done on "rows" and "columns" in the two dimensional array (similar to 2-D separable DFT), corresponding to horizontal and vertical directions in images. The schematic representation of 2-D wavelet transform for two levels is shown in Fig. 3.14. The wavelet decomposition of Barbara image for two levels is shown in Fig. 3.15. The dynamic range of the highpass coefficients is usually very low and hence the highpass coefficients in Fig. 3.15, have been multiplied by a constant factor to increase the dynamic range. It is observed in Fig. 3.15 that the change in horizontal, vertical and diagonal directions are captured by the horizontal, vertical and diagonal wavelets,

respectively. For example, the legs of the table are more conspicuous in the horizontal bands.

The above separable scheme is very easy to implement. In addition, each detail image $d^{j,k}$ is sensitive towards a particular orientation *e.g.* horizontal, vertical or diagonal. As the human visual system is more sensitive to the horizontal and vertical directions in comparison to other directions [67], the above scheme (with decomposition on rectangular grid) is well suited to image coding applications.

Recently, some authors have tried to use nonseparable 2-D wavelet transform. Most popular among them is quincunx 2-D transform which is applied on signals that are sampled on a hexagonal lattice [110]. However, no thorough performance analysis has been done for various 2-D decomposition schemes and it is not known how much performance improvement can be expected from nonseparable DWT over separable DWT.



(a) Original (b) Wavelet Decomposed

Figure 3.14 : Schematic Representation of two dimensional Wavelet transform (upto two stages)

3.9 Computational Complexity

There exist several algorithms for computing DWT. In this section, we briefly mention their relative complexity for calculating dyadic DWT. Calculation of DWT coefficients,

is basically a series of convolutions. The implementation in Fig. 3.7 does calculate the DWT (for one stage) but is highly inefficient. First, we observe that in the analysis stages, the alternate output samples are discarded. Hence, it will be enough to calculate only the even (or the odd) outputs. Similarly, in the synthesis stage, half of the input samples are zeroes. Hence, the multiplication with these zero input samples

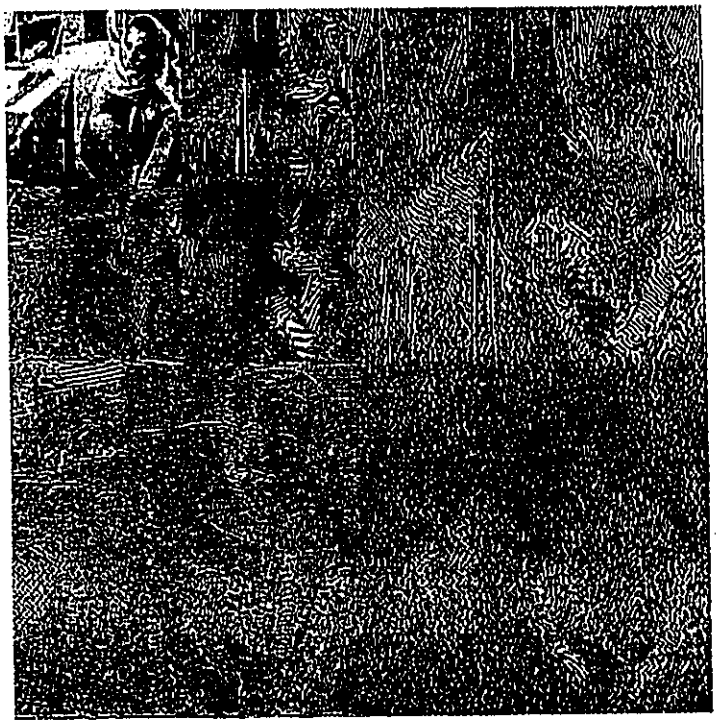


Figure 3.15 : Barbara image decomposed with Daub-8 for 2 stages

and corresponding additions are not required at all. These unnecessary computations can be easily avoided by polyphase decomposition of the input samples [99].

Let us assume that the FIR filter length is L and the number of input samples is K. With the above scheme, the computational complexity for will be as follows.

Decomposition		Multiplications	Additions
1st stage	K x	L	L-1
2nd stage	K/2 x	L	L-1
3rd stage	K/4 x	L	L-1

Thus, with J stages of decomposition, the computational complexity will be,

$$C \approx (K + \frac{K}{2} + \frac{K}{4} + ...) * 2L \text{ FLOP}$$
$$\approx 4\left(1 - 2^{-J}\right)KL \text{ FLOP}$$

where, FLOP means *floating point operations* and usually refers to floating point multiplications and additions.

With the above scheme, it is seen that the computational complexity is proportional to both the length of the input data and the filter length. The above scheme can be improved by two ways. Firstly, it is known that FFT is very efficient for doing convolution for large sequences. Hence, if the filter length is large, the complexity can be reduced appreciably using FFT, making it proportional to $\log_2 L$, instead of L. However, as will be seen later that for image coding applications, normally a short filter is preferred and FFT based method is not efficient in these cases. For short length filters, "first running FIR filtering" techniques allow one to achieve typically 30% saving in computations over the direct filterbank implementations. A comparison of direct implementation, FFT based implementations and the short length algorithm is shown in Table-3.2 [85]. The numerical entries are given only for one stage decomposition. They provide the number of floating point operations, *i.e.*, multiplications and additions per input or output samples. In Table 3.2, we see that for filter lengths less than 10, direct implementation and the short length algorithm provide competitive performance. When the filter length is between 10 and 18, short length algorithm provides superior performance. For length more than 18, FFT based algorithm provides the best performance.

2-D Dyadic DWT

Here, we will calculate the complexity of 2-D DWT for an input image of size NxN, with J levels of dyadic decomposition.

Table 3.2

Computational complexity of various DWT algorithms

Filter Length L	Direct Imple-mentation	FFT Based Algorithm	Short Length Algorithm
4	4 + 3[1]	4 + 9.33	3 + 4
6	6 + 5	4.67 + 12	4 + 6.3
8	8 + 7	5.2 + 14.1	4.5 + 8.5
10	10 + 9	5.7 + 15.3	4.8 + 14.2
12	12+ 11	6.2 + 16.7	6 + 12
16	16 + 15	6.6 + 18.2	9 + 13
18	18 + 17	6.8 + 19	8 + 17
20	20 + 21	7.1 + 19.8	7.2 + 21.4
24	24 + 23	7.3 + 20.7	12 + 18
32	32 + 31	7.9 + 22.4	18 + 22

[1]The entries are in multiplications + additions

The total computational complexity for dyadic tree, with direct implementation is,

C = Complexity of N x N images for J stages

= Complexity of $(2N+\frac{N}{2}+\frac{N}{8}+.....)$ times N point, one stage DWT

= Complexity of $\frac{8N}{3}\left(1-4^{-J}\right)$ times N point, one stage DWT

$=\frac{16N^2L}{3}\left(1-4^{-J}\right)$ FLOP

Thus, to decompose a 512 x 512 image with an 8 tap filter for 3 levels, the computational complexity will be around 11 MFLOP.

2-D Regular Decomposition

The total computational complexity for regular tree, with direct implementation is,

C = Complexity of N x N images for J stages

= Complexity of (2N+2N+.....J times) times N point, one stage DWT

= Complexity of 2JN times N point, one stage DWT

$\approx 4JLN^2$FLOP

Thus, to decompose a 512 x 512 image with an 8 tap filter for 3 levels of regular decomposition, the computational complexity will be approximately 25.16 MFLOP.

3.10 Summary

In this chapter, we reviewed the wavelet theory from the perspective of image coding application. We first discussed the usefulness of wavelets in nonstationary signal analysis. The relationship between wavelets and multiresolution analysis was then established. The details of implementation of the discrete wavelet transform were discussed. We reviewed few parameterizations of orthogonal wavelets. The regularity and vanishing moment properties of wavelets are discussed next. We provided design procedures of two wavelet families. We then extended 1-D wavelet transform to 2-D case. We conclude the chapter by comparing the computational complexity of few DWT algorithms.

Chapter 4

Wavelet Coder Design

In the previous chapters we presented a brief review of the image compression techniques and the theory and implementation of wavelets. As stated in the introduction, the final objective is to apply wavelet decomposition for image compression. In this chapter we develop a wavelet-based image coding scheme which will be used in this thesis for performance analysis of various wavelets. In section 4.2, a general strategy has been derived for quantizing wavelet coefficients for various implementations of wavelet and using different error metric. We then propose a quantization strategy for designing a minimum mean square error (MSE) coder. The relative performance of the proposed coder with other wavelet coders as well as JPEG coder are presented next. Finally, the chapter is concluded by summarizing the observations.

Wavelet-based coding techniques can be classified into two categories - i) scalar quantization [8, 9, 13, 28, 32, 83, 101, 111, 112, 113] and ii) vector quantization [7, 105]. Both approaches have their own advantages and disadvantages. It is known that the high frequency coefficients can be modelled fairly accurately with *generalized Gaussian distribution* [110]. Scalar quantizers take advantage of this fact for the design of their quantization table. On the other hand, it is known that sharp edges are characterized by frequency components of all resolutions [67]. Hence, there will be

some residual correlation between coefficients of different scales. Vector quantizers exploit the correlation among coefficients of different scales resulting in a superior coding performance.

In this thesis, we have chosen a simple scheme of uniform quantization followed by entropy coding [113]. We note that some other schemes may provide a better coding performance [105]. However, our intention is to compare the performance of various wavelets and the associated quantization strategy for adapting to HVS characteristics. We believe that the chosen scheme will be able to provide a fair comparison of different wavelets and quantization strategies.

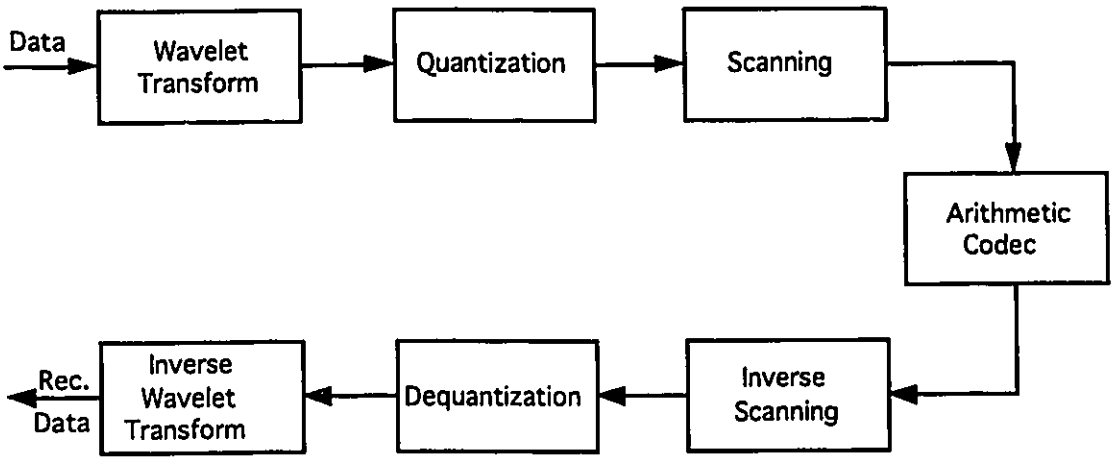


Figure 4.1 : Image Coding Scheme used in simulation

The overall coding scheme used in this thesis is shown in Fig. 4.1. The main steps of the coding scheme are as follows :

i) Forward transform (spatial --> frequency/scale domain)

ii) Bit allocation and quantization

iii) Scanning

iv) Coder and decoder

v) Inverse scanning

vi) De-quantization

vii) Inverse transform (frequency/scale --> spatial)

We will now briefly describe the simulation assumptions for steps *a*, *b*, *c* and *d*. Steps *e*, *f*, and *g* are the inverse of the steps *c*, *b* and *a*, respectively and is hence not repeated.

4.1 Transform

In chapter 2, we have discussed the two-dimensional wavelet transform. In this thesis, we have used a separable 2-D DWT, *i.e.* first we execute the row transform which is followed by the column transform. Our search for optimal wavelets has been restricted within the orthonormal compactly supported wavelets. A pure wavelet decomposition is based on a dyadic tree, *i.e.* only the lowest scale is decomposed recursively. However, we will also investigate the coding performance of regular/irregular tree. More detailed discussion will be presented in section 5.8.

The depth of the tree is, in general, determined by the length of the data and the number of FIR taps. With each decomposition, the number of rows and columns of the lowest pass band becomes halved. For efficient decomposition, the number of rows and columns of the band to be decomposed should not be less than the number of filter taps. In practice, the depth of the tree ranges from 3 to 5. There are several reasons for this. Firstly, after 3 levels of decomposition, the number of pixels in the lowest pass-band is only 1.5 % of the total. Hence, further decomposition (in the case of dyadic tree) will affect only this 1.5 % of total pixels and hence will have a little impact on the overall bitrate. Secondly, if a lowpass band contains frequency components that are equally important in subjective quality, then further decomposition may not be very efficient. Thirdly, when the image size is an odd multiple of 2^j (for example, when image size is 360x288), then one cannot decompose the image beyond j levels without taking special care to ensure perfect reconstruction. It is to be noted that when we use

3, 4 or 5 levels of decomposition, the transform is basically a truncated DWT. However, this is also generally referred as DWT in the literature.

4.2 Quantization

Once the image is decomposed into various wavelet bands, the next step is to quantize the wavelet coefficients. It is known that the bands are not equally important to the human visual system (HVS). The important bands should be transmitted more accurately and hence should be allocated additional bits. The bit allocation algorithm distributes the available bits to optimize some performance criteria. In the ideal case, the criteria should be based on HVS characteristics. Unfortunately, the adaptation to HVS is difficult to implement in practice and is still a topic of research [50, 89, 90]. Here, we have used the popular minimum mean square (MSE) criteria which is tractable analytically.

The most popular optimal MSE quantizer is the Lloyd-Max quantizer whose quantization step sizes are decided by the probability density function of the coefficients and are usually nonuniform in nature. The details of the bit-allocation algorithm for this quantizer was discussed in section 2.3.2.

It is known that for a wavelet/subband coding scheme, the high pass subbands contain a large number of zeroes. For example, the quantized high pass bands may achieve an entropy of less than 0.01 *bpp* (bits-per-pixel). A typical histogram of the quantized coefficients is shown in Fig. 4.2. This histogram has been obtained from band $d^{2,h}$ (corresponding to Fig. 3.15) of the Lena image decomposed with *Daub-8* wavelets. It has been shown that a variable length entropy coder provides a better coding performance than a fixed length coder (*e.g.* Lloyd-Max quantizer) [113]. Hence, in wavelet /DCT based coding, variable length coding (or entropy coding) is almost always used when scalar quantization is employed. It is known that with entropy coding, uniform quantization provides near optimal performance [113]. Hence, we have used a uniform quantizer followed by an entropy coding scheme.

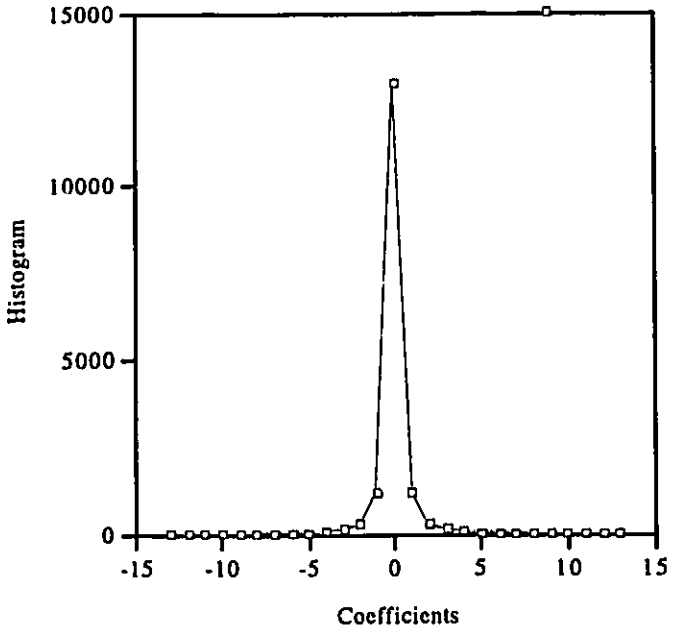


Figure 4.2 : Histogram of quantized coefficient of a high pass band

To achieve a high compression ratio, the quantization scheme should be able to handle the nonstationarity of the coefficients of various bands. To obtain the optimal performance, the complexity of the coder has to be increased significantly. For simplicity, we assume that the coefficients are stationary (spatially) within a band. Hence, a quantization step size is assigned to each band depending on some parameters, such as the variance of the coefficients of that band. Thus, if there are K bands, the quantizer design problem is to choose K quantization step sizes, each associated with a particular band, so that the overall rate is minimum with some predefined distortion constraint. A typical rate-distortion plane is shown in Fig. 4.3. Each point in the plane corresponds to a set of K quantizers. The optimization problem is to find the lower *convex hull* connecting the rate-distortions points.

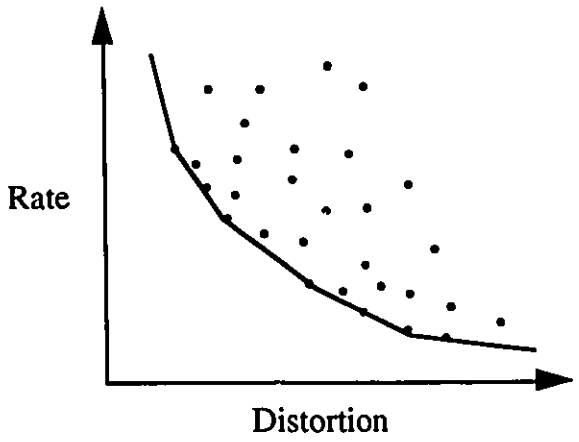


Figure 4.3 : Rate-Distortion plane, the lower convex hull represents the optimal rate-distortion combinations

In the following sections, we will describe two simple quantization schemes and a compute-intensive optimal scheme. We will provide a general scheme for calculating quantization step sizes for different implementations of wavelet transform. To calculate the quantization step-sizes, we need to know the norm of various wavelet functions as the norm provides a measure of the energy contained in a basis function.

4.2.1 Norm of the basis functions

According to Eq. (3.30) we can express a dyadic wavelet function as

$$\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}t - k)$$

The term $2^{-j/2}$ maintains the L^2 norm of $\psi_{j,k}(t)$ at various scales. Fig. 4.4 shows the Haar wavelets [38] for three consecutive scales. We see that the amplitude of the wavelet function decreases at a lower scale. This is because the support of the wavelet function is double for the successively lower scales and hence to keep the L^2-norm same in all the scales, the amplitude of the basis functions has to decrease by a factor of $1/\sqrt{2}$ in each step. The consequence of maintaining the L^2-norm is that, the peak amplitude of the wavelet coefficients increase exponentially with each level of

decomposition. To see this, recall Eq. (3.44), relating the (j+1)-th scale coefficients to the j-th scale coefficients

$$c_{j+1,k}(k) = \sum_m c_{j,m} h[m-2k] \tag{3.44}$$

Since $\sum_n h[n] = \sqrt{2}$, the dynamic range of (j+1)-th scale coefficients is almost double to that of j-th scale coefficients.

From the above relation it is easy to see that, the amplitude of low-pass transform coefficients associated with a uniform gray region will approximately be doubled ($\sqrt{2}$ times in each dimension) with each stage of 2-D decomposition. This does not pose any problem for a wavelet codec implemented with 32 bit floating point precision. However, for digital hardware implementation, it is often required to maintain the wavelet coefficients within 12-16 bit precision. This problem can be solved by scaling down the wavelet coefficients in the analysis stage. The synthesis stage will require an appropriate scaling up of the refinement coefficients to get perfect reconstruction. In other words, by scaling h[n] and g[n] in both analysis and synthesis stages appropriately, perfect reconstruction can be guaranteed. However, the norm of the associated basis function will be altered.

Thus Eq. (3.30) is modified as

$$\psi_{j,k} = a^{-j/2} 2^{-j/2} \psi(2^{-j} t - k) \tag{4.1}$$

where a is a constant introduced to scale up/down the transform coefficients. From Eq. (3.19) and (3.29), we can calculate the corresponding h[n] and g[n] for both the analysis and synthesis stages. Table 4.1 shows the relative values of h[n] and g[n] for $a = 0.5$, 1 and 2.

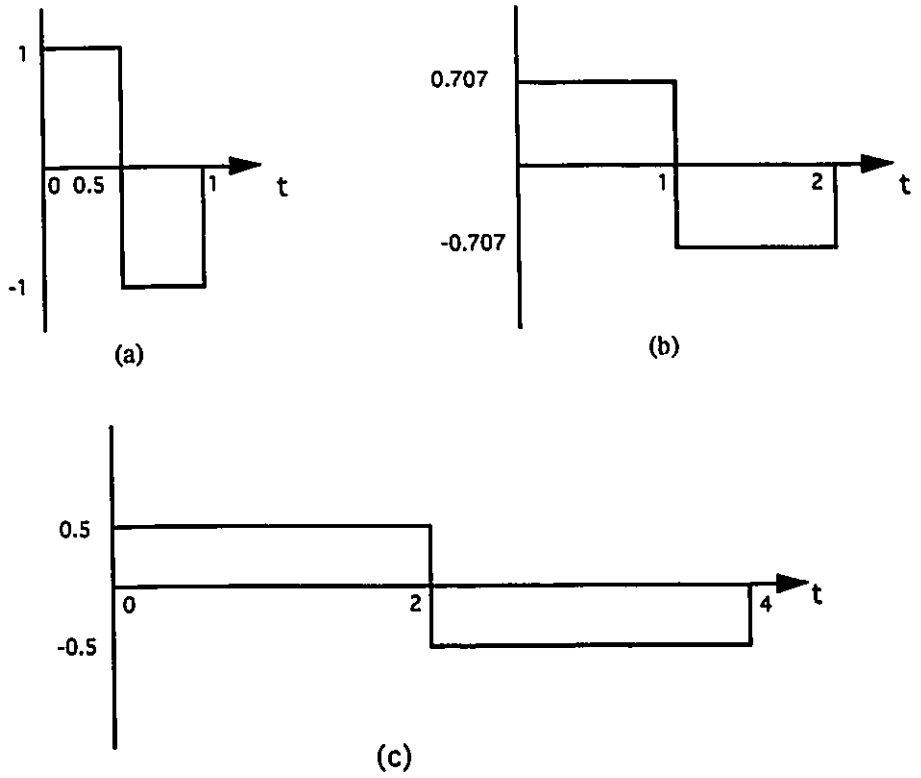


Figure 4.4 : Haar Wavelet- a) Mother Wavelet
b) One scale Coarser Resolution
c) Two scale Coarser Resolution

In Table 4.1, a=1 corresponds to a paraunitary implementation with $\sum_n h[n] = \sqrt{2}$ and the choice of a = 1/2 will limit the transform coefficients to the maximum input gray level.

Table-4.1
Scaling of refinement coefficients for various implementations

	Analysis		Synthesis	
	Low Pass Filter	High Pass Filter	Low Pass Filter	High Pass Filter
a=2	$\sqrt{2}h[n]$	$\sqrt{2}g[n]$	$\frac{1}{\sqrt{2}}h[n]$	$\frac{1}{\sqrt{2}}g[n]$
a=1	$h[n]$	$g[n]$	$h[n]$	$g[n]$
a=1/2	$\frac{1}{\sqrt{2}}h[n]$	$\frac{1}{\sqrt{2}}g[n]$	$\sqrt{2}h[n]$	$\sqrt{2}h[n]$

With the introduction of the scaling factor a, the relative norm of basis functions of consecutive scales will be changed. From Eq. (4.2), the norm of $(j+1)$-th level basis function (in 1-D case) can be calculated in L^p metric as

$$\text{Thus, } \left\|\Psi_{j+1}\right\|_{L^p} = (2a)^{-(j+1)/2}\left\{\int\left|\psi(2^{-(j+1)}t-k)\right|^p dt\right\}^{1/p}$$
$$= (2a)^{-(j+1)/2}\left\{2\int\left|\Psi(2^{-j}t'-k)\right|^p dt'\right\}^{1/p}$$
$$= 2^{\frac{1}{p}}(2a)^{-(j+1)/2}\left\{\int\left|\Psi(2^{-j}t'-k)\right|^p dt'\right\}^{1/p}$$
$$= 2^{\frac{1}{p}}(2a)^{-1/2}\left\|\psi_j\right\|_{L^p}$$

Or, in other words,

$$\frac{\left\|\Psi_{j,k}\right\|_{L^p}}{\left\|\Psi_{j+1,k}\right\|_{L^p}} = a^{-1/2}2^{(2-p)/2p} \tag{4.2}$$

From Eq. (4.2), we see that with a=1, the basis functions in all scales have the same norm in L^2-metric. However, in other metrics, the norms will be different.

4.2.2 Quantization Scheme-I

Here, we will generalize a quantization scheme used by DeVore *et al.* for a specific case of a=1/2 [27]. We assume that the reconstruction noise from every pixel is uncorrelated. Thus, given ε, a parameter controlling the compression ratio, we can choose quantized coefficients $\hat{c}_{j,k}$ which satisfy

$$\sum_{j,k}\left\|(c_{j,k}-\hat{c}_{j,k})\psi_{j,k}\right\|_{L^p} \le \varepsilon \tag{4.3}$$

In general, we try to minimize the error in L^2 metric. However, in the above equation, we have used a more general L^p, metric. In practice, this inequality is hard to satisfy. It will be much simpler, if we try to satisfy the inequality for individual pixels by ensuring Eq. (4.3) is satisfied for every element and a corresponding bound ε'. Thus,

$$\left\|(c_{j,k}-\hat{c}_{j,k})\psi_{j,k}\right\|_{L^p} \le \varepsilon' \ , \ \forall j \ \& \ \forall k$$

or,

$$\left\|(c_{j,k}-\hat{c}_{j,k})\right\|_{L^p} \le \frac{\varepsilon'}{\left\|\psi_{j,k}\right\|_{L^p}} \tag{4.4}$$

The quantization step size at the j-th scale can then be written as

$$Q_j = \frac{2\varepsilon'}{\left\|\psi_{j,k}\right\|_{L^p}}$$

In other words, the ratio of quantization step-sizes between two successive scales will be (using Eq. 6.3)

$$\frac{Q_{j+1}}{Q_j} = \frac{\left\|\psi_{j,k}\right\|_{L^p}}{\left\|\psi_{j+1,k}\right\|_{L^p}}$$

$$= a^{-1/2}2^{(2-p)/2p} \tag{4.5}$$

For 2-D wavelet decomposition with separable transform, this becomes,

$$\frac{Q_{j+1}}{Q_j} = a^{-1}2^{(2-p)/p} \tag{4.6}$$

Table-4.2 shows the ratio of quantization step-sizes of the $(j+1)$th and j-th level transform coefficients for a=2, 1, 1/2 corresponding to p = 1 and 2. Please note that a higher j corresponds to lower resolution (or scale).

Let us assume that we are using orthonormal coefficients. Hence, the step-sizes will be equal in all the bands to minimize the MSE. Given a bit-budget, we find a quantization step-size common to all the bands such that the overall desired bit-rate is achieved. However, if we want to minimize the error in L^1 metric, the step-sizes of higher resolution scales will increase by a factor of two in each successive stage. The step-sizes for wavelet implementation other than unitary, or for other error metric can be obtained from Table 4.2.

Table-4.2
Quantization step-sizes in L^p metric

	Ratio of Quantization Step Sizes		
	L^1	L^2	L^p
a=2	1	0.5	$4^{\frac{1-p}{p}}$
a=1	2	1	$2^{\frac{2-p}{p}}$
a=1/2	4	2	$4^{\frac{1}{p}}$

4.2.3 Quantization Scheme-II

The quantization scheme described above assumes that the quantization errors from various pixels are uncorrelated. This is in general not true. Here, we will develop another quantization scheme. It is known that in case of unitary transform, an optimum nonuniform quantizer distributes the noise equally across all the transform coefficients (section 2.3.2). Though, this was observed for a nonuniform Lloyd-Max quantizer (and not with a uniform quantizer followed by entropy coding), we have used this observation to quantize the wavelet coefficients. Given a distortion budget, we first calculate the total amount of noise to be introduced. For example, if we want the reconstructed image to have a PSNR of say X dB, if the maximum gray level is 255 and there are 512 x 512 pixels, the total amount of noise should be

$$N = \frac{255^2 * 512 * 512}{10^{X/10}}$$

This total noise power is distributed across the bands such that MSE in all the bands are equal. As the highest resolution bands usually have less energy, it is most likely that at a high compression ratio they will be completely discarded. Hence, in most cases, a second time redistribution of noise is required. We then find a quantization step size for each band (except the bands which will be totally discarded), which will provide the allocated average distortion. The quantization stepsize of a band is calculated in an iterative manner as follows.

$$\Delta_{k+1} = \sqrt{\frac{\varepsilon}{\sigma_k^2}}\Delta_k \tag{4.7}$$

where,

ε = The allocated MSE

σ_k^2 = MSE achieved at k-th iteration

Δ_k = The step size at k-th iteration.

Since, the rate-distortion function is convex in nature, the above iterative algorithm works well in practice. However, due to finite precision arithmetic, there is a possibility of a deadlock condition, when $\sigma_k^2\sigma_{k+1}^2 = \varepsilon^2$ and in that case, the quantization step size will shuttle between Δ_k and Δ_{k+1} and will not converge. This was observed in our simulations, though rarely. To avoid this situation, we put a maximum limit on the number of iterations. This limit also helps to exit the iterative loop when a satisfactory step-size has been obtained.

As with every iterative algorithm, the faster convergence to the desired step-sizes requires a good initial guess. If the coefficients of a band are uniformly distributed and the step-size is small, then the MSE is approximately equal to $\Delta^2 / 12$. Hence, as an initial guess, we can assume $\Delta_0 = \sqrt{12\varepsilon}$. This initial guess is almost accurate for the lowest pass-band in which case the coefficients are almost uniformly distributed. However, it is well known that the coefficients of higher pass bands have generalized Gaussian distribution and in these cases, the above mentioned initial guess will be highly inaccurate. Therefore, we modify the initial guess as, $\Delta_0 = M\sqrt{12\varepsilon}$. Table 4.3 shows the actual values of M for different bands of Lena image, decomposed in 3 stages with Daub-8 wavelet. We observe that for higher pass bands, the values of M is much higher than that of lower pass bands. Using this observation, a faster convergence of step-sizes can be achieved.

Table-4.3
Ratio of quantization step-sizes in various bands

Bands	c^3	$d^{3,h}$	$d^{3,v}$	$d^{3,d}$	$d^{2,h}$	$d^{2,v}$	$d^{2,d}$	$d^{1,h}$	$d^{1,v}$	$d^{1,d}$
M	1.01	1.07	1.18	1.3	1.58	2.13	3.77	7.78	12.9	18.25

4.2.4 An Optimal Approach

In this section, we describe a procedure for obtaining quantization step-sizes, optimal in the rate-distortion sense. Since, we are working with orthonormal wavelets, the distortions (*i.e.* the noise variances) in different bands are additive in nature. The different quantized bands are encoded separately and hence the bit-rates are also additive. The optimization problem is to find the minimum distortion D for a given bit-budget R. This will also provide us the quantization step-sizes which will produce the minimum distortion D. This constrained optimization problem can be easily converted to an unconstrained problem by relating the rate and distortion function through the Lagrange multiplier. Thus, the problem becomes to minimize the Lagrange cost function defined as [80]

$$J(\lambda) = D + \lambda R \tag{4.8}$$

Differentiating the above equation, we obtain $\lambda_{opt} = -\frac{\partial D}{\partial R}$. Each band has its own rate-distortion function, which depends on the quantization, scanning and coding schemes. It was shown [80] that the overall rate-distortion function will be optimal if all the individual bands operate at a constant slope λ on their respective rate-distortion functions. Hence, the overall rate-distortion (R-D) function can be obtained as follows.

1. Obtain the R-D function of the individual bands.
2. Choose a λ in the range $(0, -\infty)$.
3. Find the rate and distortion from the R-D function at that λ in all the bands.
4. The summation of all the rates and distortion will correspond to a point in the

overall R-D function with slope λ.

5. Change the λ and repeat step 3 and step 4.

The above procedure provides an algorithm to obtain the overall R-D function. However, it does not provide the λ which will produce the desired bit-rate. The desired λ can be obtained by an iterative algorithm described in [80].

The optimal approach does not assume anything about the coder or the quantizer. Hence, it can be used with any coding scheme. However, it does assume the convexity of the R-D functions. In our coding scheme, we have used an entropy coder and found that the R-D function is not strictly monotonic. In addition, the accuracy of the estimation of λ is very crucial. The distortion is a non-linear function of bit-rate and hence we should take only a few R-D points (so that the function is approximately linear in that range) to calculate the slope. However, the slope, calculated from a few R-D points, has seen to be somewhat random in nature. This randomness of slope is more visible in the higher resolution bands.

Comparison of the three approaches

Fig. 4.6 compares the performance of the three coding schemes described above. In the following, we will refer to the schemes of section 4.2.2, 4.2.3 and 4.2.4 as Scheme-1, Scheme-2 and Scheme-3, respectively. In Fig. 4.6, we observe that Scheme-3 provides the best PSNR at any rate. Scheme-2 provides a better coding performance than Scheme-1. This is expected because in the process of finding quantization step-sizes, Scheme-2 utilizes the difference of the *pdf* of the coefficients of various bands. Table-4.4 shows the quantization step-sizes of all the three schemes for the Lena image coded at 0.39 bits-per-pixel using Daub-8 wavelet. The image was decomposed into 3 stages.The PSNR achieved were 31.98, 32.15 and 32.3 dB for Scheme-1, Scheme-2 and Scheme-3, respectively. It is seen that Scheme-2 provides quantization step-sizes closer to those obtained from Scheme-3, which is optimal in

the mean square sense. However, it is evident from Fig. 4.6 that only a marginal improvement (of 0.2-0.3 dB) can be expected from optimizing the quantization step-sizes. Hence, in chapter 5, for comparing various wavelet bases, we have used Scheme-2 as it provides a performance closer to the optimal algorithm.

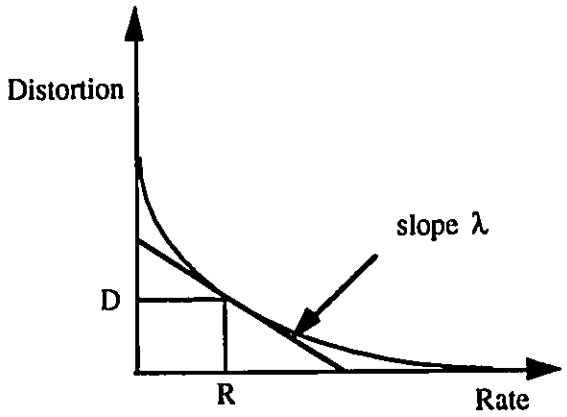


Figure 4.5 : Lagrangian cost function λ

4.3 Arithmetic Coding and Scanning

In our coding scheme, we quantize each band with its corresponding quantization step size and then encode the coefficients using arithmetic coding. The details of arithmetic coding have been discussed in section 2.2.2. Since, the statistics of the bands vary widely, we have preferred to code each band separately. This has been seen to provide a good performance. The remaining nonstationarity within a band is easily handled by an adaptive model.

Adaptive coding is by its nature a memory process. Therefore the order in which the coefficients are fed into the coder is an important issue. The higher the local stationarity of coefficients, the better the adaptation. In that context, the scan of the image will influence the performance of the coder. There are various types of scanning which are in use, *e.g.* horizontal, zigzag, Peano-Hilbert scanning, *etc.* In our coding scheme, we have used the horizontal scanning for its simplicity.

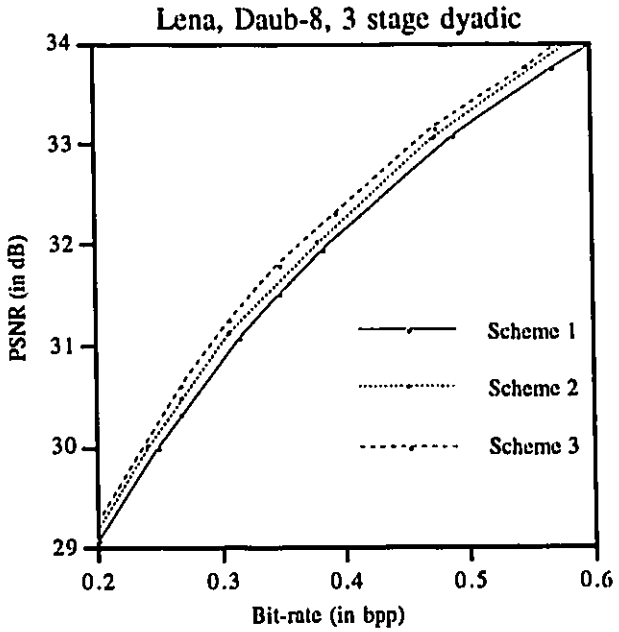


Figure 4.6 : Comparison of three quantization schemes

Table-4.4
Quantization step-sizes of Lena image
coded at 0.4 bpp

Bands[1]	Scheme-1[2]	Scheme-2	Scheme-3
c^3	44.0	25.1	29.0
$d^{3,h}$	44.0	26.1	31.0
$d^{3,v}$	44.0	28.3	35.0
$d^{3,d}$	44.0	29.5	37.0
$d^{2,h}$	44.0	32.2	37.0
$d^{2,v}$	44.0	39.1	41.0
$d^{2,d}$	44.0	43.0	48.0
$d^{1,h}$	44.0	64.9	47.0
$d^{1,v}$	44.0	*discarded*	77.0
$d^{1,d}$	44.0	*discarded*	*discarded*
Bit-rate	0.4 *bpp*	0.4 *bpp*	0.4 *bpp*
PSNR	31.98 dB	32.15 dB	32.3 dB

[1]The entries corresponds to the bands defined in Fig. 3.15
[2]The numerical entries are quantization step-sizes

4.4 Performance

The coding performance of our scheme on the *Lena image* is shown in Fig. 4.7. We have also provided the coding performance of the JPEG coder for comparison. We have used the JPEG coder which is available in Version 2.21 of John Bradley's "xview" program (available on Unix platform). The image was decomposed with Daub-8 wavelet for 3 dyadic levels. It is seen that wavelet coder provides a performance better than the JPEG coder. Further improvements in performance can be obtained with additional decomposition or, by using optimal wavelets. It should be noted that the DCT-based coding using JPEG standard has a substantially lower computational complexity than that of DWT with 8 taps (about one-third). This is because of the fact that the JPEG coder uses a highly optimized DCT algorithm (with some approximation). We believe that future research will also reduce the complexity of DWT algorithm.

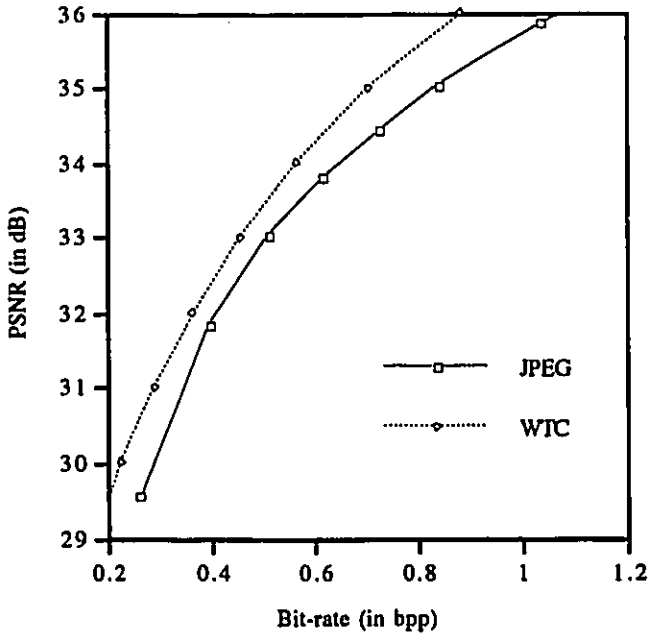


Figure 4.7 : Comparison of JPEG and WTC

4.5 Summary

In this chapter, we analyzed various wavelet based coding schemes. We provided a general quantization strategy for various implementations of wavelets and using different error metric. We proposed a coding scheme based on distributing the quantization noise equally in all the bands. It was observed that our suggested coding scheme (*i.e.* Scheme-2) provides a performance very close to that provided by Scheme-3 (which optimal in the MSE sense) at a considerably lower complexity. Hence, we have chosen Scheme-2 for the analysis to be reported in chapter 5.

Chapter 5

Choice of Wavelets for Image compression

In this chapter, we compare several wavelets based on various performance criteria. In section 5.1, we present a brief overview of the ongoing research being done in the area of wavelet-based image compression. In section 5.2, we list the wavelets and test images which have been used for the analysis. We restrict ourselves to the case of compactly supported orthonormal wavelets. First, we compare the relative performance of three well known wavelet families. In section 5.3, we provide various useful filter parameters of the wavelet families, which are known to affect the coding performance. In section 5.4, we study the usefulness of the transform coding gain as a criteria for evaluating the coding performance of different wavelets. The comparative study of the objective and subjective coding performance of these wavelet families are provided in section 5.5 and section 5.6, respectively. We also studied the effect of filter order on coding performance. In section 5.7, we find the optimal 4, 6 and 8 tap wavelet for a set of test images using the wavelet paramaterization discussed in section 3.5. An analysis of performance improvement due to the use of optimal wavelet over the three wavelet families is then discussed. In section 5.8, we propose a simple irregular tree decomposition technique and compared its performance with that of dyadic/regular tree. The performance of complex symmetrical orthonormal wavelets on real images is detailed in section 5.9. Finally, the conclusions are presented in section 5.10.

5.1. Introduction

In wavelet domain, the image is being represented by a set of basis functions, weighted by the transform coefficient values. If the basis functions are different, the image representation will also be different and the coding performance will change. Since there are numerous wavelet bases available, each with a different set of basis functions, the choice of wavelet is very crucial to achieve a good coding performance. Unfortunately, there is no single wavelet which will always provide the best performance. Given a wavelet, it should always be possible to construct an image for which the given wavelet is the optimum. Hence, some kind of search (either iterative or a full search) or design procedure may be used to obtain the optimal/sub-optimal wavelet for a given image [12, 19, 36, 46, 60, 71, 72, 94, 96, 97, 98].

Before searching for the optimum wavelet, we should establish some performance criterion. The ultimate performance criteria is the *bit-rate* versus *subjective quality* of the reconstructed image. However, the computational complexity to evaluate this criteria is very high since one has to follow the complete coding/decoding scheme. In addition, verifying the subjective quality of the reconstructed image is laborious in practice. Instead, if we can find some performance measure in the transform domain (for example, transform coding gain), the computational complexity will be reduced considerably since the inverse transform need not be calculated.

A widely used performance criteria is based on the energy compaction properties of the wavelets. There are some variants of this criteria. Most researchers have used *transform coding gain* (as defined in section 2.3.2) as the performance measure [26]. Some authors have used the criteria of maximizing the energy compaction in the first L bands [97]. If one chooses L=1, the criteria becomes that of maximizing energy in the lowest pass-band. If we assume that the quantization noise is a wide sense stationary ergodic random process and is uncorrelated with the transform coefficients, all these criteria become approximately equivalent [96].

Another well known criteria is based on optimizing the filter characteristics [3, 4, 5, 52, 53]. An objective function is defined as a function of various filter characteristics which are known to affect the coding performance. These characteristics are given relative weights. If the statistical properties of the image are known, a wavelet can be found to optimize the overall objective function. However, to use these criteria, a good knowledge of the statistical properties of the input image is required. In addition, the relative weights of various filters properties are not accurately known. More analysis has to be done to achieve a good coding performance using this criteria.

Tewfik *et al.* [97], have proposed an iterative technique to find the best wavelet basis that minimize the upper bound on the L^2-norm of the signal approximation error up to a given scale. The algorithm is complex. In addition, the solution may converge to a local optimum. The local convergence problem can be solved by using a multistart algorithm [96] which will produce all local optimum wavelets. From these locally optimum wavelets, the globally optimum wavelet can be easily determined.

The implementations described above try to find a wavelet which is optimum over the whole image. As the majority of the images are non-stationary in nature, a scene adapted filterbank structure will be able to provide a better coding performance. However, designing a perfect reconstruction (PR) time-varying filterbank (more truly, space varying) is still a subject of research. The problem arises in the transition states when the filterbank is changed. Changing the analysis and synthesis filterbanks does not automatically guarantee PR, even if the individual filterbanks have PR property. Nayebi *et al.*, have proposed a preliminary design technique for PR time varying FIR filterbank structure [69]. They used a set of intermediate synthesis filterbanks for reconstructing the data in the transition states, depending on the initial conditions. However, no performance analysis has been done to verify the effect of changing the filterbanks.

Desarte *et al.* [69], have proposed an algorithm for changing the wavelet in each resolution level and in each direction to maximize the transform gain. In this approach, PR is easily achieved as we can treat each coarse resolution image as a new image to be decomposed. As long as the filters in each resolution level are unitary, Parseval's relation will hold and an MSE coder can be designed easily.

Husoy *et al.* [46], have reported that better coding performance can be achieved by using time varying IIR filterbanks. The whole image is divided into several regions depending on the activity (low/high). A short impulse response (IR) filter should be used in the active regions (since, longer filters produce ringing artifacts in active regions) and a long impulse response filter should be used in the smooth regions (since, shorter filters produce blocking artifacts in smooth regions).

The performance in wavelet-based coding also depends on the structure of the tree decomposition. Ramachandran *et al.*, have proposed a technique, based on Lagrangian optimization, to find the best basis subtree, which minimizes the global distortion for a given bit budget [80]. However, the algorithm is computationally complex and a simple tree decomposition scheme is highly desirable.

The effect of regularity, frequency selectivity and phase on still image compression has been reported by Olivier [48]. It has been observed that regularity is relevant for short length filters and not important for longer length filters.

The computational complexity to find an optimal wavelet is usually very high. In section 3.9, we have seen that a 3 stage wavelet transform of a 512 x 512 image requires about 11 MFLOP with an 8 tap filter, which is approximately 3 times more than that required for 8x8 block DCT for same image size. Usually, the wavelet is searched in a systematic way using the parameterization described in section 3.5. An 8-tap wavelet has 3 free parameters, where each parameter can take a real value anywhere in $[0,\pi]$, or $[0,2\pi]$, depending on the type of parameterization. If the optimal wavelet is searched on a 3-dimensional grid with a spacing of $2\pi/32$ in each

dimension, a total 32^3 combinations are possible. However, in practice, one need not try all the possible combinations. It has been observed that for most natural images the optimum wavelets are clustered in some specific regions of the parameter space. Hence, we can limit the search in these regions. However, even if the optimal wavelet is searched from a set of hundred, the computational complexity is enormous. It is of interest to know if there is a wavelet which provides close performance to the optimal wavelet for most natural images. If such a wavelet is available, then the computationally intensive searching algorithms can be avoided in most cases. When the requirement is stringent and we have to employ an iterative searching method for finding the optimal wavelet, it will be of interest to know a good initial guess as the convergence of an iterative algorithm strongly depends on the initial guess.

In the following sections, we provide our analysis to address some of these questions. We relate the choice of wavelet filter and tree decomposition structure to the image statistics. Depending on the image statistics, we provide some guidelines regarding when to expect an appreciable improvement in performance resulting from usage of optimum wavelet. We provide a thorough performance analysis of three popular wavelet families. We then search for 4, 6 and 8 tap optimal wavelets for a wide variety of input image. We provide the performance improvement achieved by using the optimal wavelet over these known wavelets and relate this with the the statistics of the image.

5.2 Wavelets and Images Used

In this section we list all the wavelets and images used in the experiments reported in this chapter. We restricted ourselves to the orthonormal wavelets as orthonormality constraint provides an easy solution to error analysis. The following wavelets were used in our analysis.

1. *Daubechies* - 2, 4, 6, 8, 10, 12, 16 and 20 taps (minimum phase)

2. Least asymmetric- 8, 10 and 12 taps *Daubechies* wavelets

3. *Coiflet* - 6, 12, 18, 24 taps

4. *AHQMF* - length 4, 6, 12, 18 [Taken from Ref 4, Table-4.11]

5. A set of 4 tap filters calculated using Pollen's Parameterizations [77]

6. A set of 6 tap filters calculated using Pollen's Parameterizations [77]

7. A set of 8 tap filters calculated using Zou's Parameterizations [114]

All of the above wavelets have been discussed in section 3.7 except AHQMF wavelets. These wavelets were designed by Akansu *et al.* [4], by optimizing an objective function consisting of aliasing energy, step response, phase linearity, *etc.*

The images used in the experiments are mostly CCITT test images with the exception of three synthetic aperture radar (SAR) images.

CCITT images (all of them are of size 512 x 512) :

Airport	: An aerial image of an airport.
Airplane	: Image of an F-16 fighter plane.
Chest	: X-ray image of a human chest, containing occasional fine textures.
Girl	: A face image
Lena	: A face and shoulder image
Mandrill	: Image of face of a baboon, containing many fine textures
Sailboat	: Image of a natural scene including boat, river *etc.*
Visualmtf	: An image for finding modulation transfer function of human visual system.

SAR images :

Death Valley	: A 480x768 SAR image of Death valley, California
Galapagos Island	: A 480x704 SAR image of Western Galapagos Island
Safsaf Oasis	: A 512x512 SAR image of Safsaf oasis, Southern Egypt

The above images were selected based on their overall activity. A good measure of the overall activity of an image is the *spectral flatness*. The spectral flatness of a digital image is defined as the ratio of the arithmetic and the geometric mean of the Fourier coefficients [49]. For two-dimensional digital image, we use the following formula to calculate the spectral flatness measure (SFM).

$$SFM = \frac{\frac{1}{MxN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}|F(i,j)|^2}{\left[\prod_{i=0}^{M-1}\prod_{j=0}^{N-1}|F(i,j)|^2\right]^{\frac{1}{MxN}}}$$

where $F(i,j)$ is the (i, j)-th Fourier coefficient of the two-dimensional image.

Table-5.1 shows the spectral flatness measure of various images. The Table also shows the horizontal and vertical inter-pixel correlation of various images. The correlation coefficients were obtained by dividing the image into several 16x16 blocks and then averaging the interpixel correlation of these blocks. We note that, the two most active CCITT images are the Airport and Mandrill whereas, the two least active images are the chest and visualmtf image. SAR images are in general less correlated which is also observed from Table-5.1.

Table-5.1
Statistics of the images used

Image	S.F.M[1]	Hor. Corr	Ver. Corr
Airport	0.3714	0.81	0.81
Mandrill	0.1035	0.92	0.91
Girl	0.0119	0.93	0.93
Sailboat	0.0107	0.92	0.92
Lena	0.0106	0.92	0.93
Airplane	0.0080	0.93	0.93
Visualmt	0.0055	0.92	0.94
Chest	0.0005	0.93	0.94
Safsaf	0.1213	0.895	0.88
Galapagos	0.1095	0.91	0.80
Death valley	0.0367	0.92	0.85

[1]Spectral flatness measure

5.3. Wavelet Filter Characteristics

The subband coding system is based on the frequency selectivity property of the filterbanks. An alias free frequency split and perfect interband decorrelation of coefficients can be achieved only with ideal filterbanks with infinite duration basis functions. However, it is known that the time localization of the filters is very important in visual signal processing. In these sections, we will discuss a number of properties which are known to affect the coding performance. We will compare these properties for various wavelets. Specifically, it will be seen that the joint time-frequency localization of a wavelet is a good indicator of the coding performance. The coding performance of a filter depends on the following properties [4].

a) Regularity

b) Vanishing Moments

c) Phase Linearity

d) Step Response

e) Time-frequency window

f) Energy compaction

g) Aliasing Energy

h) Uncorrelated Subband Signals

The properties *a, b, c, d, e* and *f* depend on the filter. Properties *g* and *h* depend on the filter as well as the spectral content of the input image. In this section, we will calculate the parameters for the three wavelet families mentioned in section 5.2. A combination of various properties of the wavelet makes one wavelet superior to others and in general it is very difficult to single out a specific property which results in a good performance. The filter parameters can be derived either by using the refinement coefficients or by using the wavelet function. The wavelet function is the continuous time basis function of a wavelet and seems to be more appropriate. However, it is

achieved only after infinite iterations on the refinement coefficients. We now compare the properties of refinement coefficients. The wavelet function is used only to compare the time-frequency localization.

Regularity :

The regularity of a wavelet is difficult to estimate. Among the wavelets used in this thesis, only the regularity of Daubechies wavelets is available. The regularity of L tap Daubechies wavelet is given by [24]

Table-5.2

Regularity of Daubechies wavelets

N[1]	Regularity
4	0.5500
6	1.0878
8	1.6179
L	$\approx 0.5(L/2-1)$

[1] N is the filter order

The most regular wavelets MReg-4 and MReg-6 have a regularity of 0.73 and 1.41, respectively. The other wavelets used in this thesis have a lower regularity than Daubechies wavelet for a given support length.

Vanishing Moments

In section 3.7, we discussed the vanishing moment properties of Daubechies and Coiflet wavelets. It was observed that N tap Daubechies wavelet has N/2 vanishing moments whereas an N tap Coiflet wavelet has N/3 vanishing moments.

Step Response

The representation of edges in an image is very important to provide a good visual fidelity. The edge structures are localized in time and hence are well represented by a time localized basis function. For long FIR filters, ringing artifacts have been seen near edges [46]. An edge can be approximated as a step function and hence the step response of the low pass filter is important to achieve a good subjective quality.

The unit step response of the filter h[n] can be written as

$$a[n] = h[n] * u[n] = \sum_{i=0}^{n} h[i]$$, where u[n] is the unit step sequence.

The difference of energy between the unit step response a[n] of the filter and the unit step sequence u[n] is expressed as

$$E_s = \sum_{k=0}^{2N-1} \left[\sum_{n=0}^{k} h[n] - 1 \right]^2 \tag{5.1}$$

The value of E_s should be minimized for an optimal filter. In the above equation, $h[n]$ was normalized, so that $\sum_n h[n] = 1$. Table-5.3 shows the step response of various wavelet filters. We note that the step response error increases with filter length. This is an expected result as the higher order filters are known to produce more ringing noise near the edges. Also, AHQMF filters have appreciably lower step response error than Daubechies or Coiflet wavelets.

Nonlinearity in Phase Response

It is known that phase nonlinearity of filter can cause severe degradations in the subjective quality of the image [74]. In general, the phase linearity of an FIR filter is related to the symmetry of the filter coefficients. It is known that a perfectly symmetrical orthonormal filter is not possible. Therefore, a measure which indicates the relative asymmetry can be defined as [4]

$$E_p = \sum_n [h(n) - h(2N-1-n)]^2 \tag{5.2}$$

Table-5.3 shows the asymmetry of the various filters. It is seen that the least asymmetric Daubechies wavelet has an appreciably lower non-linearity than its minimum phase counterpart.

Table-5.3
Step response error and non-linearity of various wavelets

Wavelet	Step response error	Non-linearity	Wavelet	Step response error	Non-linearity
Haar	0.25	0.00	AHQMF-4	0.46	0.72
Daub- 4	0.45	0.75	AHQMF-6	0.51	1.20
Daub- 6	0.64	1.24	AHQMF-12	0.64	0.94
Daub- 8	0.83	0.91	AHQMF-16	0.66	0.99
Daub-10	1.02	0.97			
Daub-12	1.22	1.05	Coiflet-6	1.81	0.39
Daub-16	1.59	1.02	Coiflet-12	3.83	1.14
Daub-20	1.97	0.99	Coiflet-18	5.83	0.94
			Coiflet-24	7.83	1.03
Daub-8(LA)	2.82	0.62			
Daub-10(LA)	3.72	0.06			
Daub-12(LA)	4.93	0.55			

Time-Frequency Window :

Both time and frequency domain characteristics of a filter are important in visual signal processing. It is desirable to use a wavelet with a smaller time-frequency localization. The time localization of the finite sequence h[n] is defined as [4]

$$\sigma_n^2 = \frac{1}{E}\sum_n (n-\bar{n})^2 |h[n]|^2 \tag{5.3}$$

where,

$\bar{n} = \frac{1}{E}\sum_n n|h(n)|^2$, the centre of mass of the sequence.

and

$E = \sum_n |h[n]|^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} |H(\omega)|^2 d\omega$, is the total energy of the sequence.

Similarly, the frequency localization of h[n] is defined as

$$\sigma_\omega^2 = \frac{1}{E}\frac{1}{2\pi}\int_{-\pi}^{\pi} (\omega-\varpi)^2 |H(\omega)|^2 d\omega \tag{5.4}$$

where,

$$\varpi = \frac{1}{2\pi E}\int_{-\pi}^{\pi} \omega |H(e^{j\omega})|^2 d\omega$$

In the above equations, we have assumed that, $h[n] \leftrightarrow H(e^{j\omega})$.

It is known that for a continuous time signal, the time frequency localization is lower bounded by 0.5 [75], *i.e.*,

$$\sigma_\tau \sigma_\omega \geq \frac{1}{2}$$

For a discrete time signal, this lower bound is given by [41]

$$\sigma_n \sigma_\omega \geq \frac{|1-\mu|}{2}, \quad \text{where } \mu = \frac{|H(-1)|^2}{E}$$

Table-5.4
Time-frequency localization of scaling functions

Scaling Fn	σ_n^2	σ_w^2	$\sigma_n^2\sigma_w^2$	$\sigma_n\sigma_w$
Haar	0.081	132.2	10.71	3.273
Daub- 4	0.098	14.37	1.407	1.186
Daub- 6	0.132	5.200	0.688	0.829
Daub- 8	0.173	4.169	0.721	0.849
Daub-10	0.219	3.842	0.842	0.918
Daub-12	0.271	3.693	1.000	1.000
Daub-16	0.389	3.560	1.385	1.177
Daub-20	0.526	3.499	1.841	1.356
Daub-8(LA)	0.111	4.172	0.463	0.680
Daub-10(LA)	0.168	3.842	0.647	0.804
Daub-12(LA)	0.138	3.693	0.511	0.715
AHQMF-4	0.100	14.94	1.488	1.220
AHQMF-6	0.141	9.501	1.345	1.160
AHQMF-12	0.394	4.616	1.817	1.348
AHQMF-16	0.557	4.461	2.484	1.576
Coiflet-6	0.086	11.46	0.986	0.993
Coiflet-12	0.110	4.075	0.447	0.668
Coiflet-18	0.132	3.665	0.483	0.695
Coiflet-24	0.152	3.534	0.535	0.732

As the wavelet satisfies a zero moment, H(-1) = 0 and hence the time-frequency localization of a wavelet is also lower bounded by $\frac{1}{2}$.

Table-5.4 shows the time, frequency and joint time-frequency localization of scaling functions of the different wavelets. Table-5.5 shows the corresponding values for the wavelet function. This is also graphically shown in Fig. 5.1. It is observed that frequency localization improves slowly with filter length, but time localization degrades more rapidly. As a result, time-frequency window has a minima somewhere between 10-18 taps, depending on the wavelet type. For Daubechies family, least asymmetric 8-12 wavelets provide the best time-frequency window.

Table-5.5
Time-frequency localization of wavelet function

Wavelet Fn.	σ_n^2	σ_w^2	$\sigma_n^2\sigma_w^2$	$\sigma_n\sigma_w$
Haar	0.082	226.3	18.60	4.313
Daub- 4	0.122	56.70	6.895	2.626
Daub- 6	0.180	9.221	1.658	1.288
Daub- 8	0.251	3.858	0.967	0.983
Daub-10	0.332	2.508	0.832	0.912
Daub-12	0.423	1.941	0.822	0.907
Daub-16	0.633	1.503	0.952	0.975
Daub-20	0.877	1.332	1.168	1.081
Daub-8(LA)	0.156	3.97	0.620	0.787
Daub-10(LA)	0.203	2.503	0.508	0.713
Daub-12(LA)	0.193	1.941	0.375	0.612
AHQMF-4	0.123	60.16	7.421	2.724
AHQMF-6	0.200	30.94	6.184	2.487
AHQMF-12	0.633	7.157	4.529	2.128
AHQMF-16	0.909	5.538	5.034	2.244
Coiflet-6	0.110	46.00	5.049	2.247
Coiflet-12	0.158	3.504	0.553	0.744
Coiflet-18	0.196	1.83	0.358	0.598
Coiflet-24	0.227	1.439	0.327	0.572

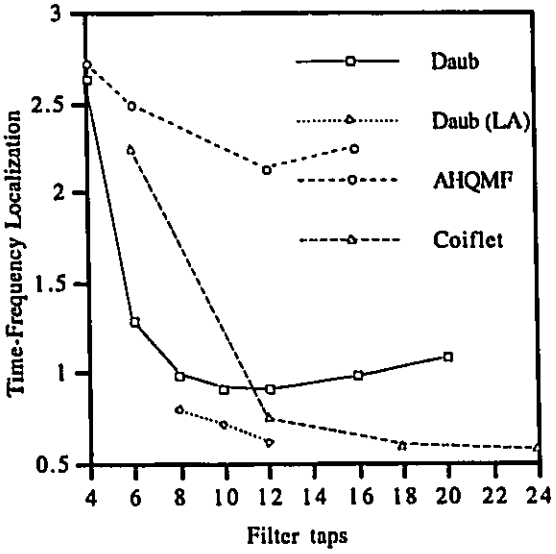


Figure 5.1 : Time-frequency localization of various wavelets

5.4 Transform coding gain

In section 2.3.2, we defined the transform coding gain, which is a useful measure for comparing the energy compaction capability of the various transforms. Here, we rewrite Eq. 2.35 so that it can be easily used to compute the transform gain of various filters. Let us assume that there is a total of M bands after the decomposition and the total number of pixels in *k-th* band is P_k. The arithmetic and geometric mean of the variances can be expressed as

$$\sigma_{am}^2 = \frac{1}{N}\sum_{k=1}^{M} P_k \sigma_k^2$$

$$\sigma_{gm}^2 = \left(\prod_{k=1}^{M} \left(\frac{\sigma_k^2}{P_k} \right)^{P_k} \right)^{1/N}$$

where, N is the total number of coefficients, *i.e.*, $N = \sum_{k=1}^{M} P_k$

Hence, the *transform coding gain* can be expressed as,

$$TCG = \frac{\sigma_{qm}^2}{\sigma_{gm}^2} = \frac{\frac{1}{N}\sum_{k=1}^{M} P_k \sigma_k^2}{\left(\prod_{k=1}^{M}\left(\frac{\sigma_k^2}{P_k}\right)^{P_k}\right)^{1/N}}$$

Fig. 5.2 shows the transform gain of various wavelets on different test images. Fig. 5.2a shows the comparison of TCG for 2, 3 and 4 stage dyadic decomposition of Lena image. It is seen that there is an appreciable gain from 2 stage to 3 stage decomposition. However, beyond 3 stage decomposition, only a marginal gain of TCG is achieved. This has been observed with all the images. In Fig. 5.2b-f, we compare the TCG's of various filters only for 3 levels of decomposition. Fig. 5.2.b compares the TCG's of various wavelets on Lena image. We observe that longer tap filters provide a higher TCG. However, for Daubechies family it is seen that, beyond 16 taps the TCG saturates. In fact, the 16-tap filter has a higher TCG than the 20 tap filter. It is also observed that Daub-4 and AHQMF-4 provide almost similar performance, but in the 6 tap case, AHQMF-6 has a higher gain than Daub-6 and substantially higher than Coiflet-6. We also note from Fig. 5.1b that the least asymmetric Daubechies wavelets have a higher gain than their minimum phase counterpart. This result is probably due to the better time-frequency localization of these least asymmetric wavelets.

The TCG of Lena, Girl, Airplane, Sailboat, *etc.* range from 25 to 40 for a 3 stage decomposition using the given wavelets and hence it is expected that an appreciable coding performance will be achieved by transform coding. However, in the case of Mandrill and Galapagos Island, we observe that the TCG's are very low, in the range 4.0-5.0. Lower TCG's have also been observed for Airport, Safsaf and Death Valley images, whose results have not been not shown. This suggests that these images contain an appreciable amount of details and transform coding cannot provide high

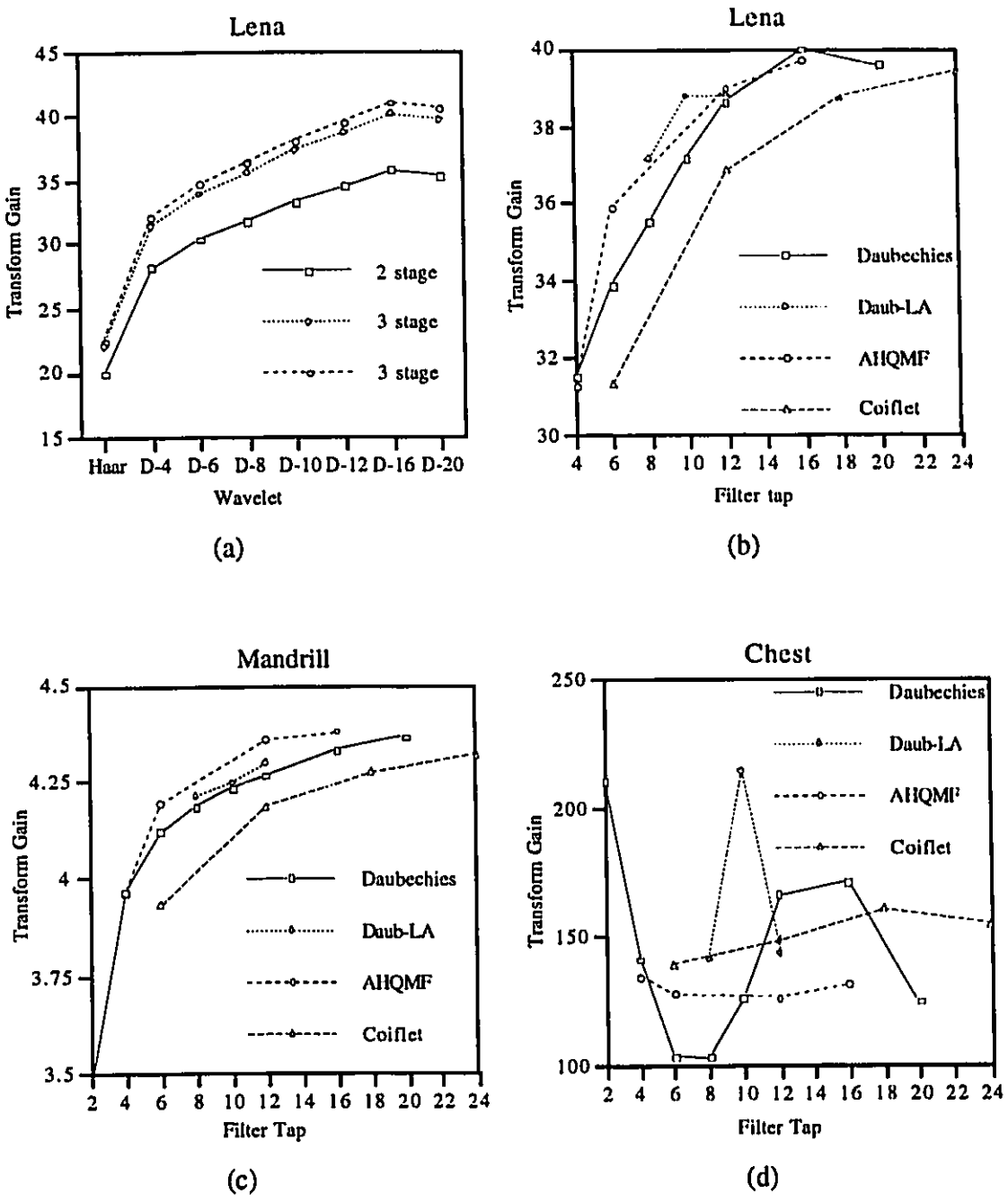
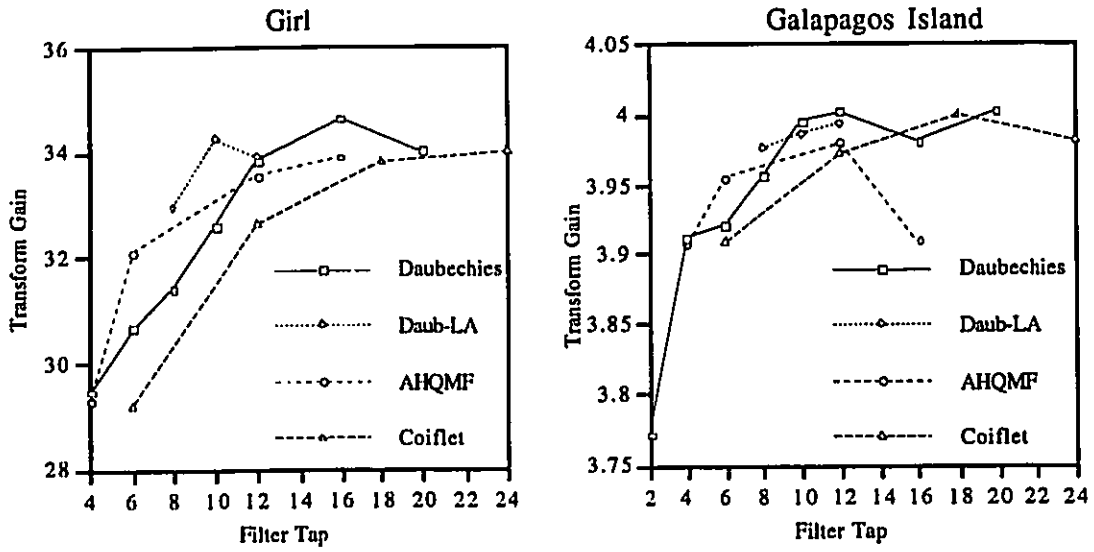


Figure 5.2 : Comparison of transform coding gain of various filters on different test images



(e) (f)

Figure 5.2 (Continued).

coding performance. It is observed that, the fluctuation of TCG's of various wavelets is very small for images with high activity. Hence, by using an optimal wavelet, an appreciable improvement in coding performance may not be achieved.

From Fig. 5.2d, we note that the TCG of chest image is very high. This is also true for visualmtf, whose TCG's has not been shown. The higher TCG of these images comes from their low spectral content (which is measured by the spectral flatness measure). From the figure, we observe that the TCG is very much sensitive to the change in filters. Among the filters considered here, Haar and least asymmetric Daub-10 wavelets provide the best coding gain. The reason for the sensitivity of TCG can be understood from Table 5.6 which shows the variance of different bands of Lena, decomposed by Daub-10 wavelet and chest decomposed with Daub-10 and Haar wavelet. It is observed from the table that spectral properties of Lena is well behaved while that of chest shows occasional spikes in the wavelet domain. For the chest image, the variance map is ill conditioned to calculate the TCG. If the variance of one of the higher resolution bands is very low the overall geometric mean of the variance of

the coefficients will change appreciably and consequently the TCG will vary somewhat randomly. It can be seen from Fig. 5.2d that while changing from Haar to Daub-4, the TCG changes from 210 to 140. We suggest that in these cases (*i.e.* for low SFM), the TCG should be used carefully as a measure of performance [63].

Table-5.6
Variance of various bands of Lena and Chest image

Bands	Lena	Chest	
	Daub-10	Daub-10	Haar
c^3	163489.9	150027.1	151405.6
$d^{3,h}$	5611.7	2618.0	3265.4
$d^{3,v}$	1696.1	895.5	484.6
$d^{3,d}$	1492.4	38.9	38.5
$d^{2,h}$	691.7	743.3	557.0
$d^{2,v}$	265.2	69.2	79.9
$d^{2,d}$	176.3	3.5	4.4
$d^{1,h}$	81.2	68.9	32.6
$d^{1,v}$	34.8	30.6	6.4
$d^{1,d}$	16.5	0.52	0.7
Transform Gain	37.2	125.9	209.8

5.5 Objective performance

In the last section, we compared various wavelets on the basis of their transform coding gain. Here, we will compare the objective coding performance using the scheme-2 of section 4.2.3. The coding performance is expressed in peak signal to noise ratio (PSNR) which is defined as

$$PSNR = 10 \log_{10}\left(\frac{255^2}{MSE}\right)$$

where MSE stands for the mean square error and is the average energy of the error image.

In Fig. 5.3-5.8, we compare the coding performance of various wavelets on Lena, Mandrill, Visualmtf and Chest images. The relative performance of other images are

very similar to that of Lena or, Mandrill image and hence have not been shown here. Visualmtf and Chest images showed a different performance pattern and hence detailed analysis of their performance has been shown. In Fig-5.3-5.6, we compare the performance of various wavelets within individual wavelet family for a wide range of bit-rates.

In Fig. 5.3, the performance on Lena image is considered. It is observed that the coding performance improves initially with increasing number of taps. However, beyond 12-18 taps, the performance saturates and sometimes even degrades. For Daubechies/ AHQMF family, the performance saturates at 10-12 taps. However, for Coiflet family, the performance improves slowly even beyond 18 taps. It is believed that this is a result of the time-frequency localization of Daubechies family degrading after 10 taps, whereas Coiflet family maintains similar time-frequency localization even for higher number of taps. The performance of various wavelets on Mandrill image is shown in Fig. 5.4. It is observed that the relative performance is similar to that for Lena image. Similar performance patterns have also been observed for other images.

In Fig. 5.5, we observe that the coding performance on Chest image, varies appreciably over different wavelets. For Daubechies wavelet, the coding performance oscillates with the filter order. For AHQMF family, the performance degrades monotonically with increasing number of taps. The order of change in the performance is appreciably more than that observed in case of Lena/Mandrill images. However, in Coiflet family, the performance variation over wavelets with different order, is relatively small. Fig. 5.6 shows the performance of various wavelets on Visualmtf. We observe that the change in objective performance is very high compared to that observed in case of Lena/Mandrill images. Daub-16 provides 3-4 dB performance improvement over Daub-4. Similar relative performance is also seen for AHQMF and Coiflet family.

In Fig. 5.7, we have shown the bit-rates required to code the above images at a fixed PSNR. We observe that the relative performance for Lena, Mandrill and Visualmtf images, are very similar. For lower order filter, Daubechies (min. phase) wavelet and AHQMF provide competitive performance and it is dfficult to say which wavelet will perform better. Coiflets provide poor performance at lower order. However, with increasing number of taps, their performance improves steadily. For most images, it has been observed that Coiflets outperform Daub/AHQMF wavelets at a filter order of 18. For Chest image, we observe that the relative performance is much different from others. It is observed that the performance of Daubechies wavelet is oscillating with the filter order. The performance of AHQMF is appreciably poorer than Daubechies wavelet. Coiflet provides the best performance and the performance is invariant with the filter order.

Fig. 5.7 and 5.8 provide the comparison of minimum phase and least asymmetric Daubechies wavelet. It is observed that for most images, appreciable performance improvement is achieved with least asymmetric Daubechies wavelet of a given support. These least asymmetric wavelets have better time-frequency localization. In addition, they have better phase linearity. For *chest* and *visualmtf*, an appreciable improvement (more than 1 dB) can be achieved using least asymmetric Daub-8 over minimum phase Daub-8. For Visualmtf and Chest images, we observe that Daub-LA wavelets provide poorer performance beyond 10 taps. However, for all other images they saturate at higher order and can be used to achieve a better coding performance.

Transform Coding Gain versus Objective Performance

To find the utility of transform coding gain as a measure of coding performance, we compare Fig. 5.2 and Fig. 5.7. We observe that the coding performance does not correspond well to the coding gain. For example, the coding gain of Coiflet-18 is less than Daub/AHQMF-16 for most images and we expect that Daub/AHQMF-16 will perform better than Coiflet-18. However, in practice the coding performance of Coiflet-

18 is better than Daub/AHQMF-16 for most images. If we analyze critically, we will find many such small deviations. These little deviations of performance behaviour are expected because coding gain (TCG) provides a measure of the energy compaction of a wavelet for a specific image and it does not have the knowledge of the quantization and coding scheme to be followed. In our analysis we have employed an entropy coder and this may be another reason for such deviations. We believe that the performance of non-uniform Llyod-Max quantizer which allocates bits according to *pdf* and variance of the coefficients, will provide a better performance match to the TCG (since the TCG is calculated based on the variance of the coefficients).

However, if we compare the TCG and objective performance, they correspond well on average for most images. We observe that Chest and Visualmtf images are the exceptions. For Chest image, Haar wavelet (*i.e.* Daub-2) provides the highest TCG, but its actual coding performance is poorer than most others. Similar performance has been observed for Visualmtf image. One primary reason of this deviation may be that the TCG's for these images are ill conditioned (This has been discussed in detail in section 5.4). In practice, while quantizing the coefficients, the low energy bands of the decomposed images (see Table 5.6) are discarded in the first instance and hence the energy distribution of the high and moderate energy bands have a good influence over the coding performance. So, we suggest that for low activity images TCG's may not provide a good indication of the coding performance of the wavelet.

5.6 Subjective Performance

In the previous section, we compared the objective coding performance of the various wavelets on several test images. It was observed that inside a wavelet family, the objective coding performance improves initially when the number of filter taps is increased. However, beyond 10-16 taps, the performance saturates and sometimes degrades. In this section, we compare the subjective performance of various wavelets.

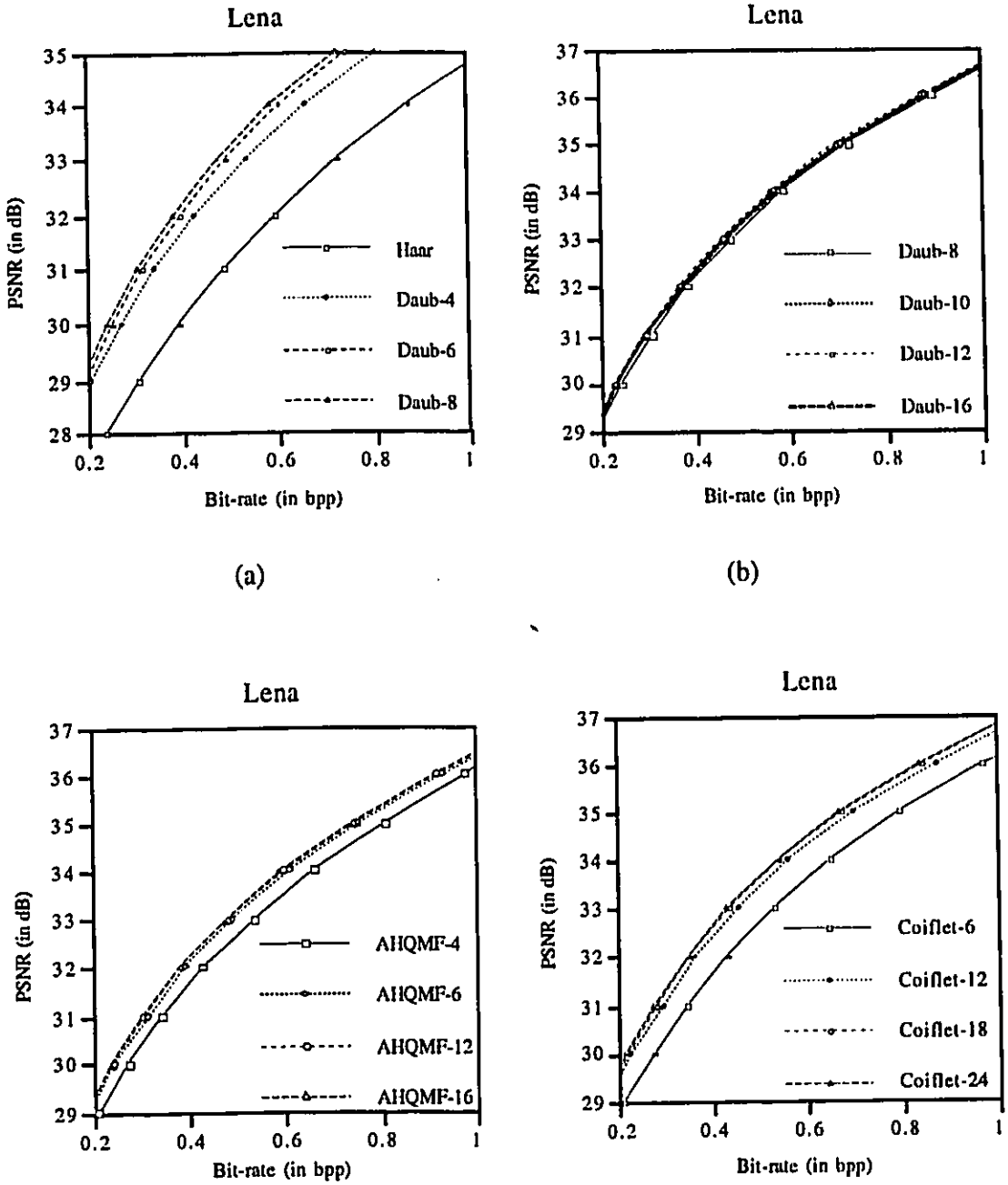


Figure 5.3 : Comparison of objective performance of various wavelets on Lena image

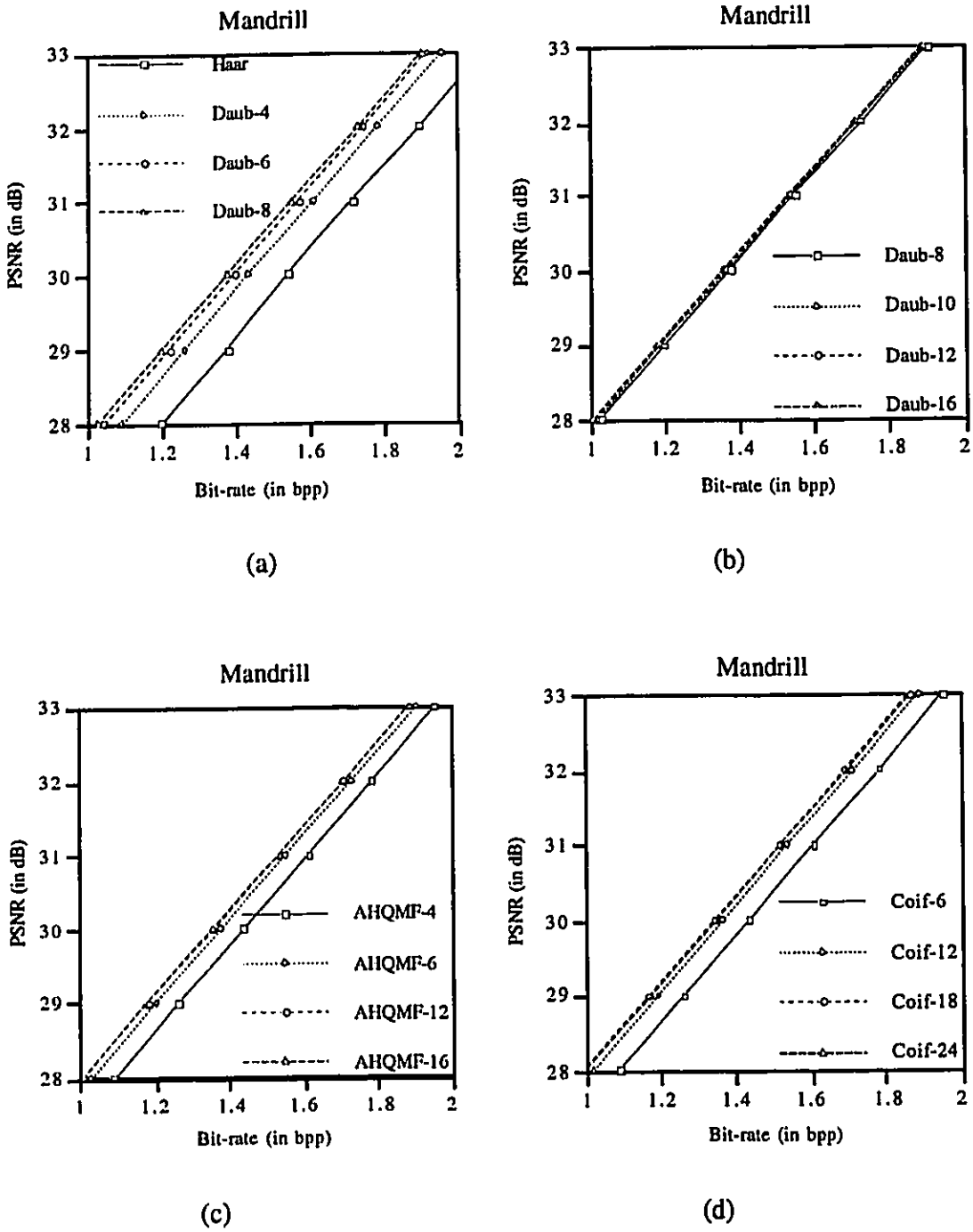


Figure 5.4 : Comparison of objective performance of various wavelets on Mandrill image.

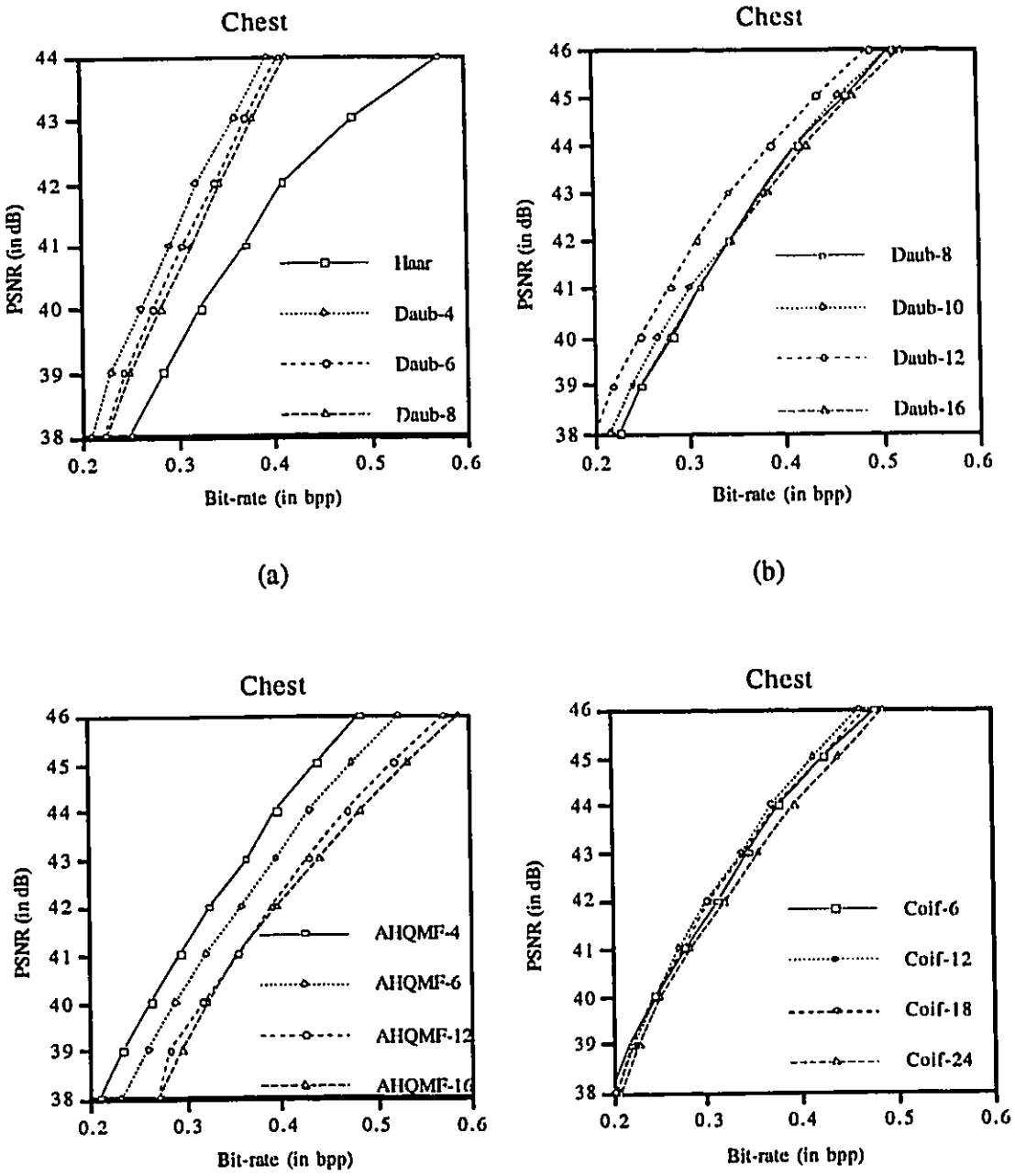


Figure 5.5 : Comparison of objective performance of various wavelets on Chest image.

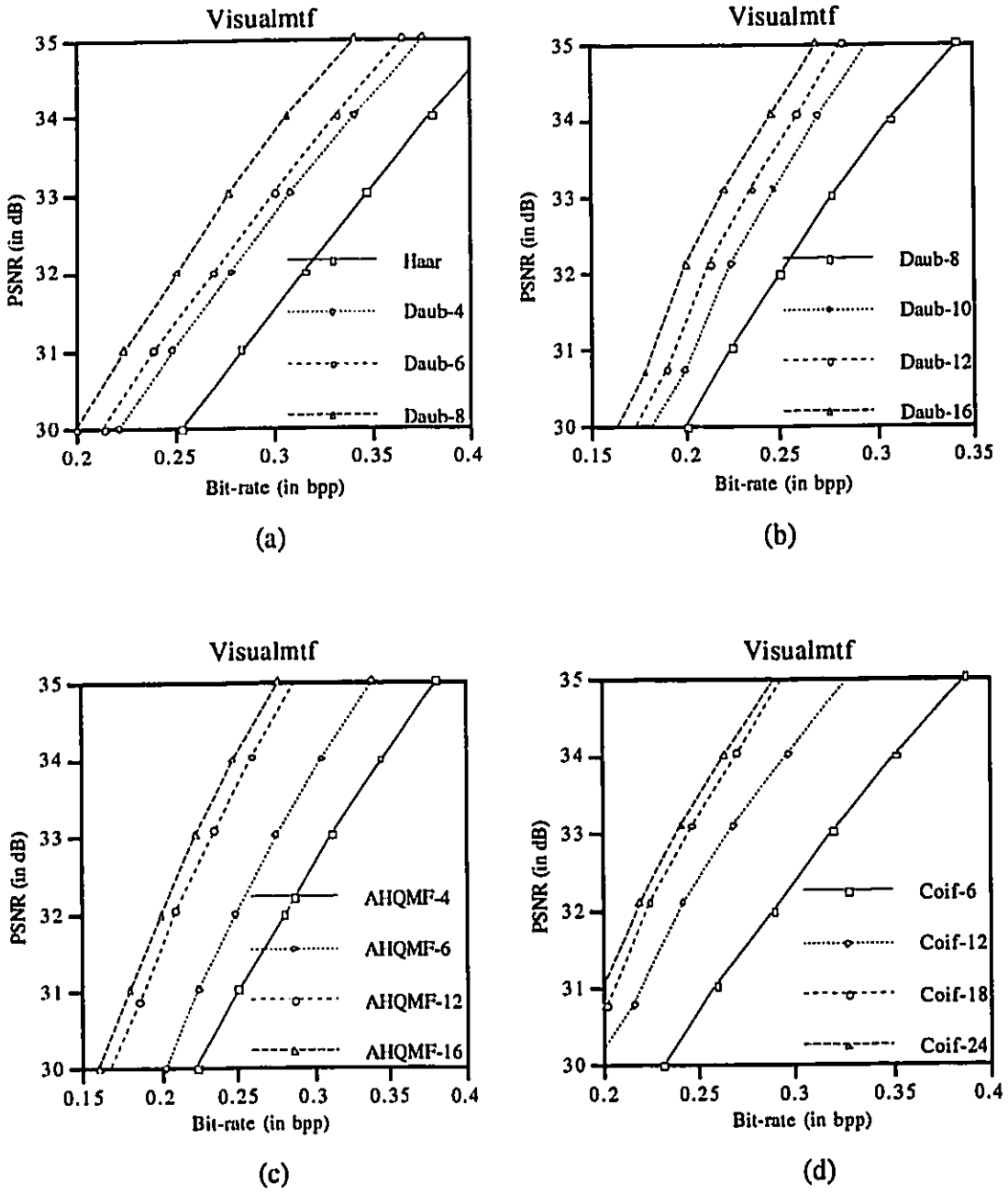


Figure 5.6 : Comparison of objective performance of various wavelets on Visualmtf image.

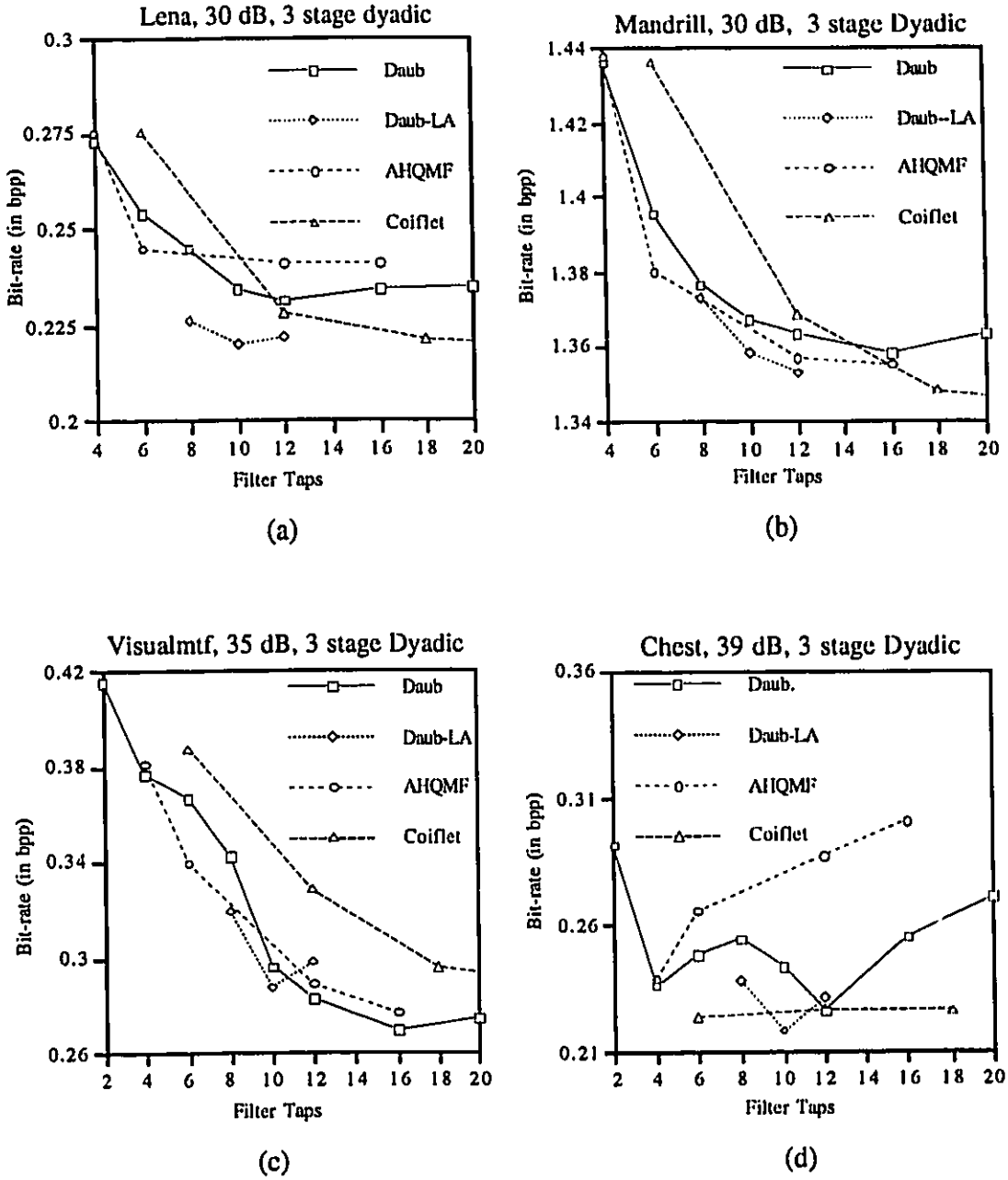


Figure 5.7 : Relative objective performance of three wavelet families

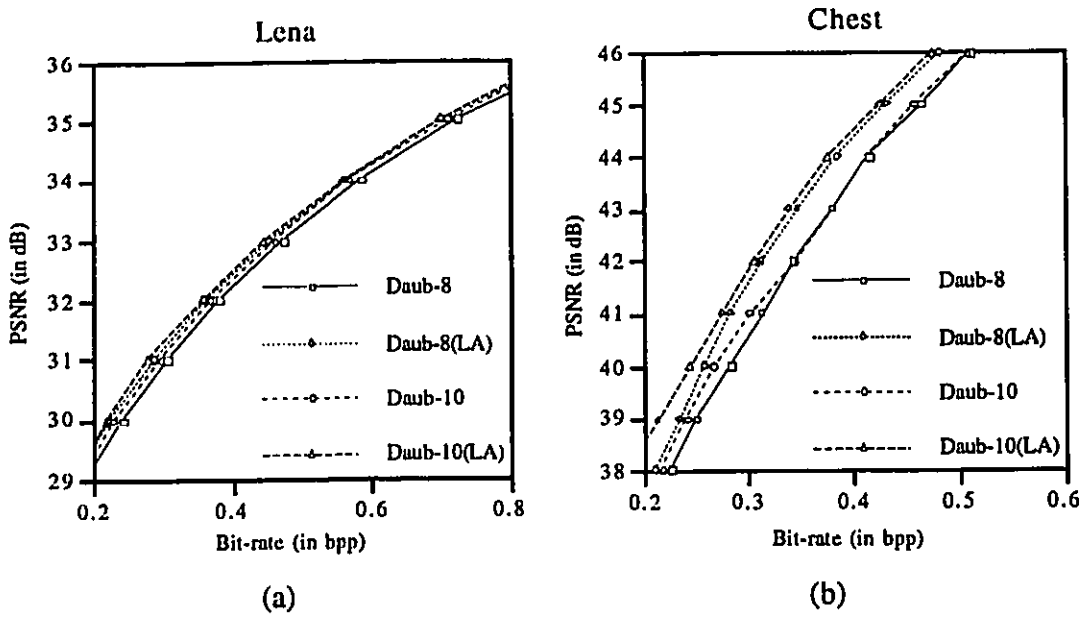


Figure 5.8 : Comparison of objective performance of minimum phase and least asymmetric Daubechies wavelets.

We do this in two different ways which are discussed below.

First, we code the image at a fixed distortion. Table 5.7 provides the bit-rate of Lena image coded at 29 dB PSNR using the various wavelets. The objective of this test was to compare the importance of regularity and time localization. Since, the images are coded at a fixed PSNR, the total amount of noise is same in all the images. Fig. 5.9 shows the Lena image coded at 0.25 *bpp*, by Haar, Daubechies 2, 6, 10, 16 tap wavelets. Fig. 5.9a shows the *Lena* image coded by Haar wavelet. It is observed that the edges are well represented. However, there is annoying checkerboard effect, specially in smooth regions. This is due to the box function like basis functions of Haar wavelet. Fig. 5.9b shows *Lena* image coded by Daub-6. The checkerboard effect is no more visible. However, the edges are not as sharp as that of Fig. 5.9a. In addition, the edge between the chin and the shoulder region is not distinguishable. However, the overall quality is much better than that of Fig. 5.9a. The subjective performance of

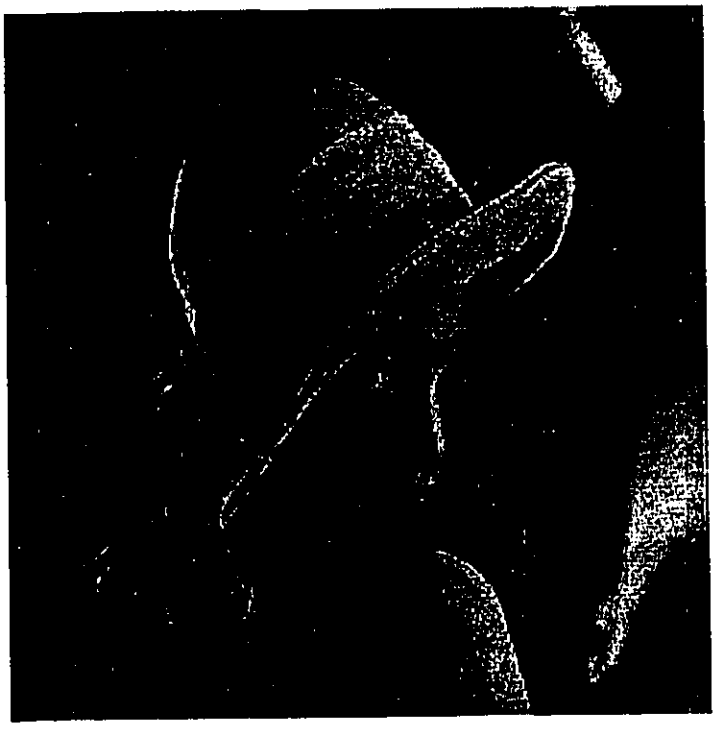
Daub-10 is shown in Fig. 5.9c. The visual quality of Fig. 5.9b and Fig. 5.9c are very similar. Some of the edges such as the lip, the white bar at top-right corner, *etc.*, are better represented in Fig. 5.9b. On the other hand, eyes, chin and shoulder border, are better represented in Fig. 5.9c. The least asymmetric Daub-10 provides a marginal improvement over minimum phase Daub-10 wavelet.

Fig. 5.9d shows the performance of Daub-16. The quality of this image is poorer than that of Fig. 5.9b and Fig. 5.9c. The edges are blurred. This is more pronounced in the chin and shoulder region. The performance of Daub-20 is not shown here, but the subjective performance is worse than that of Daub-16. There are artifacts throughout the image. These artifacts are due to the poor time localization of Daub-20.

In the above subjective test, we observe that though the regularity of Daub-16 is much higher than that of Daub-6, subjective quality of the reconstructed image coded by Daub-16, is poorer than that coded by Daub-6. We anticipate that the poorer time localization of Daub-16, is the main reason for its poor performance. We stress that increasing the regularity of wavelet beyond 2, may not improve the subjective performance. In that case, other filter properties will have more control on coding performance.

We have compared the subjective performance of various wavelets at a fixed PSNR. We now compare the subjective performance at a fixed bit-rate. Fig. 5.10 shows the Lena image coded at 0.25 bpp by Daub 6, Daub-10, Daub-16 and Daub-10 (LA). We observe that Daub-10 provides better performance than Daub-6 and Daub-16. The chin and shoulder border of Fig. 5.10b are better represented than both Fig 5.10a and Fig 5.10b. The edges in Fig. 5.10a shows some roughness and also there are few artifacts present in the image. In Fig. 5.10c, the edges are too blurred and also there are artifacts throughout the image. Fig. 5.10b provides a good balance. Fig. 5.10d shows the performance of least asymmetric Daub-10 tap wavelet. We observe that

(a)



(b)

Figure 5.9 : Subjective performance of various wavelets on Lena image. All images were coded at 29 dB PSNR. The images were coded by a) Haar, b) Daub-6, c) Daub-10, d) Daub-16.

(c)

(d)

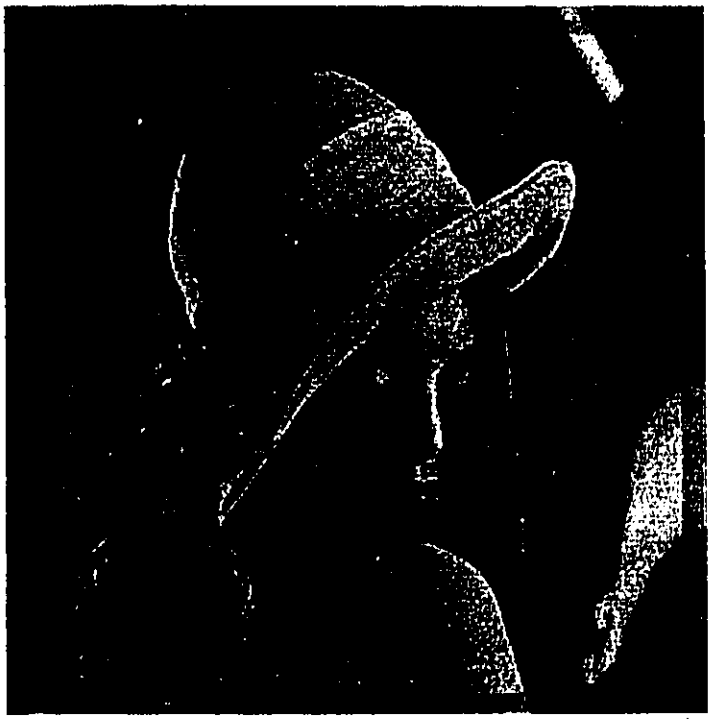
Figure 5.9 : (Continued)

(a)

(b)

Figure 5.10 : Subjective performance of various wavelets on Lena image. All images were coded at 0.25 *bpp*. The images were coded by a) Daub-6, b) Daub-10, c) Daub-16, d) Daub-10(LA).

(c)



(d)

Figure 5.10 : (Continued)

Fig. 5.10d is sharper than Fig. 5.10b and overall looks better. We note that the PSNR of Fig. 5.10d is about 0.3 dB more than that of Fig. 5.10b, due to the better energy compaction ability of Daub-10(LA).

Table-5.7
Bit-rate of Lena image
at 29 dB PSNR

Wavelet	Bit-rate (in bpp)
Haar	0.307
Daub-6	0.196
Daub-10	0.182
Daub-16	0.182
Daub-20	0.181
Daub-10(LA)	0.174

5.7 Search for the optimal wavelet

In the previous sections, we compared the coding performance of three wavelet families. These wavelets provide a good performance. However, to achieve the best performance, we have to search for a wavelet which will provide the optimal performance for the specific image under consideration. In this section, we will search for the optimal wavelet for the test images individually at a fixed PSNR. The amount of improvement in performance will determine if such a search is waranted or not. Searching for optimal wavelet is in general very compute intensive. Hence, we limit ourselves to the case of 4, 6 and 8 tap wavelets. For 4 and 6 tap cases, we use Pollen's parameterization. For 8 tap case, we have used Zou's parameterization. These two parameterization schemes have been discussed in section 3.5. Table-5.8 shows the parameters for the various wavelets.

5.7.1 Four-tap case

Finding the optimal wavelet for 4 tap case is relatively easy as there is only one free parameter. Fig. 5.11 shows the bit-rate obtained using various wavelets on Lena,

Mandrill and Galapagos Island for specified PSNR of 30 dB. It is observed that the three performance curves are very similar and the minima of all the three curves are around parameter 1.0 and 5.2. The wavelets corresponding to θ and $(2\pi-\theta)$ are minimum and maximum phase (*i.e.* one is the time reversed sequence of the other) of the same wavelet. Hence the performance curves are symmetrical around parameter π. It is observed that minimum bit-rate in all the cases is achieved with parameter somewhere in the range 0.95-1.2. We note that Daubechies wavelet (whose Pollen parameter is 1.04) falls in this region. Fig 5.11 (b), (d), (f) show the blow ups of the curves around the minima. It is seen that the performance around the minima does not change appreciably within some parameter range and hence any wavelet in this range can be used without sacrificing coding performance. Table 5.9 shows the optimal wavelet for various images. We note that the optimal wavelet provides negligible performance improvement over Daub-4. It can be concluded that among 4 tap wavelets, one may use Daub-4 without sacrificing any coding performance compared to the optimal 4-tap wavelet [65].

Table - 5.8
Parameters of Daub/AHQMF/Coiflet wavelets

Wavelets	4 tap (Pollen)	6 tap (Pollen)	8 tap (Zou)
Daubechies	1.047197	1.359803 -0.782106	2.044428 1.363058 1.616764
Daubechies 8 tap (LA)	NA	NA	1.131350 2.840627 1.967648
AHQMF	1.0140	1.350 -1.006	NA
Coiflet	NA	1.147 0.424	NA

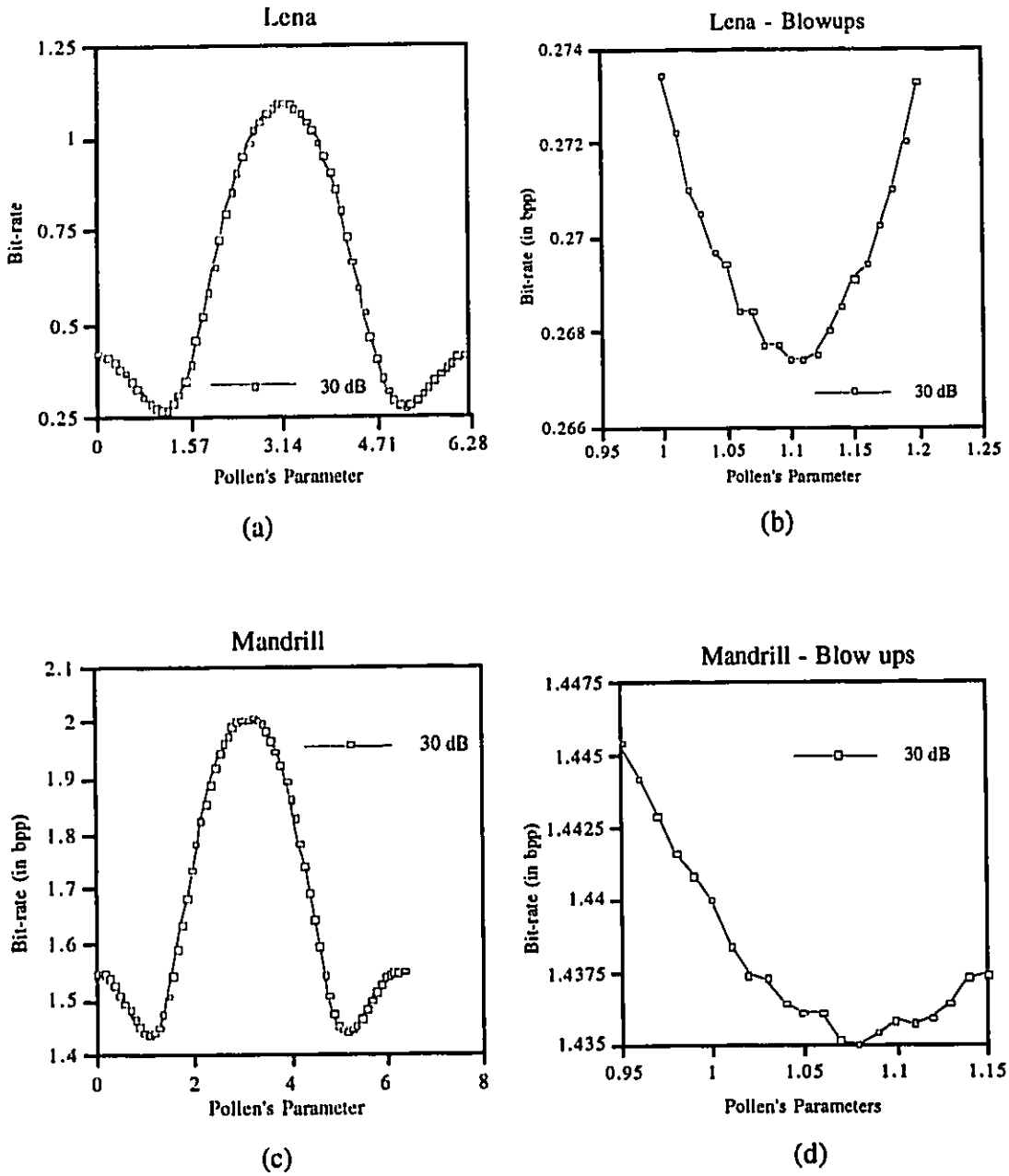


Figure 5.11 : Coding performance of various 4 tap wavelets. a) Lena, coded at 30 dB, b) Mandrill, coded at 30 dB, c) Galapagos Island, coded at 25 dB PSNR.

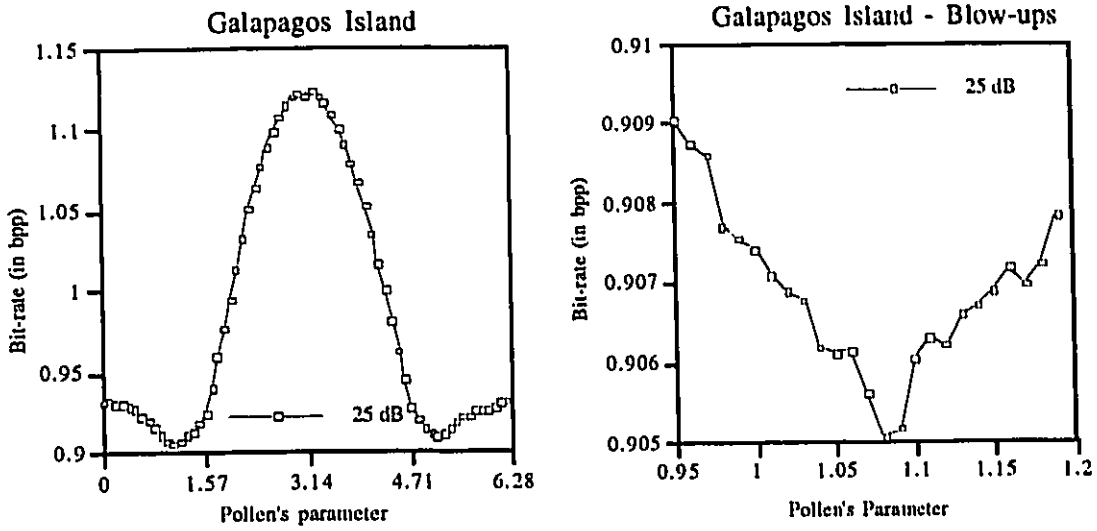


Figure 5.11 (Continued).

5.7.2 Six-tap case

The performance curves of 6 tap-wavelets for Lena and Mandrill are shown in Fig. 5.12a and 5.12b, respectively. It is seen that the 2-D performance plots are very similar. For better visual interpretation we have plotted the reciprocal of the bit-rate. We see that the maxima's occur almost in the same parameter space. Similar plots have also been obtained for other images. Table 5.10 shows the parameters of the wavelets which provides minimum bit-rate at 30 dB PSNR and the maximum transform gain for all the test images. It is seen that the transform gain corresponds to the minimum bit-rate for all images except airport, visualmtf and chest. This is due to the irregular nature of the spectral properties of these three images. Table-5.11 provides the performance gain of optimal wavelet over Daub-6, AHQMF-6 and Coiflet-6. It is seen that for highly active images, the performance improvement is negligible with optimal wavelet. For images with medium activity, AHQMF-6 performs very close to the optimal wavelet. However, for chest image, Coiflet-6

provides the best performance. In general, for images with nonmonotonic spectrum, the optimal wavelet is not easily predictable [63].

Table - 5.9
Performance improvement due to optimum 4 tap wavelet

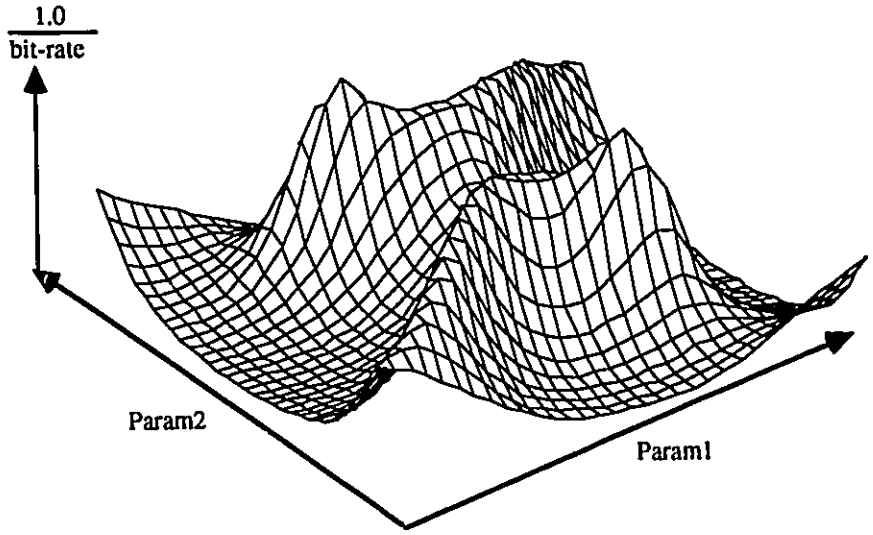
Image	Pollen's Param-eter	Minimum Bit-rate[1] (in bpp)	Range of Pollen's parameters[2]	Bit-rate with Daub-4	% gain[3] over Daub-4
Airport	1.23	1.909	1.14-1.32	1.913	0.23
Mandrill	1.08	1.435	1.03-1.14	1.436	0.07
Airplane	0.99	0.306	0.93-1.04	0.369	0.33
Sailboat	1.09	0.541	1.07-1.15	0.544	0.75
Girl	1.14	0.107	1.07-1.20	0.108	0.93
Lena	1.10	0.267	1.06-1.14	0.273	1.12
Visualmtf	1.14	0.220	1.07-1.21	0.222	0.91
Chest	1.08	0.231	1.06-1.11	0.231	0.13

[1]Bit-rate at 30 dB PSNR for all images except chest for which it is 39 dB
[2]Range of Pollen's parameters at which bit-rate is almost same
[3]Improvement due to optimal wavelet over Daubechies wavelet

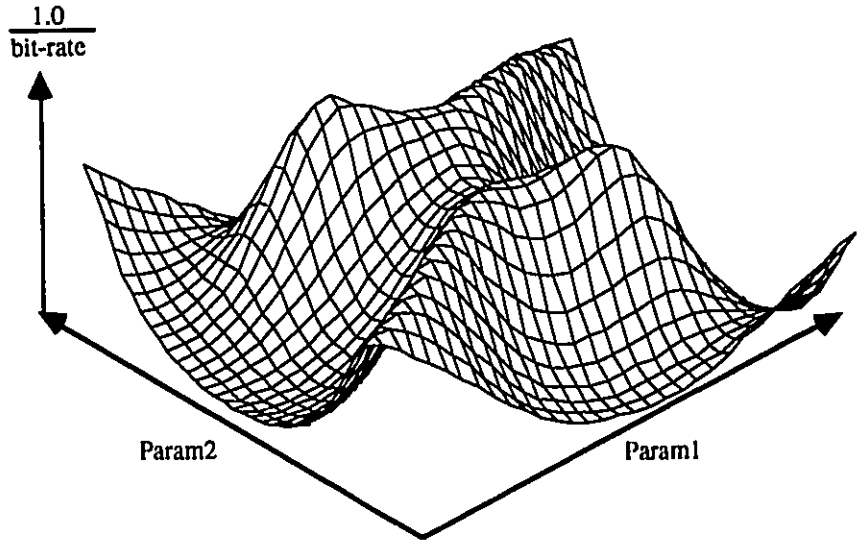
5.7.3 Eight tap case

Searching for an optimal 8-tap wavelet is much more computationally intensive than 4 or 6 taps since there are 3 variable parameters in this case. Table 5.12 provides the minimum bit-rate achieved when coding various test images at 30 dB PSNR. The entries in the second column provide the transform gain associated with the wavelet which produces minimum bit-rate. The three free parameters corresponding to the optimal 8-tap wavelet as described in [114] are also provided in the Table-5.12.

The performance improvement due to optimal wavelet over Daub-8 wavelet is shown in Table-5.13. It is observed from the Table that the least asymmetric Daub-8 wavelet provide very close performance to the optimal wavelet. We also note that an improvement of only 1-5 % can be achieved when the optimal wavelet is used in place of Daub-8 (LA) wavelet for most images. The only exception is again visualmtf and chest, where an appreciable improvement in coding performance is observed. Thus, we



(a)



(b)

Figure 5.12 : Coding Performance of various 6 tap wavelets
a) Lena and b) Mandrill, coded at 30 dB PSNR

Table - 5.10
Optimal 6 tap wavelets for various images

Image	Minimum Bit-rate		Max. Transform Gain	
	Bit-rate Trans. Gain	param1 param2	Bit-rate Trans. Gain	param1 param2
Airport	1.907 1.395	1.55 -1.25	1.925 1.430	1.25 -0.95
Mandrill	1.377 4.206	1.30 -0.95	1.379 4.209	1.30 -1.00
Airplane	0.287 27.172	1.38 -0.92	0.291 27.820	1.28 -0.90
Sailboat	0.483 28.244	1.34 -0.94	0.490 28.372	1.28 -0.91
Girl	0.096 31.998	1.39 -1.03	0.096 32.108	1.34 -0.99
Lena	0.240 35.48	1.39 -0.97	0.242 35.850	1.35 -1.00
Visualmtf	0.199 26.199	1.25 -1.00	0.253 356.736	1.58 -1.57
Chest	0.218 134.620	1.09 0.31	0.250 217.981	1.55 -1.40

Table - 5.11
Performance improvement due to optimal wavelets
(Bold entries are the ones closest to optimum)

Image	SFM	% Gain of bit-rate due to optimal wavelet over		
		Daub-6	AHQMF-6	Coiflet-6
Airport	Very high	**0.5**	**0.5**	**0.5**
Mandrill	High	1.3	**0.2**	4.3
Airplane	Medium	**0.5**	1.2	5.4
Sailboat	Medium	1.6	**0.5**	11.1
Girl	Medium	7.9	**0.5**	13.1
Lena	Medium	4.7	**0.9**	13.3
Visualmtf	Low	7.9	**2.9**	16.7
Chest	Very low	12.0	19.4	**0.4**

can conclude that for images with medium to high spectral activity, 8 tap LA Daubechies wavelet generally provides performance very close to optimum. For low spectral activity images, we have to search for the optimum.

Table - 5.12
Optimal 8 tap wavelets for various images

Image	Minimum Bit-rate	Transform Gain[1]	Param-1	Param-2	Param-3
Airport	1.901	1.400	1.30	2.95	1.80
Mandrill	1.357	4.301	1.04	2.83	2.02
Girl	0.091	32.872	1.50	3.33	2.23
Sailboat	0.469	28.971	0.86	2.57	1.89
Lena	0.217	38.500	1.00	2.75	1.95
Airplane	0.280	27.393	1.19	2.97	2.06
Visualmtf	0.173	32.304	1.29	0.01	2.16
Chest	0.172	247.070	1.77	0.13	3.03

[1]Transform Gain corresponding to wavelet which provides minimum bit-rate.

In general, if we decide to search for the optimal 8-tap wavelet, one should initially search in the parameter space around Daub-8 (LA). The entries of Table-5.13 indicate that in most cases, the Eucledian distance between the optimal wavelet parameters and that of Daub-8 (LA) is small. However, images with nonmonotonic spectrum are the exceptions and we can search for the optimal wavelets using the multistart algorithm of [96].

Table - 5.13
Performance improvement due to optimal 8 tap wavelets

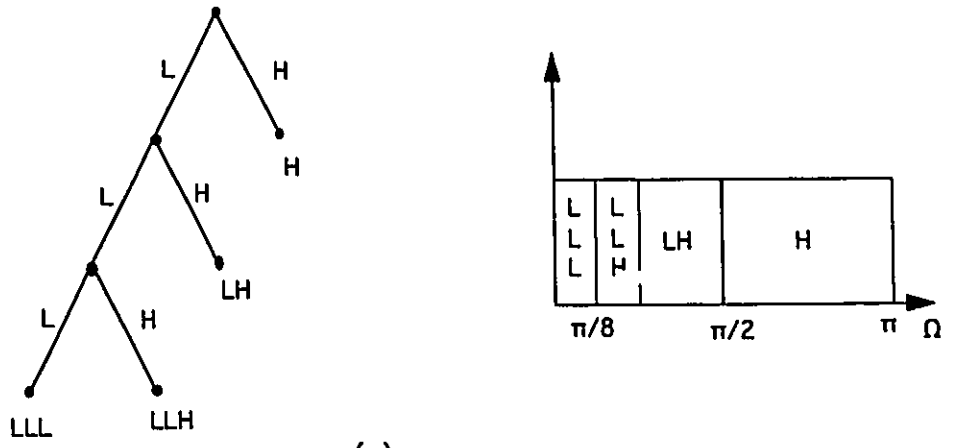
	% Gain of bit-rate due to optimal wavelet over	
Image	Daub-8	Daub-8 (LA)
Airport	1.21	0.50
Mandrill	1.43	1.18
Girl	7.58	3.19
Sailboat	2.43	1.43
Lena	11.43	2.63
Airplane	5.54	0.89
Visualmtf	17.1	7.25
Chest	31.80	22.19

5.8 Adaptive Tree Structure :

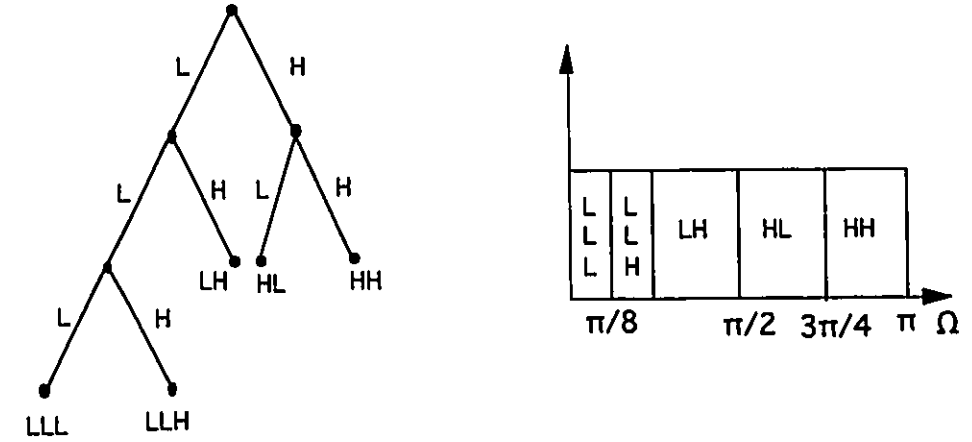
The objective of wavelet/subband decomposition is to compact most of the energy in fewer bands. A pure wavelet decomposition is based on a dyadic tree, *i.e.* only the lowest scale is decomposed recursively. However, it is expected that a regular tree will provide better coding performance at the expense of additional computational complexity. In subband coding, the signal decomposition is based on a regular tree for improved performance. We note that the theory of wavepackets has been developed in wavelet literature [14, 107], where the high-pass bands are also decomposed. The tree structure and the corresponding frequency band split of dyadic, regular and a typical irregular tree decomposition (for 1-D case) are shown in Fig. 5.13, for a 3 level decomposition. Table-5.14 shows the improvement due to regular tree over dyadic tree (more specifically quadtree, for 2-D images). A higher performance improvement is seen for the low SFM images, mainly because the local peaks in the scale space can be finely quantized due to larger number of bands.

However, when the number of bands is too large, the performance deteriorates because of two reasons - i) it is uneconomical to code separately the small bands with the same statistics and ii) the extra information (*e.g.* tree structure, quantization step-sizes *etc.*) need to be sent to the decoder. In addition, the computational complexity also increases with the tree depth. We note that a 3 level regular tree has a computational complexity double that of a 3 level dyadic tree. Hence, it is reasonable that an irregular tree decomposition will provide best coding performance with a computational complexity between dyadic and regular tree.

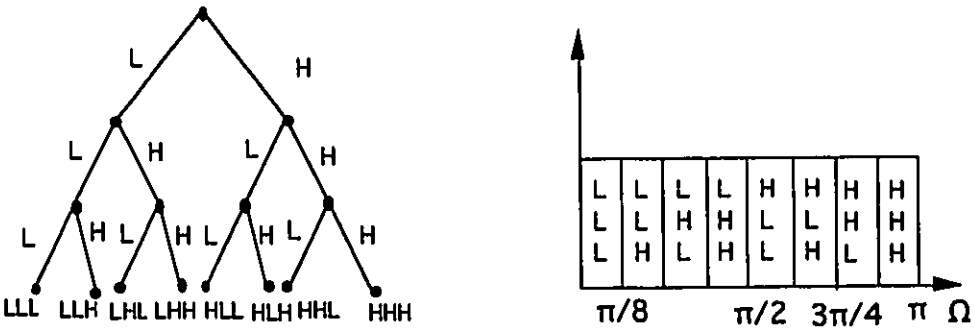
Optimal tree decomposition schemes have been described in [40, 80]. However, the proposed techniques are compute intensive. We propose a simple image adaptive tree decomposition algorithm using local transform gain and energy compaction. When a node is decomposed, four local bands are created. Let the variance of the four local bands be σ_1^2, σ_2^2, σ_3^2, σ_4^2. Then, the local transform gain can be defined as



(a)



(b)



(c)

Figure 5.13 : Various Tree Decompositions (1-D) and Corresponding Frequency Band Split (a) Dyadic, (b) Irregular, (c) Regular

$$TCG_{local} = \frac{\left(\sigma_1^2 + \sigma_2^2 + \sigma_3^2 + \sigma_4^2\right)/4}{\left(\sigma_1^2\sigma_2^2\sigma_3^2\sigma_4^2\right)^{1/4}}$$

The decomposition will result in appreciable energy compaction if the local transform gain, TCG_{local} exceeds a predetermined threshold. Thus, the decomposition is kept if

$$TCG_{local} > 1 + \varepsilon$$

where ε is a small positive number (usually in the range 0.05-0.10). If the above inequality is not satisfied, the decomposition may be considered as non-profitable and the decomposed bands are merged by taking the inverse transform. If the decomposition is considered beneficial, they are further decomposed. Among the four bands, the bands which contain appreciable energy (say more than 5-10% of the total energy of the parent node) are decomposed further. This simple top-down splitting technique performs well for most images. Table-5.14 shows the performance of three level tree decomposition. A regular 3 level decomposition will produce 64 bands whereas a 3 level dyadic decomposition produces only 10 bands. We observe that for most images a performance improvement of 3-20% can be achieved with this adaption. Low activity images may provide a much better coding performance. We note that an irregular decomposition of 20-30 bands provides optimal performance at a significantly reduced computational complexity [63].

5.9 Image Compression Using Complex Wavelet

In the previous sections, we have considered only the real orthonormal wavelets. It is known that symmetrical orthonormal real wavelet (*i.e.* linear phase wavelets) does not exist. However, symmetrical filters are desirable for two reasons. First, they can be cascaded in a pyramidal filter structure without the need for phase compensation. Secondly, perfect reconstruction can be achieved with symmetrical extension of data in the border. This reduces the discontinuities at the borders.

Table - 5.14
Bit-rate for dyadic/regular/irregular trees at 30 dB PSNR[1]

Image	Dyadic Tree	Regular Tree	Irregular Tree		% impr-ovement
			Bit-rate	Bands	
Airport	1.924	1.971	1.924	10	0.00
Mandrill	1.377	1.299	1.297	31	5.81
Airplane	0.296	0.236	0.235	34	20.56
Sailboat	0.481	0.447	0.446	43	7.11
Girl	0.098	0.096	0.095	28	2.44
Lena	0.245	0.216	0.219	28	9.2
Visualmtf	0.342	0.084	0.084	22	75.51
Chest	0.251	0.179	0.178	34	29.08

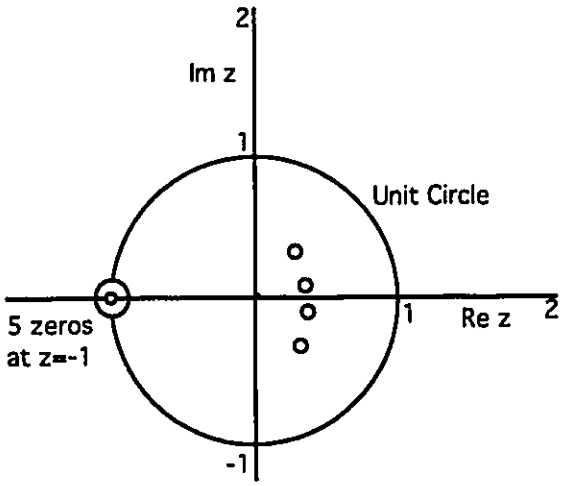
[1]Chest image is coded at 39 dB.

Design of complex wavelets have been discussed in [31, 54, 82]. Let us consider the specific case of Daubechies wavelets. Fig. 5.14a shows nine zeros of Daubechies 10 tap wavelet. If we flip the alternate zeros inside the unit circle relative to their reciprocal conjugate as shown in Fig. 5.14b, the corresponding filter will have a symmetrical impulse response and satisfy all the properties of wavelet. The refinement coefficients of 10 tap Daubechies complex symmetrical wavelet are provided in Table D.5 (Appendix D). We note that the underlying wavelet function will also be symmetric.

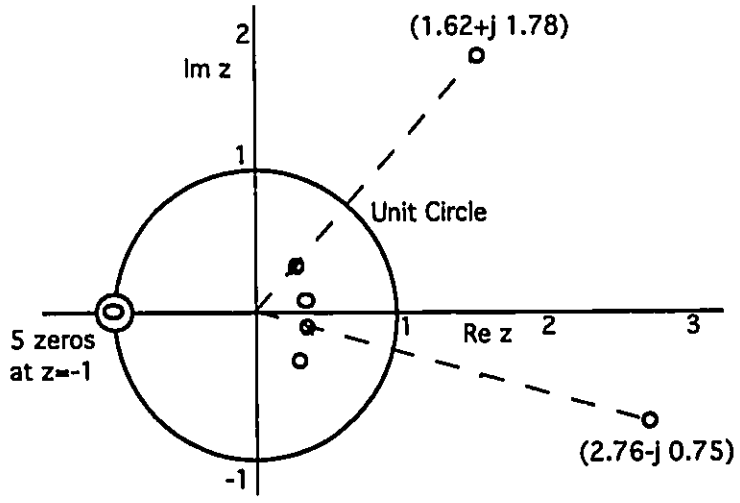
The input image is divided into two equal segments. The first and second segment are considered as real and imaginary components respectively. So the real image can be considered as a complex image with half the size. A complex wavelet transform was taken followed by coding and quantization using scheme-2 of section 4.2.3.

The coding performance of complex Daub-10 wavelet is shown in Fig. 5.15. It is seen that the objective performance of complex wavelet is about 0.5 dB poorer than real Daub-10. However, the greatest penalty comes from subjective quality of the reconstructed image. After decoding when we reconstruct the image by separating the real and imaginary components, each half of the image will have considerable amount of leakage from the other half. This leakage is due to the fact that we treated two unrelated segments of the image as real and imaginary components. So, each wavelet

coefficient represents two regions of the image. Hence, if the coefficients are quantized, after reconstruction, there will be some residual leakage which cannot be avoided. This leakage can be avoided only by treating the whole image as real component and initializing the imaginary component as zeroes. However, in that case, the objective performance may further degrade. Fig. 5.16 shows Lena image coded at 0.25 *bpp*. The improvement expected by the symmetrical extension and phase



(a)



(b)

Figure 5.14 : Zeroes of Daub-10 wavelet. a) Real, b) Complex

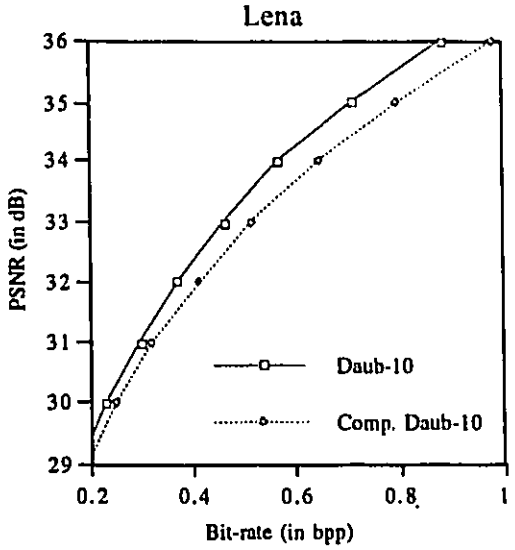


Figure 5.15 : Performance of Complex wavelet on Lena image

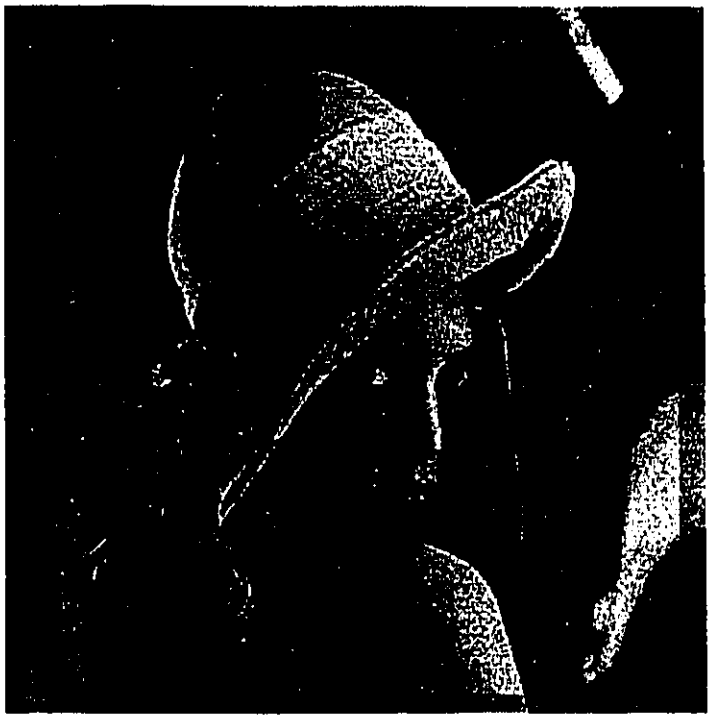


Figure 5.16 : Lena image coded at 0.25 bpp by complex Daub-10 wavelet

compensation from the use of complex wavelet could not be achieved due to the degradation of overall subjective quality of the reconstructed image.

Though, the concept of complex wavelet is novel, unfortunately it was not immediately applicable in compressing real images. However, it might be useful in other applications such as numerical simulation [31], compressing complex radar data *etc.*

5.10 Conclusion

In this chapter, we have presented a thorough analysis of the coding performance of three wavelet families. We compared the wavelets based on their coding gain, objective and subjective performance. It was shown that for most images except the low activity images (such as visualmtf and chest), *transform coding gain* corresponds well with the actual coding performance. Simulation results were provided to show the effect of filter order on the coding performance. It was shown that filter order should not in general exceed 12, with 8-12 taps generally providing the best time-frequency localization. Least asymmetric wavelets have been seen to provide a better coding performance than the corresponding minimum phase wavelet. Spectral flatness was proposed as a measure indicating the need for searching the "optimal" wavelet decomposition. It was shown that such a search was warranted only for low spectral flatness images. Otherwise, *Daubechies* wavelets or *AHQMF's* are close enough to the optimum and not much improvement can be expected by conducting a full search. A simple irregular tree decomposition algorithm was proposed which has been seen to provide better coding performance than both dyadic and regular tree at a reduced complexity. The degree of improvement using the irregular tree, is appreciably more than that is achieved by using optimal wavelets (which are difficult to determine) over known standard wavelets. Finally, we observed that complex symmetrical orthonormal wavelets, inspite of having some interesting properties, did not provide compression comparable to that obtainable by real wavelets.

Chapter 6

HVS Adapted Wavelet Coding

In Chapter 5, we have investigated wavelet-based coding that is optimal in the mean square error (MSE) sense. In this chapter we will incorporate human visual system (HVS) characteristics in the coding scheme to enhance the visual quality of the reconstructed image. First, we describe a Fourier-domain model of the HVS. We then suggest a method for finding the relative importance of various bands of a wavelet. We next propose a quantization scheme optimal in the visually weighted mean square error sense. We then conclude the chapter by summarizing our observations.

6.1. Introduction

The goal of image compression is to reduce the number of bits in the digital representation of an image with minimum perceived loss of subjective quality. In image compression, the image quality is ultimately judged by the human observer. Traditionally, the MSE criterion has been used as a measure of reconstructed image quality. However, it has often been found that the MSE and its variants do not correlate well with subjective quality assessment [66]. This observation is more pronounced at high compression ratio. The reasons are not well understood, but it is believed that the MSE does not adequately track the types of degradations caused by

image compression techniques. In addition, it does not adequately "mimic" what the human visual system does while assessing the image quality.

Some researchers have investigated to quantify the subjective/quality assessment by incorporating elaborate models of the visual process [58]. Developing such a model leads to a greater insight, efficiency, and exactness in the process of developing a prospective compression scheme. Hence, it will provide rapid, precise and mathematically tractable quality assessment in place of time consuming, labour intensive, subjective quality assessments using human observers. Such models have been devised in an attempt to simulate the effects of many of the parameters affecting vision such as orientation, field angle, Mach bands, *etc.*, but their utility for practical problems is limited due to their complexity, inherent unknowns, and need for a detailed *a priori* knowledge of viewing environments.

The effectiveness of an HVS model is ultimately judged by the quality of the reconstructed image, which is usually done by subjective tests and is expressed as a *mean opinion score* (MOS) of a group of human observers viewing the reconstructed image. The Comite Consultatif International de Radiodiffusion (CCIR) has drawn up specifications concerning observers, grading scales, and viewing conditions in Recommendations-500 [59]. One of the most important parameters is the viewing distance which is usually six times the screen height. There are different methods for quality evaluation. The most commonly used one is the impairment scale which is shown in Table-6.1. The observers are asked to provide an integer between 1 and 5 depending on the quality of the reconstructed image and then the mean of the opinion values is considered to be the MOS.

6.2. HVS Model

Modelling human visual system accurately is difficult task. Hence, one has to resort to few assumptions to simplify the development of a model. First, one assumes that

HVS is a linear system which is in general true for low contrast images. This assumption greatly simplifies the quantitative analysis since one can easily apply the knowledge of Fourier based techniques. When nonlinearity is unavoidable, it can be introduced in the HVS model by preprocessing the image with a logarithmic function, *i.e.* by homomorphic filtering. Secondly, one assumes that the HVS is isotropic in spatial domain, *i.e.* eyes do not have any preferential direction (towards horizontal, vertical or diagonal). Though, it has been observed that the sensitivity of eyes in the diagonal direction is about 3 dB less than that in the horizontal or vertical direction, this assumption is used by most researchers [70]. This helps us to extend the one dimensional model to two dimensions easily. Other assumptions can be that the scenes viewed are monochromatic, with observer preferred luminance levels, and are viewed by an observer for a preferred length of time.

Table-6.1

CCIR impairment scale

Q Factor	Quality	Impairment
5	Excellent	Imperceptible
4	Good	Percetible, but not annoying
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very annoying

A number of factors has been seen to affect the noise sensitivity of the eye. The most prominent is the frequency sensitivity of the eyes. Other factors are background luminance and texture masking [100]. We now briefly describe these properties.

Frequency Sensitivity

Many researchers have measured the human threshold contrast sensitivity to periodic patterns *e.g.* sine waves, square waves, *etc.*, viewed at a range of spatial frequencies [43, 56, 66, 67]. In Fig. 6.1, an oscillating high frequency pattern is shown on a grey background in two dimensional space [21]. Any line drawn horizontally across the picture would intersect sinusoidal changes in intensity whose amplitude is constant

but whose frequency increases exponentially from left to right. If we move vertically upward in picture, the amplitude of the wave decreases exponentially. If we hold the picture at an arms length, go up from the bottom of each vertical stripe and mark a point where it is just not visible, the points from the vertical stripes will make a curve similar to that shown in Fig. 6.2. That is, the middle stripes will have the highest sensitivity and the higher and lower frequencies will be visible only when their amplitudes are relatively high. The stripe where the sensitivity will be highest, depends on the distance between the picture and the eyes. If we increase the distance, the peak shifts to the left, because the spatial frequency (in cycles/degree) associated with the stripes will change. The curve of Fig. 6.2 is known as the *modulation transfer function* of the eyes. In the literature, many such curves with little variation of frequency response, have been reported. The spatial frequency at which the eyes have the highest sensitivity have been reported to vary from 2-8 cycles/degree. The response curve shown in Fig. 6.2 shows the peak at 2 cycles/degree. Here, the unit *cycles/degree* refers to a spatial sinusoidal wave whose wavelength makes an angle of one degree to the observers eye, at a specified distance between the eyes and the scene.

In Fig. 6.2, it is observed that HVS response is relatively poor at higher frequencies. It is important to know the maximum frequency which can be perceived by eyes. This can be calculated from the *visual acuity* of eye, which is defined as the smallest angular separation at which the individual lines (if we assume that the image consists of horizontal and vertical lines of pixels) can be distinguished. Visual acuity varies widely as much as from 0.5′ to 5′ (minutes of arc) depending on the contrast ratio and the keenness of the individual's vision [47]. An acuity of 1.7′ is often assumed for practical purposes and this corresponds to approximately 18 cycles/degree.

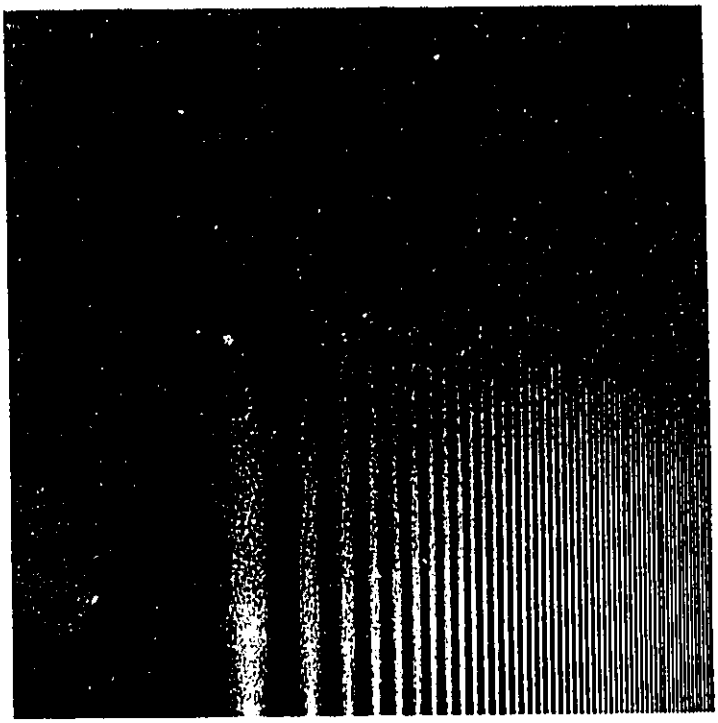


Figure 6.1 : An oscillating pattern for modelling HVS response [21]

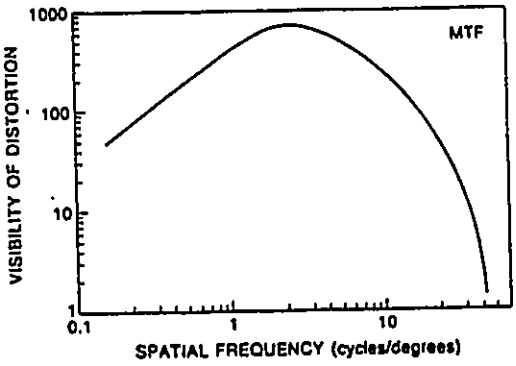


Figure 6.2 : Noise Visibility function of the human visual system [50]

Background Luminance

Noise sensitivity depends on the background luminance as specified by Weber's law [21, 50], *i.e.* the eye is less sensitive to noise in a brighter background. Thus, if the background luminance is high, more noise can be introduced without degrading the visual quality. Some simulations were carried out by Safranek *et al.* [89], to relate the noise sensitivity of eyes to the grey level of the background (in the sub-band domain). Fig. 6.3 shows a typical curve, where we see that the visibility of distortion is maximum at the mid grey-level range of the transform coefficients and decreases in both directions away from the mid-range.

Texture-masking

The frequency sensitivity and background luminance provide an account of HVS characteristics for a flat field image. Since, humans are more sensitive to noise in flat fields than in textured regions, texture-masking model improves the coding performance by adding more noise in the coefficients which represent a highly textured region in the original image [89].

Wavelet Transform and HVS

Traditionally, HVS model is assumed to be a combination of different visual channels which act as a bank of spatial-frequency filters. Hence, the visual system acts as a Fourier analyzer. The channel properties are given by the modulation transfer function. The response of a visual channel is given by the convolution of the excitation function and the impulse response of the HVS. However, the Fourier transform is defined through an integral which covers the whole spatial domain, whereas the retinal vision is always limited by a geometrical shape [33]. In addition, the visual pathways are found to be non-linear filters. Hence, Fourier transform can explain the visual process only in a limited way.

Marcelja has provided a mathematical description of the visual process using Gabor transform. Since Gabor transform uses a localized spatial window, it provides a good local description of the most important elements of the image. However, it has been reported that Gabor model cannot take into account the retinal automatic gain control satisfactorily and there is a lack of harmony between the visual analyzer and the stimulus profile [33].

Recently, HVS has been analyzed in the wavelet domain. It was shown that the visual system analyzes an image in a way that is similar to the multiresolution decomposition of the wavelet function [33, 62]. However, more work needs to be done before we can directly incorporate these results in image compression applications. In our analysis, we have used a Fourier model because of its simplicity. However, we expect that, in future when a good wavelet model of HVS will be available, it will improve the adaption of wavelet coefficients to HVS and hence will result in an appreciable improvement in coding performance.

6.3. Perceptual coding

Though most of the HVS models have been developed in Fourier domain, unfortunately, it is not very efficient in energy compaction. Traditionally, DCT has been used for its better performance and recently, wavelet/subband coding is being used at low bit-rate coding for its several advantages. The flowchart of an ideal wavelet based coding algorithm adapting HVS properties is shown in Fig. 6.4. We note that "H" in Fig. 6.4, is the actual transfer function of the HVS and not the noise visibility curve shown in Fig. 6.2. The main drawback of this ideal scheme is that it is very compute intensive. The price for adapting to HVS is the complexity of four separate 2-D Fourier transforms. In addition, Fourier transform produces a set of complex coefficients from the real image. Coding complex data will require more bits, resulting in a lower compression ratio.

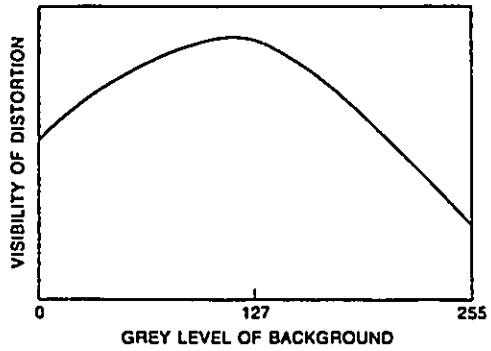


Figure 6.3 : Noise sensitivity versus background luminance [50]

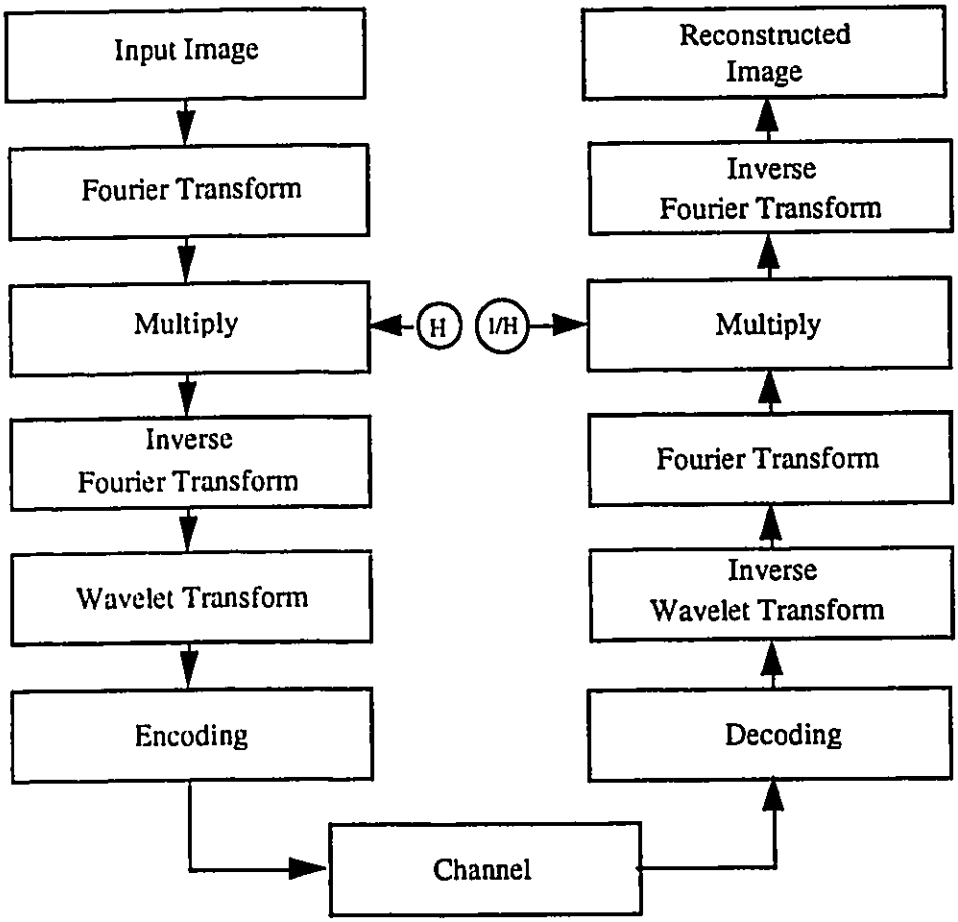


Figure 6.4 : A Wavelet Based Coding Scheme with Fourier-domain HVS Model

In this section, we provide a detailed discussion regarding how to choose the quantization step sizes in various bands of wavelet coefficients so that the results are subjectively better. The fact that the eyes are less sensitive at higher (and also at very low frequency) spatial frequencies can be exploited by allowing more noise in the high frequency coefficients. This can be done by using a frequency weighted MSE cirterion. With this scheme, we can use the MSE coder described in section 4.3 with little modification to adapt it to the HVS.

The objective of the MSE coder described in section 4.3 was to minimize the mean square error. An analogous situation for the perceptual coder is to minimize the perceptual noise which can be expressed as [70]

$$N_{perceptual} = \sum_{u=0}^{N-1}\sum_{v=0}^{M-1} NV(r)\left[T(u,v) - \hat{T}(u,v)\right]^2 \qquad (6.1)$$

$$= \sum_{u=0}^{N-1}\sum_{v=0}^{M-1} NV(r)[\Delta(u,v)]^2$$

where,

$N_{Perceptual}$ = Perceptual Noise

$NV(r)$ = Rotationally symmetric noise visibility of HVS with $r^2 = u^2 + v^2$

$T(u,v)$ = Fourier transform coefficients of the original image

$\hat{T}(u,v)$ = Quantized Fourier coefficients

$\Delta(u,v) = T(u,v) - \hat{T}(u,v)$, *i.e.* quantization error

M, N = Number of Fourier coefficients in orthogonal u, v directions

u, v = 2 dimensional spatial frequency space.

Modelling the noise according to Eq. (6.7) is attractive in the sense that, we are still working within the framework of MSE quantizers, the only difference being that the use of a frequency weighted noise power allocation algorithm.

In the Fourier domain, each coefficient corresponds to an exponential basis function (which has infinite resolution in frequency, but no resolution in time), which can be used directly in Eq. (6.1). However, for wavelet transform, any basis function represents a time-scale decomposition. Hence, the spectrum of a wavelet basis function spreads across the whole frequency axis. Hence, Eq. (6.1) will be difficult to use for wavelets. A more practical approach will be to find the relative visual importance of the basis functions of each band of wavelet coefficients depending on their spectral contents. Hence, we modify Eq. (6.1) as

$$N_{perceptual} = W_{c^k} N_{c^k} + \sum_{i=1}^{K} \sum_{j=h,v,d} W_{d^{i,j}} N_{d^{i,j}} \tag{6.2}$$

where,

$N_{Perceptual}$ = Perceptual noise

$W_{c^k}, W_{d^{i,j}}$ = Perceptual weights of (coarse and detail) various bands

$N_{c^k}, N_{d^{i,j}}$ = Total noise power of various bands

K = No. of decomposition level

In section 6.3.1, we will describe a simple coding scheme for HVS adaptation. This scheme, like most others, overlooks many important details of HVS characteristics, makes assumption which are not true in general, but nevertheless provides a good visual quality of the reconstructed image in most cases. In section 3.2, we will modify the image adapted optimal coding scheme of section 4.2.4 to include HVS characteristics. To achieve this objective, we first find the spectral response of various wavelet basis functions. Weighting these spectral response with the noise visibility curve, we find the relative importance of various bands. Using these results, we will be able to design a coder which is optimal in the weighted mean square sense. We conclude by comparing the relative performance of the two coding schemes.

6.3.1 A Simple Quantization Scheme

In Fig. 6.2, we see that the HVS response is maximum at 2 cycles/degree. We also observe that the slope of the curve beyond 2 cycles/degree, is approximately 2, in log-log scale. The slope of 2, is equivalent to

$$|H(f)|^2 = \frac{1}{f^2} \tag{6.3}$$

The above relation indicates that the noise visibility decreases with increasing frequency. If we double the frequency, the noise visibility decreases by a factor of 4. If we assume that most of the quantization noise is limited within the slope of the HVS response, it will simplify our calculation of quantization step sizes. The MSE coder can be modified to allocate 4 times more noise in the successive high pass bands. Accordingly, Eq. (4.4) can be written in L^2 -metric as

$$K_j^{1/2}\left\|(c_{j,k} - \hat{c}_{j,k})\right\|_{L^2} \le \frac{\varepsilon'}{\left\|\psi_{j,k}\right\|_{L^2}} \tag{6.4}$$

where, K_j is the perceptual weight of j-th band.

The quantization step size at j-th scale can then be written as

$$Q_j = \frac{2\varepsilon'}{K_j^{1/2}\left\|\Psi_{j,k}\right\|_{L^2}}$$

Or, in other words, the ratio of quantization step sizes between two successive scales will be

$$\frac{Q_{j+1}}{Q_j} = \frac{K_j^{1/2}\left\|\psi_{j,k}\right\|_{L^2}}{K_{j+1}^{1/2}\left\|\psi_{j+1,k}\right\|_{L^2}}$$

Since, $K_j = 4K_{j+1}$ and $\left\|\psi_{j,k}\right\|_{L^2} = \left\|\psi_{j+1,k}\right\|_{L^2}$ (this is true only for orthonormal coefficient set) we can express above relation as

$$\frac{Q_{j+1}}{Q_j} = 2 \tag{6.5}$$

Thus, we get a simple quantization scheme in which each successive higher resolution bands will be quantized with a double stepsize.

L^1 or L^2 Metric ?

Devore *et al.*, have suggested that the HVS characteristics can be easily incorporated by measuring the error in L^1 metric. They proposed to use an error metric - L^p, such that

$$\left\| c_j \psi_{j,k} \right\|_{L^p} = \left\| c_{j+1} \psi_{j+1,k} \right\|_{L^p} \tag{6.6}$$

where,

c_j = contrast threshold in j-th band

c_{j+1} = contrast threshold in $(j+1)$-th band

ψ_j = basis function of j-th scale

ψ_{j+1} = basis function of $(j+1)$-th scale

Since, the eye is less sensitive at higher frequencies, the contrast threshold at higher frequencies will be higher than that of lower frequencies, *i.e.*

$$c_{j+1} = 4\, c_j$$

Substituing the above in Eq. (6.6), we see that the following relation should hold

$$\left\| \psi_{j,k} \right\|_{L^p} = 4 \left\| \psi_{j+1,k} \right\|_{L^p} \tag{6.7}$$

Devore *et al.*, argued that if p=1, the relation (6.7) is satisfied and hence, the error should be measured in L^1 metric. However, the above argument does not seem to be correct. In [27], a specific implementation of wavelets (corresponding to a =1/2 of Table 4.1) was used and only in that special case, relation (6.7) is true. In other cases, different values of p, will satisfy Eq. (6.7). Though, the argument in [27] is not true in general, it is observed from Table-4.2 that measuring error in L^1 metric gives more weight to the low frequency bands by a factor of two (in the signal level). This is

equivalent of saying that in terms of noise power, it introduces four times more noise in each successive higher octave band than that is introduced by L^2 metric. Hence, measuring error in L^1 metric, will provide results close to weighted L^2 metric. However, they may not be exactly same and since the proposed L^1-metric was derived from weighted L^2 metric, the latter may be more general.

6.3.2 Proposed Optimal Coding Algorithm

The quantization scheme described in the previous section assumed that

i) The noise visibility decreases as frequency increases.

ii) The frequency bands associated with wavelet decomposition are ideal.

iii) The centre of mass of the frequency bands are related logarithmically.

The above assumptions are only approximations of the HVS properties. However, to get the optimum result, one should carefully include the details of the HVS curve. Note that the basis functions have energy across the whole frequency spectrum of the HVS response curve. In addition, they are not limited to the downward slope of the HVS curve. In this section, we will make a detailed study of the spectrum of the wavelet basis functions. It will be seen that the relative importance of the consecutive wavelet bands are not exactly in the ratio of 1:4. However, when we combine the rate-distortion characteristics and the HVS characteristics, the quantization step-sizes become very similar to that of the coding scheme of section 6.3.1 in most cases. However, our algorithm is more adapted to the image and is robust. In addition, it provides an analytical justification of the superior performance of the coding scheme of [27], even though the HVS response curve is much different from the actual curve reported in the literature.

6.3.2.1 Spatial Bandwidth of Wavelet Basis Functions

In this section, we will find the spatial bandwidth of various wavelet basis functions. The spatial bandwidth will be calculated in *cycles/degree*, as this is more appropriate

in visual signal processing. For a particular wavelet (say Daub-8), there will be various basis functions corresponding to different bands of decomposition. First, we calculate the centre frequency and bandwidth of various bands with dyadic tree decomposition, in one dimension. This gives an idea of the nature of frequency response of the wavelet basis functions. We then calculate the relative visual importance of the various bands for the two-dimensional case.

1-D Case

In practice, one works with the *refinement* coefficients. Hence, the basis functions of a discrete wavelet transform, can be thought as a sequence of impulse responses. The number of impulses for a basis function at tree depth i, is $(2^i-1)(L-1)+1$, where L is the order of the filter. The basis functions of Daub-8 wavelet are shown in Fig. 6.5 for 3 levels. Each basis function will consist of a band of frequencies which are shown in Fig. 6.6. The frequencies are expressed in radians. Note that the basis functions of Fig 6.5 and the power spectrum of Fig. 6.6 have been normalized for easy comparison and they correspond to each other only within a constant factor. Fig. 6.6 shows the frequency spread of basis functions of Daub-8 for three dyadic stages.

In order to express various parameters *e.g.* centre frequency, bandwidth *etc.*, in cycles/degree, we adopt a conversion procedure. We assume that the viewing distance is 6 times the picture height. At this viewing distance, the image will subtend an angle of 9.527 degrees ($=2\tan^{-1}(1/12)$) at the observer's eyes in any one dimension. Let us, assume that the image size is 512 x 512. With this image size, one degree will correspond to 54 (=512/9.527) pixels in the image. In other words, the wavelength of a spatial sinusoidal wave of frequency 1 cycles/degree will be 54 pixels, with the assumed viewing conditions.

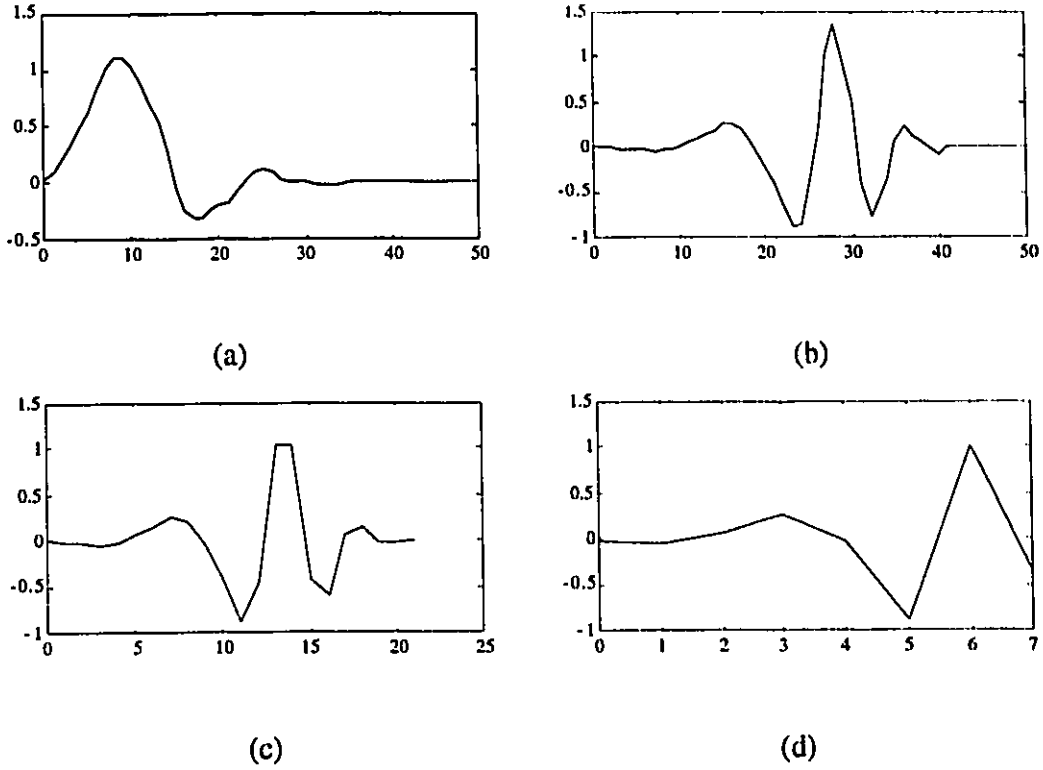


Figure 6.5 : Basis functions (in discrete sequence) of Daub-8 wavelet in dyadic tree. (a) 3rd level scaling function, (b), (c) and (d) are 3rd, 2nd and 1st level wavelet functions respectively.

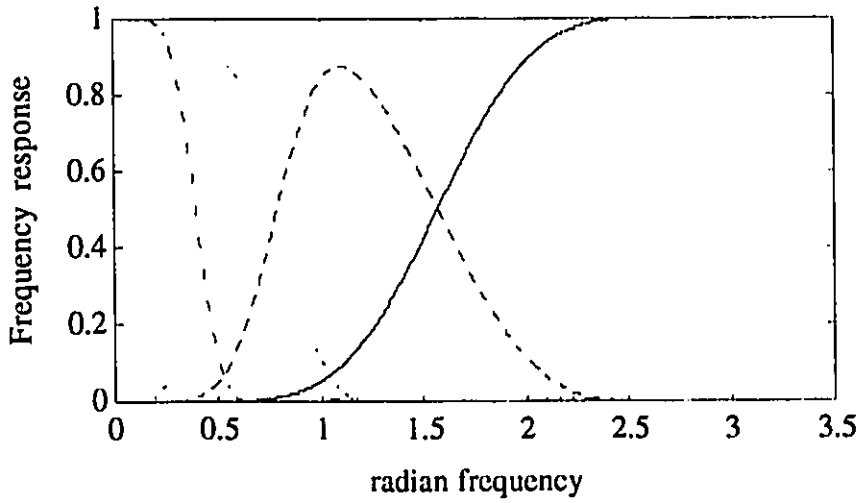


Figure 6.6 : Frequency response of one scaling and three wavelet functions of Daubechies-8 wavelet (for 3 stages)

The centre frequency and the bandwidth as defined in Eq. (5.4) have been calculated for Daub-4, Daub-8 and Daub-16. The results are provided in Table-6.4. The stage 1-w corresponds to the wavelet function of tree depth 1 and the stage 5-w and 5-s correspond to the wavelet and scaling functions of the tree depth 5, respectively. The following observations can be made from Table-6.4. First, the frequency spread of the basis functions cover from 0-25 cycles/degree *i.e.*, it covers the most useful range of the HVS. Secondly, the centre frequencies of various bands closely follow the octave nature of the transform. For example, the centre frequencies of Daub-16 are approximately 20, 10, 5, 2.5, 1.3 and 0.4 cycles/degree. However, the bandwidth of various basis functions do not follow the octave nature exactly. For Daub-4, we see that the bandwidth of wavelet function at different stages do not change logarithmically. This is due to the poor frequency localization of Daub-4 wavelet, despite its better time localization. For Daub-8, the frequency localization is better and for Daub-16, it is very good.

Table-6.2

Centre frequency ($\bar{f}$) and frequency spread (σ_f) of various wavelet bands

Stage	Daub-4		Daub-8		Daub-16	
	$\bar{f}$	σ_f	$\bar{f}$	σ_f	$\bar{f}$	σ_f
1-w[1]	19.55	4.84	19.9	4.39	20.05	4.14
2-w	10.84	3.56	10.59	3.00	10.37	2.57
3-w	5.92	3.31	5.38	1.89	5.19	1.31
4-w	3.07	2.26	2.70	1.03	2.60	0.66
5-w	1.56	1.39	1.35	0.53	1.30	0.33
5-s	0.51	0.69	0.44	0.31	0.42	0.26

[1] x-w corresponds to the wavelet function at tree depth x.

2-D Case

In Fig. 6.6, we observe that the frequency response of the various wavelet basis functions are far from ideal. Though, their energy is concentrated in some frequency region, appreciable energy is present elsewhere in the spectrum. This is also true in

the two- dimensional case. Thus, when estimating the relative importance of various bands, we cannot assume "brick wall" transition of various frequency bands. In addition, the HVS reponse curve is nonlinear. Hence, a better measure will be to integrate the power spectrum of each band weighted by $NV(r)$, the noise visibility function, in the whole two dimensional frequency space (which is limited by Nyquist frequencies). The relative importance of various bands for Daub-4, Daub-8 and Daub-16, are provided in Table 6.5 and Table 6.6. The entries in Table-6.5 and Table-6.6, provide the reciprocal of the visual importance of various bands of the wavelet transform for 4 and 3 stages of decomposition, respectively. From these Tables, it is observed that the visual importance of the consecutive bands are not in the ratio of 1:4:16 .... For lower resolution bands, the entries are very similar. In addition, for 4 stages of decomposition, we observe that the most important band is not the lowest pass band as is generally expected. The horizontal and vertical detail bands $d^{4,h}$, $d^{4,v}$ are more important to HVS than $d^{4,s}$. This may be surprising, but if we look at the basis functions, we will see that the bands $d^{4,h}$, $d^{4,v}$ contain some crucial (edge) information for producing image with good fidelity. Secondly, it is observed that the horizontal and vertical bands of each stage, are equally important. This is expected since we assumed an isotropic HVS model, which has no preferential direction (towards horizontal, vertical or diagonal). The weight of a diagonal band is usually less than that from the corresponding horizontal or vertical bands because its spectral centre of mass is at a higher frequency than that of the other two.

6.3.2.2 Coding Scheme

We will assume the same coding environment as that of section 4.2.4, *i.e.*, the distortion and rates in different bands are additive in nature. So the optimization problem is to find minimum perceptual distortion for a given bit-budget. This constrained optimization problem can be easily converted to an unconstrained problem

by relating rate and perceptual distortion. Thus, the Lagrangian cost function of Eq. (4.8) becomes

$$J(\lambda) = \frac{1}{K}.D + \lambda R \tag{6.14}$$

where K is the perceptual weight of each band.

Table - 6.3
Visual weights of various bands of Daub. wavelet
(4 stages of decomposition)

Bands[1]	Daub-4	Daub-8	Daub-16
$d^{4,s}$	1.15	1.2	1.21
$d^{4,h}, d^{4,v}$	1.0	1.0	1.0
$d^{4,d}$	1.14	1.14	1.14
$d^{3,h}, d^{3,v}$	1.39	1.37	1.35
$d^{3,d}$	1.83	1.79	1.75
$d^{2,h}, d^{2,v}$	2.72	2.77	2.79
$d^{2,d}$	4.41	4.34	4.32
$d^{1,h}, d^{1,v}$	8.89	10.0	10.34
$d^{1,d}$	23.70	27.0	29.41

[1] as defined in Fig. 3.14

Table - 6.4
Visual weights of various bands of Daub. wavelet
(3 stages of decomposition)

Bands	Daub-4	Daub-8	Daub-16
$d^{3,s}$	1.0	1.0	1.0
$d^{3,h}, d^{3,v}$	1.28	1.27	1.25
$d^{3,d}$	1.72	1.67	1.62
$d^{2,h}, d^{2,v}$	2.54	2.56	2.58
$d^{2,d}$	4.12	4.1	4.0
$d^{1,h}, d^{1,v}$	8.31	9.1	9.6
$d^{1,d}$	22.12	25.5	27.17

Table 6.5 and Table 6.6 provide the reciprocal of perceptual weights (1/K) of various bands of Daub-4, Daub-8 and Daub-16 wavelets. The rate-distortion function will be optimal in the weighted MSE sense, if all the individual bands operate at a constant

slope λ on their respective weighted rate-distortion plane. The overall rate-distortion (R-D) function can be obtained as follows.

1. Obtain the R-D function of the individual bands.
2. Choose a λ in the range $(0,-\infty)$.
3. Find the rate and distortion from the R-D function at that λ / K in all the bands, where, K is the perceptual weight.
4. The summation of all the rates and distortion will correspond to a point in the overall R-D function at slope λ.
5. Change the λ and repeat step 3 and step-4.

The above procedure provides an algorithm to obtain the overall R-D function. However, it does not provide a λ which will produce the desired bit-rate. The desired λ can be obtained by an iterative algorithm described in [80].

The above optimal approach does not assume anything about the coder or the quantizer. Hence, it can be used with any coding scheme. However, it assumes the convexity of the R-D function. For higher resolution bands, λ / K may be very large and before reaching to that slope, all coefficients may be discarded. Usually, at a high compression ratio few highest resolution bands will be totally discarded.

6.3.3 Simulation Results

Detailed simulations have been carried out using several test images. In all cases, the reconstructed images have better subjective quality compared to the image coded using a normal MSE coder [64]. Here, we will compare the performance of two coding schemes described in the previous sections. For the remainder of the chapter, we will refer the schemes of section 6.3.1 and 6.3.2 as scheme-1 and scheme-2, respectively. Fig. 6.7a and Fig. 6.7b show the Lena image coded at 0.23 *bpp* using scheme-1 and scheme-2, respectively. The image was coded using Daub-8 wavelet with 4 stages of decomposition. Table 6.7 shows the quantization step sizes of the various bands for both methods. It is seen that the 1st, 2nd and 3rd stage bands have very similar

quantization step sizes. The only appreciable difference in the step-sizes are seen in the low pass bands. In scheme-1, more weight has been given to the lowest band. However, in scheme-2, almost equal weights have been given to the few lowest pass bands. Since the overall bit-rate and distortion depend on all the bands, the difference in quantization step sizes in the lowest resolution bands will make little difference in the visual quality of the image. However, some impact is seen in Fig. 6.7. In Fig 6.7b, the edges are slightly better represented. Lena's chin is more smooth in Fig. 6.7b. In addition, the edges of the white bar in the top right hand are better represented in Fig. 6.7b. Table 6.7 also provides the mean perceptual noise of the reconstructed image. This has been calculated with respect to $d^{4,h}$ band. From Table 6.7, we can see that, scheme-2 provides smaller mean perceptual noise than scheme-1.

In Fig. 6.8, we compare the subjective quality of Barbara image, coded at 0.4 *bpp*. This image was coded using Daub-8, with 3 stages of decomposition. Table 6.8 shows

Table - 6.5
Quantization step-sizes of Lena image
coded at 0.23 bpp

Bands	Scheme-1	Scheme-2
$d^{4,s}$	10.8	31
$d^{4,h}$	21.8	31
$d^{4,v}$	21.8	32
$d^{4,d}$	21.8	36
$d^{3,h}$	43.5	37
$d^{3,v}$	43.5	41
$d^{3,d}$	43.5	49
$d^{2,h}$	87	66
$d^{2,v}$	87	81
$d^{2,d}$	87	99
$d^{1,h}$	174	160
$d^{1,v}$	174	160
$d^{1,d}$	174	discarded
Bit-rate	0.23 bpp	0.23 bpp
Mean Perceptual noise	16.5	15.26

(a)

(b)

Figure 6.7 : Lena, coded at 0.23 *bpp* using
a) scheme-1, b) scheme-2

(a)

(b)

Figure 6.8 : Barbara, coded at 0.4 *bpp* using
a) scheme-1, b) scheme-2

Table - 6.6
Quantization step-sizes of Barbara image coded at 0.4 *bpp*

Bands	Scheme-1	Scheme-2
$d^{3,s}$	17	29
$d^{3,h}$	35	37
$d^{3,v}$	35	40
$d^{3,d}$	35	46
$d^{2,h}$	71	59
$d^{2,v}$	71	68
$d^{2,d}$	71	83
$d^{1,h}$	142	120
$d^{1,v}$	142	discarded
$d^{1,d}$	142	discarded
Bit-rate	0.4 bpp	0.4 bpp

the quantization step sizes in both methods. We observe that except the 4-th stage scaling band, all other bands have much similar quantization step sizes. However, this little difference is reflected in the subjective quality of the reconstructed image. The texture of Barbara's pants is better represented in Fig. 6.8b. Also, the distortion at the edges of the right leg of the table is less visible in Fig. 6.8b.

6.4 Conclusion

Two coding schemes adapting the human visual system have been investigated. Traditional MSE coder provides equal weights to all the bands. Hence, it uses valuable bits to encode higher spatial frequency bands which are less important to HVS. Scheme-1 uses a quantization scheme which gives more weight to the low frequency scales and ignores the importance of the mid-frequency scales which are also very important to the HVS. A good balance is provided by the coding scheme-2, described in section 6.3.3, which gives appropriate weights to all the bands. This has been seen to provide better subjective quality for all the images.

Further improvements can be carried out using a better HVS model. Ironically, we are using the Fourier model to encode wavelet coefficients, whereas wavelet transform is

capable of modelling the HVS much better than Fourier transform. A wavelet model of HVS will enhance the capability of perceptual coding of wavelet coefficients. In addition, one can use the masking property of HVS, effect of background luminance *etc.*, to improve the coding performance.

Chapter 7

Conclusions and Future Work

In this thesis, we first presented a procedure for designing a wavelet based coder using scalar quantization. We then compared the performance of three known wavelet families. The wavelets were compared based on their coding gain, objective and subjective performance. It was shown that for most images, except the low activity images such as visualmtf and chest, *transform coding gain* corresponds well with the actual coding performance. It was also observed that the filter order should not in general exceed 12, with 8-12 taps generally providing the best time-frequency localization. Least asymmetric wavelets in general provide a better coding performance than the corresponding minimum phase wavelet. Hence, while searching for optimal wavelets of order higher than eight, least asymmetric Daubechies wavelet is a good starting point.

Spectral flatness is proposed as a measure to search for the "optimal" wavelets. It is observed that such a search is warranted only for low spectral flatness images. Otherwise, *Daubechies* wavelets or *AHQMF's* are close to the optimum (at least for low order) and not much improvement in performance can be expected by conducting the full search. Irregular tree decomposition has been seen to provide a better performance than both dyadic and regular tree. The degree of improvement using the irregular tree, is appreciably more than that can be achieved using optimal wavelets

(which are difficult to determine) over known standard wavelets. We also observe that complex symmetrical orthonormal wavelets, though having some interesting properties, are not competitive with the corresponding real wavelet for image compression.

We also proposed a quantization scheme for adapting to the human visual system. We calculated the relative visual importance of various bands from the spatial bandwidth of wavelet basis functions and frequency sensitivity of HVS. It is observed that the visual importance of the lowest passbands are very similar and the visual importance decreases exponentially for the highest passbands. Combining the rate-distortion functions of various bands with their visual importance, the optimum quantization step-sizes were calculated. It was observed that in most cases the optimum quantization step sizes are very similar to that achieved by the quantization scheme described in [27]. Our work provides an analytical justification of the superior performance of the quantization scheme of [27], even though the HVS frequency sensitivity curve considered, was far from the actual curves reported in the literature. However, our proposed model is more adaptive, robust and provides a better performance for a wide variety of images.

Future work

Regarding the choice of wavelets, further work can be carried out in many directions. First, the observations of this thesis are true for natural images. The same experiments can be done on video sequences. The statistics of the error images of a video sequence is widely different from the statistics of the natural images. Hence, we expect new observations in the case of video. Secondly, developing a fast algorithm for searching the optimal wavelet is crucial if it is to be implemented in practical applications. In this regard, more research should be done to understand the relative importance of the various filter properties of a wavelet.

Extensive research has been done to achieve optimal objective performance in wavelet based coding. However, little research has been done to tune the coder to adapt to the HVS. There is wide scope for improving the coding performance by exploiting the HVS properties such as noise masking, background luminance and frequency sensitivity of HVS. Modelling the HVS in wavelet domain may further improve the coding performance.

Recently, Moving Pictures Expert Group (MPEG) of International Standardization Organization (ISO) has initiated a standarization effort, known as MPEG-4, addressing generic audiovisual coding at very low bit-rates (upto 64 kbits/s) with applications in videotelephony, mobile audiovisual comminications, *etc.* As wavelets are known to provide superior coding performance over discrete cosine transform (DCT) at low bit-rate, it is expected that they will be incorporated in the new standard.

Wavelets are also being experimented extensively in video processing. They are being applied in video indexing, browsing, retrieval, *etc.* It can also be applied in general image processing applications, such as restoration, filtering and pattern recognition. Developing fast hardware architectures for calculating wavelet transform is also an area to be investigated.

Bibliography

[1] S. O. Aase and T. Ramstad, "Some Fundamental Experiments in Subband Coding of Images," *Proc. of SPIE*, Vol. 1605, 1991.

[2] N. Abramson, *Information Theory and Coding*, McGraw-Hill, New York, 1963. This text book contains the first reference to what was to become the method of arithmetic coding (pp. 61-62)

[3] A. N. Akansu and Y. Liu, "On Signal Decomposition Techniques," *Optical Engineering*, Vol. 30, No. 7, July 1991.

[4] A. N. Akansu and R. A. Haddad, *Multiresolution Signal Decomposition : Transform, Subbands and Wavelets,* Academic Press Inc., 1992.

[5] A. N. Akansu and H. Caglar, "A Measure of Aliasing Energy in Multiresolution Signal Decomposition," *Proc. of ICASSP*, 1992.

[6] J. B. Allen and L. R. Rabiner, "A Unified Approach to Short-Time Fourier Analysis and Synthesis," *Proc. of IEEE,* Vol. 65, pp. 1558-1564, 1977.

[7] M. Antonini, M. Barlaud, P. Matheiu and I. Daubechies, "Image Coding Using Wavelet Transform", *IEEE Transactions on Image Processing*, Vol.1, No.2, April 1992.

[8] J. Argast, M. Rampton, X. Qiu and T. Moon, "Image Compression with Wavelet Transform," *Proc. of SPIE*, Vol. 2094, pp. 1347-1356, 1993.

[9] M. R. Banham and B. J. Sullivan, "A Wavelet Transform Image Coding Technique with a Quadrature Structure", *Proc. of ICASSP*, 1992.

[10] H. J. Barnard, *Image and Video Coding using a Wavelet Decomposition*, Ph.D. Thesis, Delft University, 1994.

[11] P. Burt and E. Adelson, "The Laplacian Pyramid as a Compact Image Code", *IEEE Transactions on Communications*, Vol. 31, pp.532-540, April 1983.

[12] H. Caglar, Y. Liu and A. N. Akansu, "Statistically Optimized PR-QMF Design," *Proc. of SPIE*, Vol. 1605, 1991.

[13] J. Chen, S. Itoh and T. Hasimoto, "Scalar Quantization Noise Analysis and Optimal Bit Allocation for a Wavelet Pyramid Image Coding System," *ICCS/ISITA* 1992, Singapore.

[14] C. K. Chui, *An Introduction to Wavelets*, Academic Press, Inc., 1991

[15] C. K. Chui Series Editor, *Wavelets : A Tutorial in Theory and Applications*, Academic Press, Inc., 1992.

[16] R. J. Clarke, *Transform Coding of Images*, Academic Press Inc., 1985.

[17] A. Cohen and J. Froment, "Image Compression and Multiscale Approximation," in *Wavelets : Algorithms and Applications*, Y. Meyer Eds.

[18] A. Cohen, I. Daubechies and J. C. Feauveau, "Biorthogonal bases of Compactly Supported Wavelets," *Communications on Pure and Applied Mathematics*, Vol. XLV, pp. 485-560, 1992.

[19] R. R. Coifman and M. V. Wickerhauser, "Entropy Based Algorithms for Best Basis Selections," *IEEE Transactions on Information Theory*, Vol. 38, No. 2, pp. 713-718, March 1992.

[20] J. M. Combes *et al.*, Eds., *Wavelets, Time-Frequency Methods and Phase Space*, Springer-Verlag, 1989.

[21] T. N. Cornsweet, *Visual Perception*, Academic Press, 1970.

[22] I. Daubechies, "Orthonormal Bases for Compactly Supported Wavelets", *Communications of Pure and Applied Mathematics*, pp. 909-996, November 1988.

[23] I. Daubechies, "The Wavelet Transform, Time-frequency Localization and Signal Analysis", *IEEE Trans. on Information Theory*, Vol.36, No.5, pp.961-1005, May 1990.

[24] I. Daubechies, *Ten Lectures on Wavelets*, SIAM, Philadelphia, 1992.

[25] L. D. Davisson, "Rate-Distortion Theory and Applications", *Proceedings of the IEEE*, pp.156-164, July 1972.

[26] P. Desarte, B. Macq and D. T. M. Slock, "Signal Adapted Multiresolution Transform for Image Coding," *IEEE Transactions on Information Theory*, Vol. 38, No. 2, March 1992.

[27] R. A. Devore, B. Jawerth and B. J. Lucier, "Image Compression through Wavelet Coding," *IEEE Trans. on Information Theory*, Vol. 38, No. 2, pp. 719-746, March 1992.

[28] T. Ebrahimi and M. Kunt, "Application of an Optimally Locallized fast Wavelet Transform in Image Compression," *Proc. of ICASSP*, 1992.

[29] J. B. J. Fourier, "Theorie Analytique de la Chaleur," in *Oeuvres de Fourier, tome Premier*, G. Darboux, Ed., Paris, Gauthiers-Villars, 1888.

[30] D. Gabor, "Theory of Communications," *J. Inst. Elect. Eng.* (London), Vol. 93, No III, pp. 429-457, 1946.

[31] L. Gagnon, J. M. Lina and B. Goulard, "Application of Complex Daubechies Wavelets to Numerical Simulation of a Nonlinear Signal Propagation Model," Preprint, Univ. de Montreal, 1993.

[32] D. L. Gall and A. Tabatabai, "Subband Coding of Digital Images using Symmetric Short Kernel Filters and Arithmetic Coding Techniques," *Proc. of ICASSP*, 1988.

[33] L. Gaudart, J. Crebassa and J. P. Petrakian, "Wavelet Transform in Human Visual Channels," *Applied Optics*, Vol. 32, No. 22, August, 1993.

[34] R. C. Gonzales and P. Wintz, *Digital Image Processing*, Addison-Wesley, 2nd Edition, 1987.

[35] R. A. Gopinath, *Wavelets and Filter Banks - New Results and Application*, Ph.D Thesis, Rice University, 1993.

[36] T. Greiner, E. Eberle and M. Pandit, "Design of Signal-Matched Wavelets - Theory and Results," *Signal Processing VI : Theories and Applications*, 1992.

[37] A. Grossman and J. Morlet, "Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape," *SIAM Journal of Mathematical Analysis*, Vol. 15, pp. 723-736, 1984.

[38] A. Haar, "Zur Theorie der orthogonalen Funktionensysteme," *Math. Ann.*, Vol. 69, pp. 331-371, 1910.

[39] A. Habibi, "Future Trends in Image Coding," *Proc. of SPIE*, Vol. 1771, Applications of Digital Signal Processing XV, 1992.

[40] P. Haddad, M. Barlaud and P. Matheiu, "Optimization of Distortion-rate in Image Coding : An Application of Wavelet Packets," *Signal Processing VI : Theories and Applications*, 1992.

[41] R. A. Haddad, A. N. Akansu and A. Benyassine, "Time-Frequency Localization in Transforms, Subbands and Wavelets : A Critical Review," *Optical Engineering*, Vol. 32, No. 7, July 1993.

[42] F. Hartung and J. H. Husoy, "Wavelet and Subband Coding of Images - A Comparative Study," *SPIE International Symposium on Optical and Applied Science and Engineering*, San Diego, CA, July 1993.

[43] R. F. Hess and E. R. Howell, "The Threshold Contrast Sensitivity Function in Strabismic Amblyopia : Evidence for a Two Type classification," *Vision Research*, Vol. 17, 1977.

[44] F. Hlawatsch and G. F. Boudreaux Bartels, "Linear and Quadratic Time-Frequency Signal Representation," *IEEE SP Magazine*, April 1992.

[45] D. A. Huffman, "A Method for the Construction of Minimum-redundancy Codes", *Proceedings of the I.R.E.*, pp.1098-1101, September 1952.

[46] J. H. Husoy and S. O. Aase, "Image Subband Coding with Adaptive Filter Banks," *Proc. of SPIE*, Vol. 1818, VCIP-1992.

[47] A. Inglis, *Video Engineering*, McGraw-Hill, 1993.

[48] A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[49] N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*, Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

[50] N. S. Jayant, J. Johnston and R. Safranek, "Signal Compression based on Models of Human Perception," *Proc. of IEEE*, Vol. 81, No. 10, October 1993.

[51] Joint Photographics Experts Group, ISO/IEC/JTC1/SC2/WG8, "JPEG Draft International Standard", 1991.

[52] J. Katto and Y. Yasuda, "Performance Evaluation of Subband Coding and Optimization of its Filter Coefficients," *Proc. of SPIE*, Vol. 1818, 1992.

[53] J. Katto, K. Komatsu and Y. Yasuda, "Short Tap and Linear Phase PR Filter Banks for Subband Coding of Images," *Proc. of SPIE*, Vol. 1818, 1992.

[54] W. Lawton, "Applications of Complex Valued Wavelet Transform to Subband Decomposition," *IEEE Transactions on Signal Processing*, Vol. 41, No. 12, December 1993.

[55] A. S. Lewis and G. Knowles, "Image Compression using 2-D Wavelet Transforms", *IEEE Trans. on Image Processing*, vol.1, no.2, April 1992.

[56] J. O. Limb, "Distortion Criteria of the Human Viewer", *IEEE Transactions on Systems, Man and Cybernatics*, pp. 778-793, December 1979.

[57] Y. Linde, A. Buzo, R. Gray, "An Algorithm for Vector Quantizer Design", *IEEE Transactions on Communications*, pp.84-95, January 1980.

[58] F. X. J. Lukas and Z. L. Budrikis, "Picture Quality Prediction based on a Visual Model," *IEEE Transactions on Communications*, Vol. COM-30, pp. 1679-1692, July 1982.

[59] B. Macq, "Weighted Optimum Bit Allocations to Orthogonal Transforms for Picture Coding," *IEEE Journal of Selected Areas in Communications*, Vol. 10, No. 5, June 1992.

[60] B. Macq and J. Y. Mertes, "Optimization of Linear Multiresolution Transforms for Scene Adaptive Coding," *IEEE Transactions on Signal Processing*, Vol. 41, No. 12, December 1993.

[61] S. G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation", *IEEE Trans. on Pattern Anal. and Mach. Intel.*, Vol.11, No.7, July 1989.

[62] S. G. Mallat, "Multifrequency Channel Decomposition of Images and Wavelet Model," *IEEE Trans. on ASSP*, pp. 2091-2110, 1989.

[63] M. K. Mandal, S. Panchanathan and T. Aboulnasr, "Wavelets for Image Compression," *Proc. of IEEE International Symposium on Time-Frequency and Time-Scale Analysis*, Oct. 1994.

[64] M. K. Mandal, T. Aboulnasr and S. Panchanathan, "Wavelet-based Image Coding Using HVS Properties," submitted to *SPIE International Symposium on Optical Science, Engineering and Instrumentation*, July 1995.

[65] M. K. Mandal, S. Panchanathan and T. Aboulnasr, "Choice of Wavelets for Image Compression," submitted to *Canadian Workshop on Information Theory*, May 1995.

[66] J. L. Manos and D. J. Sakrison, "The Effects of a Visual Fidelity Criterion on the Encoding of Images," *IEEE Trans. on Information Theory*, Vol. IT-20, pp. 525-536, July 1974.

[67] D. Marr, *Vision*, Freeman, New York, 1982.

[68] Y. Meyer Editor, *Wavelets : Algorithms and Applications*, SIAM, Philadelphia, 1993.

[69] K. Nayebi, T. P. Barnwell and M. J. T. Smith, "Analysis-Synthesis Systems with Time Varying Filter Bank structure," *Proc. of ICASSP*, Vol. IV, pp 617-620, 1992.

[70] N. B. Nill, "A Visual Model Weighted Cosine Transform for Image Compression and Quality Assessment," *IEEE Trans. on Communications*, Vol. COM-33, No. 6, June 1985.

[71] J. E. Odegard, R. A. Gopinath and C. S. Burrus, "Optimal Wavelets for Signal Decomposition and the Existence of Scale Limited Signals," *Proc. of ICASSP*, part-IV, pp. 597-600, 1992.

[72] P. Onno and C. Guillemot, "Tradeoffs in the Design of Wavelet Filters for Image Compression," *Proc. of SPIE*, Vol. 2094, pp 1536-1547, 1993.

[73] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall, Englewood Cliffs, NJ, 1989.

[74] A. V. Oppenheim, A. S. Willsky and I. T. Young, *Signals and Systems*, Prentice Hall, 1983.

[75] A. Papoulis, *Signal Analysis*, McGraw Hill, 1977.

[76] R. Pasco, *Source Coding Algorithms for Fast Data Compression*, Ph.D Thesis, Stanford Univ., 1976.

[77] D. Pollen, "SU(2, F[Z,1/Z]) for F a subfield of C," *J. American Mathematical Society*, Vol. 3, pp. 611-624, July 1990.

[78] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical receipes in (C) : The Art of Scientific Computing*, Second Edition, Cambridge, 1992.

[79] M. Rabbani and P. W. Jones, *Digital Image Compression Techniques*, SPIE Optical Engineering Press, 1991.

[80] K. Ramachandran and M. Vetterli, "Best Wavelet Packet Bases in a Rate-Distortion Sense," *IEEE Trans. on Image Proc.*, Vol. 2, No. 2, April, 1993.

[81] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic Press, London, 1990.

[82] H. L. Resnikoff, "A Guide to Wavelets and Their Application," *Proc. of SPIE*, Vol. 1830, Curves and Surfaces in Computer Vision and Graphics III (1992).

[83] E. Reusens and T. Ebrahimi, "New Techniques for Subband/Wavelet Transform Coefficient Coding Applied to Still Image Compression," *Proc. of SPIE*, Vol. 1771, 1992.

[84] O. Rioul and M. Vetterli, "Wavelets and Signal Processing," *IEEE SP Magazine*, October 1991.

[85] O. Rioul and P. Duhamel, "Fast Algorithms for Discrete and Continuous Wavelet Transforms," *IEEE Transactions on Information Theory*, Vol. 38, No. 2, March 1992.

[86] O. Rioul, "Regular Wavelets : A Discrete-Time Approach," *IEEE Transactions on Signal Processing*, Vol. 41, No. 12, December 1993.

[87] J. J. Rissanen, "Generalized Kraft Inequality and Arithmetic Coding", *IBM J. Res. Dev.* 20, pp.198-203, May, 1976.

[88] M. B. Ruskai, G. Beylkin, R. Coifman, I. Daubechies, S. Mallat, Y. Meyer and L. Raphael, editors, *Wavelets and Their Applications*, Jones and Bartlett, Boston, MA 1992.

[89] R. J. Safranek and J. D. Johnston, "A Perceptually Tuned Subband Image Coder with Image Dependent Quantization and Post Quantization Data Compression," *Proc. of ICASSP*, pp. 1945-1948, 1989.

[90] J. A. Saghri, P. S. Cheatham and A. Habibi, "Image Quality Measure based on a Human Visual System Model", *Journal of Optical Engineering*, pp. 813-818, July 1989.

[91] D. J. Sakrision, "On the Role of the Human Observer and a Distortion Measure in Image Transmission," *IEEE Trans. on Communications*, Vol. COM-25, No-11, November 1977.

[92] C. E. Shannon, "Coding Theorems for a Discrete Source with a Fidelity Criterion", *IRE National Convention Record*, Part 4, pp.142-163, 1959.

[93] M. J. T. Smith and S. L. Eddins, "Analysis/Synthesis Techniques for Subband Image Coding," *IEEE Trans. on ASSP*, Vol. 38, No. 8, August 1990.

[94] D. Stanhill and Y. Y. Zeevi, "Finding the Optimal Wavelet Functions for Image Representation," *Proc. of SPIE*, Vol. 2094, pp. 1691-1700, 1993.

[95] G. Strang, "Wavelets and Dilation Equations : A Brief Introduction," *SIAM Review*, Vol. 31, No. 4, pp 614-627, December, 1989.

[96] D. Taubman and A. Zakhor, "A Multistart Algorithm for Signal Adaptive Subband Systems," *Proc. of ICASSP*, 1992.

[97] A. H. Tewfik, D. Sinha and P. Jorgensen, "On the Optimal Choice of a Wavelet for Signal Representation," *IEEE Trans. on Information Theory*, Vol. 38, No. 2, March 1992.

[98] M. Unser, "On the Optimality of Ideal Filters for Pyramid and Wavelet Signal Approximation," *IEEE Trans. on Signal Processing*, Vol. 41, No. 12, Dec. 1993.

[99] P. P. Vaidyanathan, "Multirate Digital Filters, Filter banks, Polyphase Networks, and Applications : A Tutorial," *Proc. of the IEEE*, Vol. 78, No. 1, January 1990.

[100] P. P. Vaidyanathan, *Multirate Systems and Filter banks*, Prentice Hall, 1993.

[101] L. Vanderpole, "Hierarchical Transform and Subband Coding of Video Signals," *Signal Processing : Image Communications* 4 (1992).

[102] M. Vetterli and C. Harley, "Wavelets and Filter Banks : Theory and Design," *IEEE Trans. on Signal Processing*, September, 1992.

[103] E. Walch and E. Karnin, "A Fractal Based Approach to Image Compression," *Proc. of ICASSP*, pp 529-532, 1986.

[104] G. K. Wallace, "The JPEG Still Picture Compression Standard", *Comunications of the ACM*, vol.34, no.4, pp.30-45, April 1991.

[105] Xiping Wang and S. Panchanthan, "Wavelet Transform Coding using NIVQ", *Visual Communications and Image Processing '93, SPIE*, Vol.2094, Part 2, pp.999-1009, Cambridge, Massachussetts, Novemeber 1993.

[106] R. O. Wells, Jr., "Parameterizing Smooth Compactly Supported Wavelets," *Trans. of the American Mathematical Society*, 1993.

[107] M. V. Wickerhauser, "High Resolution Still Picture Compression," *Digital Signal Processing* 2, 204-226 (1992).

[108] R. Wilson, A. D. Calway and E. R. S. Pearson, "A Generalized Wavelet Transform for Fourier analysis : The Multiresolution Fourier Transform and Its Application to Image and Audio Signal Analysis," *IEEE Transactions on Information Theory*, Vol. 38, No. 2, March 1992.

[109] I. H. Witten, R. M. Neal and J. G. Cleary, "Arithmetic Coding for Data Compression,", *Communications of the ACM*, Vol 30, pp 520-540, June, 1987.

[110] J. W. Woods Eds, *Subband Coding of Images*, Kluwer, 1991.

[111] J. W. Woods and T. Naveen, "A Filter Based Bit-Allocation Scheme for Subband Compression of HDTV," *IEEE Trans. on Image Proc.*, Vol. 3, No. 3, pp. 436-440, July 1992.

[112] Z. Xiong, N. P. Galatsanos and M. T. Orchard, "Marginal Analysis Prioritization for Image Compression on a Hierarchical Wavelet Decomposition," *Proc. of ICASSP*, 1993.

[113] W. R. Zettler, J. Huffman and D. C. P. Linden, "Application of Compactly Supported Wavelets to Image Compression," *Proc. of SPIE*, Vol. 1224, Image Processing Algorithms and Techniques, pp. 150-160, (1990).

[114] H. Zou and A. H. Tewfik, "Parameterization of Compactly Supported Orthonormal Wavelets," *IEEE Trans. on Signal Processing*, Vol. 41, No. 3, March 1993.

Appendix A

In this appendix, we summarize some of the important formulae relating wavelet transform, which have been discussed in Chapter 3.

The following notations have been used.

$x(t) \leftrightarrow X(\omega)$	: Fourier transform pair
$\phi(t)$	: Scaling function
$\psi(t)$	: Wavelet function
h[n]	: Lowpass filter coefficients
g[n]	: Highpass filter coefficients

Table-A.1

Comparison of Fourier Transform and Wavelet Transform

Fourier Transform	Wavelet Transform
$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$	$CWT_x(a,b) = a^{-1/2}\int_{-\infty}^{\infty} x(t)\psi^*(\frac{t-b}{a})dt$
$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}d\omega$	$x(t) = \frac{1}{C_\psi}\int_{-\infty}^{\infty}\int_{0}^{\infty} CWT_x(a,b)a^{-1/2}\psi(\frac{t-b}{a})\frac{dadb}{a^2}$
$X(n) = \frac{\omega_0}{2\pi}\int_{0}^{2\pi/\omega_0} x(t)e^{-jn\omega_0 t}dt$	$CWT_x(j,k) = \int_{-\infty}^{\infty} a_0^{-j/2}\psi(a_0^{-j}t - kb_0)x(t)dt$
$x(t) = \sum_{n=-\infty}^{\infty} X(n)e^{jn\omega_0 t}$	$x(t) = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} CWT(j,k)a_0^{-j/2}\psi(a_0^{-j/2} - kb_0)$

Table-A.2

Some Formulae relating Scaling and Wavelet Function

Time Domain	Frequency Domain
$\phi(t)=\sum_k h_k\phi(2t-k)$	$\Phi(\omega)=\frac{1}{\sqrt{2}}H(\omega/2)\Phi(\omega/2)$
$\psi(t)=\sum_k g_k\phi(2t-k)$	$\psi(\omega)=\frac{1}{\sqrt{2}}G(\omega/2)\phi(\omega/2)$
$\langle\phi(t),\phi(t-k)\rangle=\delta_{0,k}$	$\sum_k\lvert\Phi(\omega+2k\pi)\rvert^2=1$
$\sum_n h[n]=\sqrt{2}$	$H(0)=\sqrt{2}$
$g[n]=(-1)^n h[N-1-n]$	$G(\omega)=-e^{-j\omega(N-1)}H(-\omega+\pi)$
$\sum_n h[n]h[n-2k]=\delta_{0,k}$	$\lvert H(\omega)\rvert^2+\lvert H(\omega+\pi)\rvert^2=2$
$\sum_n g[n]g[n-2k]=\delta_{0,k}$	$\lvert G(\omega)\rvert^2+\lvert G(\omega+\pi)\rvert^2=2$
$\sum_n h[n]g[n-2k]=0$	$H(-\omega)G(\omega)+H(-\omega+\pi)G(\omega+\pi)=0$
$\sum_n (h[n-2k]h[p-2k]+g[n-2k]g[p-2k])=\delta_{p,k}$	$\lvert H(\omega)\rvert^2+\lvert G(\omega)\rvert^2=2$

Appendix B

The parameterization of orthonormal wavelets has been discussed in section 3.5. There we provided the 4 tap and 6 tap Pollen's parameterization [77] and 4 tap Zou's parameterization [114] formula. Here we provide 6 tap and 8 tap parameterization of Zou's. The range of the parameters are $[0, \pi]$

6 tap wavelets

Two free parameters $\alpha, \beta \in [0, \pi]$

The refinement coefficients are given by

$$
\begin{aligned}
h_0 &= \frac{1}{\sqrt{2}}[1 - A + B + C - D] \\
h_1 &= \frac{1}{\sqrt{2}}[1 - A + B - C + D] \\
h_2 &= \frac{1}{\sqrt{2}}[A - 2B - C + 2D] \\
h_3 &= \frac{1}{\sqrt{2}}[A - 2B + C - 2D] \\
h_4 &= \frac{1}{\sqrt{2}}[B - D] \\
h_5 &= \frac{1}{\sqrt{2}}[B + D]
\end{aligned}
\tag{B.1}
$$

where,

$$
\begin{aligned}
A &= \cos^2(\alpha) + \cos^2(\beta) \\
B &= \cos(\alpha)\cos(\beta)\cos(\alpha - \beta) \\
C &= \sin(\alpha)\cos(\alpha) + \sin(\beta)\cos(\beta) \\
D &= \sin(\alpha)\cos(\beta)\cos(\alpha - \beta)
\end{aligned}
$$

8 tap wavelets

Three free parameters $\alpha, \beta, \gamma \in [0, \pi]$

The refinement coefficients are given by

$$
\begin{aligned}
h_0 &= \frac{1}{\sqrt{2}}[1 - A + B - C + D - E + F] \\
h_1 &= \frac{1}{\sqrt{2}}[1 - A + B - C - D + E - F] \\
h_2 &= \frac{1}{\sqrt{2}}[A - 2B + 3C - D + 2E - 3F] \\
h_3 &= \frac{1}{\sqrt{2}}[A - 2B + 3C + D - 2E + 3F] \\
h_4 &= \frac{1}{\sqrt{2}}[B - 3C - E + 3F] \\
h_5 &= \frac{1}{\sqrt{2}}[B - 3C + E - 3F] \\
h_6 &= \frac{1}{\sqrt{2}}[C - F] \\
h_7 &= \frac{1}{\sqrt{2}}[C + F]
\end{aligned}
\tag{B.2}
$$

where,

$$A = \cos^2(\alpha) + \cos^2(\beta) + \cos^2(\gamma)$$

$$B = \cos^2(\gamma)[\cos^2(\alpha) + \cos^2(\beta)] + \cos(\alpha)\cos(\beta)\cos(\alpha - \beta) + \frac{1}{2}\sin(\gamma)\cos(\gamma)[\sin(2\beta) + \sin(2\alpha)]$$

$$C = \cos(\alpha)\cos(\gamma)\cos(\alpha - \beta)\cos(\beta - \gamma)$$

$$D = \frac{1}{2}[\sin(2\alpha) + \sin(2\beta) + \sin(2\gamma)]$$

$$E = \frac{1}{2}\left[\sin(2\alpha)\{\cos^2(\beta) + \cos^2(\gamma)\} + \sin(2\beta)\{\sin^2(\alpha) + \cos^2(\gamma)\}\right] + \frac{1}{2}\sin(2\gamma)\{\sin^2(\alpha) + \sin^2(\beta)\}$$

$$F = \sin(\alpha)\cos(\gamma)\cos(\alpha - \beta)\cos(\beta - \gamma)$$

Appendix - C

It is known that increasing vanishing moments of h[n] (the refinement coefficients) and $\phi(t)$, increases the regularity or differentiability of $\phi(t)$. Here, we will show that if $H(z)$ has L zeros at $z=-1$ (i.e. at $\omega=\pi$) and of order N, then

i) The low pass filter has L vanishing derivatives at $\omega=\pi$.

ii) The high pass filter has L vanishing derivatives at $\omega=0$.

iii) The high pass filter coefficients has L vanishing moments.

iv) the scaling function decays as fast as $\left[\sin c\left(\frac{\omega}{2}\right)\right]^L$

Proof [4]

If $H(z)$ has L zeros at $z=-1$, we can write it as,

$$H(z)=\left(\frac{1+z^{-1}}{2}\right)^L P(\omega) \tag{C.1}$$

Putting $z=e^{j\omega}$, we get

$$\begin{aligned} H(e^{j\omega}) &= \left(\frac{1+e^{-j\omega}}{2}\right)^L P(\omega) \\ &= e^{-j\omega L/2}\left(\cos\left(\frac{\omega}{2}\right)\right)^L P(\omega) \\ &= \sum_n h[n]e^{-j\omega n} \end{aligned}$$

Taking derivatives of the above equations for m times, we get

$$\frac{d^m H(e^{j\omega})}{d\omega^m}=\left(\cos\left(\frac{\omega}{2}\right)\right)^{L-m} p_m(\omega)=\sum_n (-jn)^m h[n]e^{-j\omega n} \tag{C.2}$$

This results,

$$\left.\frac{d^m H(e^{j\omega})}{d\omega^m}\right|_{\omega=\pi}=(-j)^m\sum_n n^m(-1)^n h[n]=0, \qquad \text{m=0, 1, ...L-1}$$

(C.3)

So, we see that, the derivative of has L vanishing derivative at $\omega = \pi$. We also see that, the *refinement* coefficients have L vanishing moments, i.e.

$$\sum_n n^m (-1)^n h[n] = 0, \qquad \text{m=0, 1, ...L-1} \tag{C.4}$$

From Eq. (3.35), we see that if H(z) has L zeros at $\omega = \pi$, G(z) will have L zeros at $\omega = 0$. That means, we can write,

$$\begin{aligned} G(e^{j\omega}) &= \left(\frac{1-e^{-j\omega}}{2}\right)^L Q(\omega) \\ &= \left(\sin\left(\frac{\omega}{2}\right)\right)^L Q'(\omega) \\ &= \sum_n g[n] e^{-j\omega n} \end{aligned}$$

Taking derivatives of the above equations for m times, we get

$$\frac{d^m G(e^{j\omega})}{d\omega^m} = \left(\sin\left(\frac{\omega}{2}\right)\right)^{L-m} q'_m(\omega) = \sum_n (-jn)^m g[n] e^{-j\omega n} \tag{C.5}$$

This results,

$$\left.\frac{d^m G(e^{j\omega})}{d\omega^m}\right|_{\omega=0} = (-j)^m \sum_n n^m g[n] = 0, \qquad \text{m=0, 1, ...L-1} \tag{C.6}$$

So, we see that the high pass filter $G(e^{j\omega})$ has also L vanishing derivative at $\omega = 0$. From the above equation, we can also conclude that,

$$\sum_n n^m g[n] = 0, \qquad \text{m=0, 1, ...L-1} \tag{C.7}$$

Now we will show that the scaling functions generated by h[n], decays as fast as $\left[\sin c\left(\frac{\omega}{2}\right)\right]^L$

From (C.1), we can write,

$$\left|H(e^{j\omega})\right| = \left|\cos\left(\frac{\omega}{2}\right)\right|^L \left|P(e^{j\omega})\right|$$

(C.8)

Substituting this in Eq. (3.22), we get

$$|\phi(\omega)| = \prod_{k=1}^{\infty} \left| \left(\cos \frac{\omega}{2^{k+1}} \right) \right|^L \prod_{k=1}^{\infty} \left| \frac{1}{\sqrt{2}} P\left(e^{\frac{j\omega}{2^k}} \right) \right|$$

Using the relation, $\sin(\omega) = 2\sin(\omega/2)\cos(\omega/2)$, we can write in another way,

$$|\phi(\omega)| = \left| \frac{\sin(\omega/2)}{\omega/2} \right|^L \prod_{k=1}^{\infty} \left| \frac{1}{\sqrt{2}} P\left(e^{\frac{j\omega}{2^k}} \right) \right| \tag{C.9}$$

So, we see that decays as fast as $\left[\sin c\left(\frac{\omega}{2} \right) \right]^L$, provided $\prod_{k=1}^{\infty} \left| \frac{1}{\sqrt{2}} P\left(e^{\frac{j\omega}{2^k}} \right) \right|$ is bounded.

Appendix-D

Here, we provide the scaling (refinement) coefficients of Daubechies wavelets, Coiflets, AHQMF of various orders which have been used in this thesis. Here, N refers to the number of taps, L refers to the number of vanishing moments of the wavelet. All the filter coefficients are normalized so that, $\sum h_n = \sqrt{2}$

Table D.1

Minimum Phase Daubechies Wavelet for N = 4, 6, 8, 10, 12, 16, 20 taps
[Ref 24, Table 6.1].

	n	h[n]
N=4 L=2	0	0.482962913144
	1	0.836516303737
	2	0.224143868042
	3	-0.129409522551
N=6 L=3	0	0.332670552950
	1	0.806891509311
	2	0.459877502118
	3	-0.135011020010
	4	-0.085441273882
	5	0.035226291885
N=8 L=4	0	0.230377813308
	1	0.714846570552
	2	0.630880767939
	3	-0.027983769416
	4	-0.18703481171
	5	0.030841381835
	6	0.032883011666
	7	-0.010597401785
N=10 L=5	0	0.160102397974
	1	0.603829269797
	2	0.724308528437
	3	0.138428145901
	4	-0.242294887066
	5	-0.032244869584
	6	0.077571493840
	7	-0.006241490212
	8	-0.012580751999
	9	0.003335725285

	n	h[n]
N=12 L=6	0	0.111540743350
	1	0.494623890398
	2	0.751133908021
	3	0.315250351709
	4	-0.226264693965
	5	-0.129766867567
	6	0.097501605587
	7	0.027522865530
	8	-0.031582039318
	9	0.000553842201
	10	0.004777257511
	11	-0.001077301085
N=16 L=8	0	0.054415842243
	1	0.312871590914
	2	0.675630736297
	3	0.585354683654
	4	-0.015829105256
	5	-0.284015542961
	6	0.000472484573
	7	0.128747426620
	8	-0.017369301001
	9	-0.044088253930
	10	0.013981027917
	11	0.008746094047
	12	-0.004870352993
	13	-0.000391740373
	14	0.000675449406
	15	-0.000117476784

Table D.1
(continued)

	n	h[n]
N=20 L=10	0	0.026670057900
	1	0.188176800077
	2	0.527201188931
	3	0.688459039453
	4	0.281172343660
	5	-0.249846424327
	6	-0.195946274377
	7	0.127369340335
	8	0.093057364603
	9	-0.071394147166
	10	-0.029457536821
	11	0.033212674059
	12	0.003606553566
	13	-0.010733175483
	14	0.001395351747
	15	0.001992405295
	16	-0.000685856694
	17	-0.000116466855
	18	0.000093588670
	19	-0.000013264202

Table D.2
Least asymmetric Daubechies Wavelet
(Ref. 24, Table 6.3)

	n	h[n]
N=8 L=4	0	-0.075765713783
	1	-0.029635529211
	2	0.497618659383
	3	0.803738752112
	4	0.297857812795
	5	-0.099219531725
	6	-0.012603969093
	7	0.032223098127
N=10 L=5	0	0.027333068345
	1	0.029519490926
	2	-0.039134249302
	3	0.199397533976
	4	0.723407690403
	5	0.633978963456
	6	0.016602105764
	7	-0.175328089908
	8	-0.021101834024
	9	0.019538882735
N=12 L=6	0	0.015404109327
	1	0.003490712084
	2	-0.117990111148
	3	-0.048311742586
	4	0.491055941927
	5	0.787641141028
	6	0.337929421728
	7	-0.072637522786
	8	-0.021060292512
	9	0.044724901770
	10	0.00176771184
	11	-0.007800708324

Table D.3

Coiflets for N = 6, 12, 18, 24 taps [Ref 24, Table 8.1]

	n	h[n]
N=6 L=2	0	-0.072732619512
	1	0.337897662457
	2	0.852572020212
	3	0.384864846864
	4	-0.072732965112
	5	-0.015655728135
N=12 L=4	0	0.016387336463
	1	-0.041464936781
	2	-0.067372554722
	3	0.386110066823
	4	0.812723635449
	5	0.417005184423
	6	-0.076488599078
	7	-0.059434418646
	8	0.023680171946
	9	0.005611434819
	10	-0.001823208870
	11	-0.000720549643
N=18 L=6	0	-0.003793512864
	1	0.007782596427
	2	0.023452696141
	3	-0.065771911281
	4	-0.061123390002
	5	0.405176902409
	6	0.793777222625
	7	0.428483476377
	8	-0.071799821619
	9	-0.082301927106
	10	0.034555027573
	11	0.015880544863
	12	-0.009007976136
	13	-0.002574517688
	14	0.001117518770
	15	0.000466216960
	16	-0.000070983303
	17	-0.000034599772

	n	h[n]
N=24 L=8	0	0.000892313668
	1	-0.001629492012
	2	-0.007346166327
	3	0.016068943964
	4	0.026682300156
	5	-0.081266699680
	6	-0.056077313316
	7	0.415308407030
	8	0.782238930920
	9	0.434386056491
	10	-0.066627474263
	11	-0.096220442033
	12	0.039334427123
	13	0.025082261844
	14	-0.015211731527
	15	-0.005658286686
	16	0.003751436157
	17	0.001266561929
	18	-0.000589020756
	19	-0.000259974552
	20	0.000062339034
	21	0.000031229875
	22	-0.000003259680
	23	-0.000001784985

Table D.4

A set of optimal PR-QMF [Ref-4, Table-4.11]

	n	h[n]
N=4	0	0.466675669
	1	0.840588657
	2	0.240431112
	3	-0.133481875
N=6	0	0.442766931
	1	0.805049213
	2	0.352529377
	3	-0.146445561
	4	-0.088189527
	5	0.048503129
N=12	0	0.360838504
	1	0.744306049
	2	0.490757098
	3	-0.036047928
	4	-0.222383198
	5	-0.005408341
	6	0.128127832
	7	0.000007678
	8	-0.079675397
	9	0.018522733
	10	0.029441941
	11	-0.014273411
N=16	0	0.349996497
	1	0.731063819
	2	0.505852096
	3	-0.010803415
	4	-0.229358399
	5	-0.029975411
	6	0.134362313
	7	0.026991307
	8	-0.089102151
	9	-0.017502278
	10	0.062860841
	11	0.006564367
	12	-0.045242724
	13	0.009260600
	14	0.017738308
	15	-0.008492207

Table D.5

Complex symmetrical Daubechies Wavelet [31]

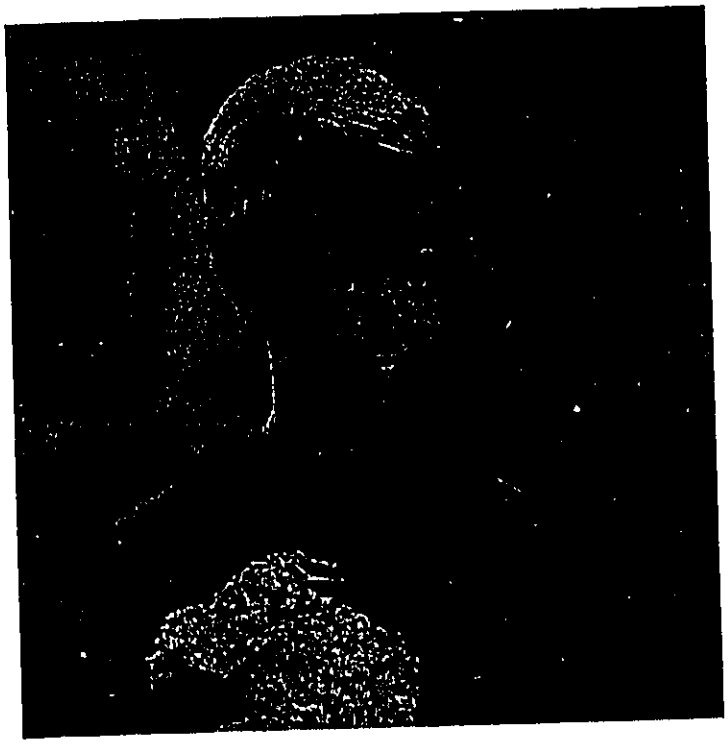
	n	h[n]
Real	0	0.010492450
	1	-0.017128908
	2	-0.080639704
	3	0.151379708
	4	0.643003234
	5	0.643003234
	6	0.151379708
	7	-0.080639704
	8	-0.017128908
	9	0.010492450
Imag.	0	0.020590437
	1	-0.008728528
	2	0.117947473
	3	0.094223656
	4	-0.182852164
	5	-0.182852164
	6	0.094223656
	7	0.117947473
	8	-0.008728528
	9	0.020590437

Appendix - E

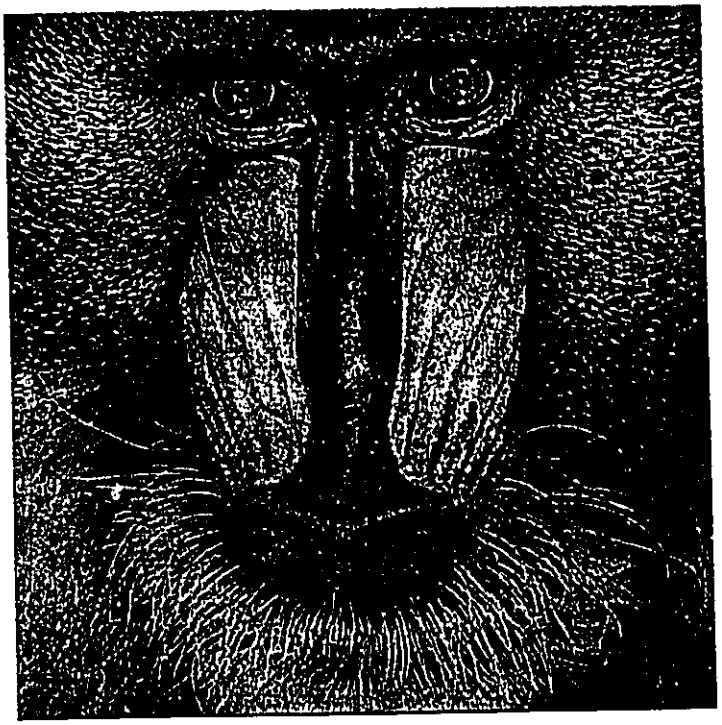
In this appendix, we show the test images used in this thesis. The test image "Visualmtf" has been shown in Fig. 6.1 and hence not shown here.

Lena (512x512)

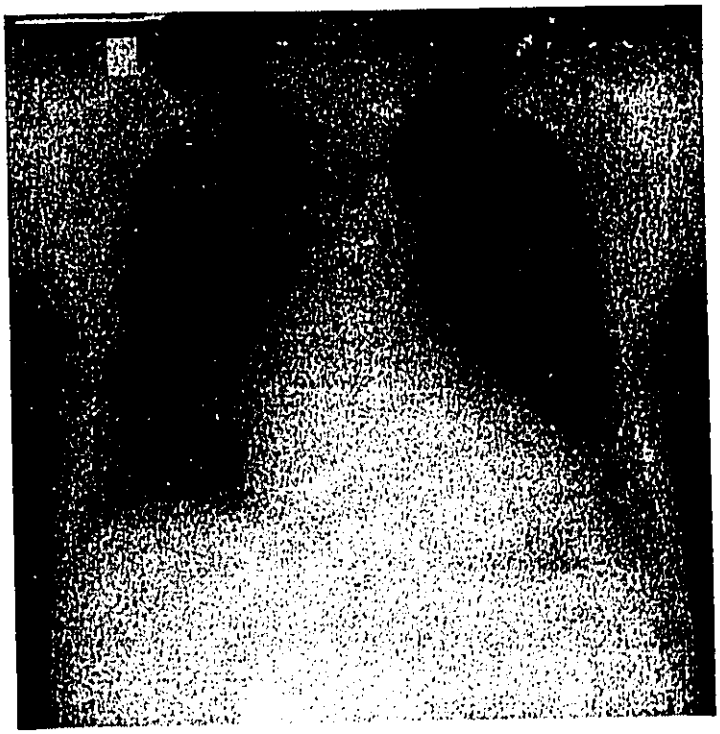
Barbara (512x512)



Girl (512 x 512)

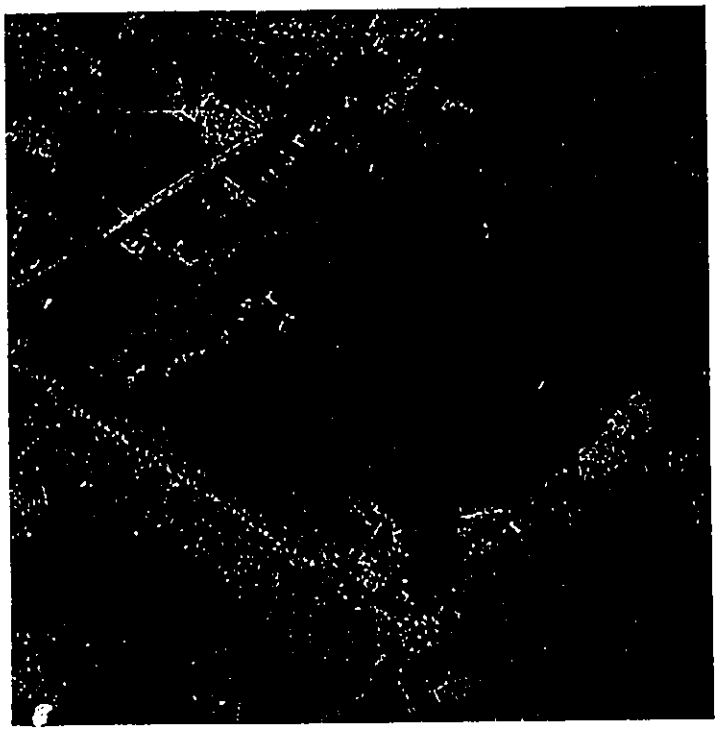


Mandrill (512x512)

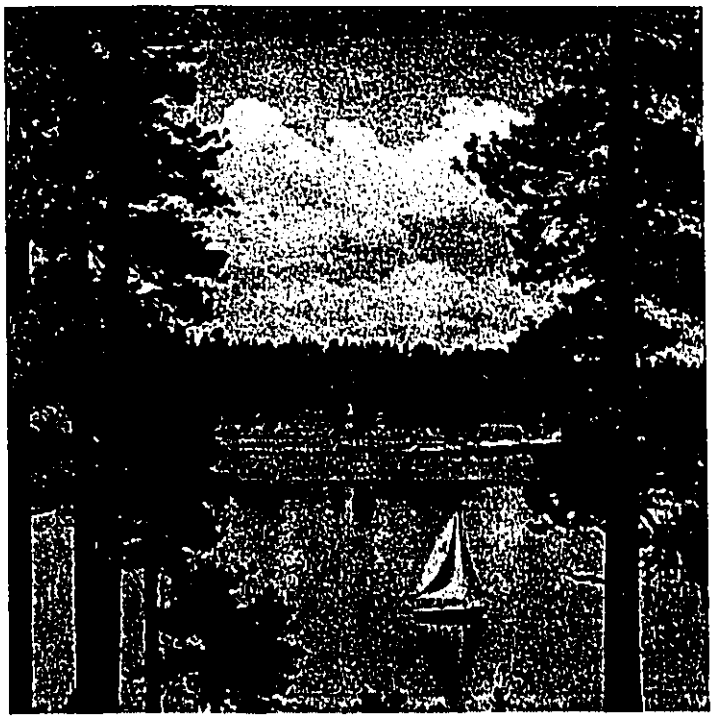


Chest (512x512)

Airplane (512 x 512)



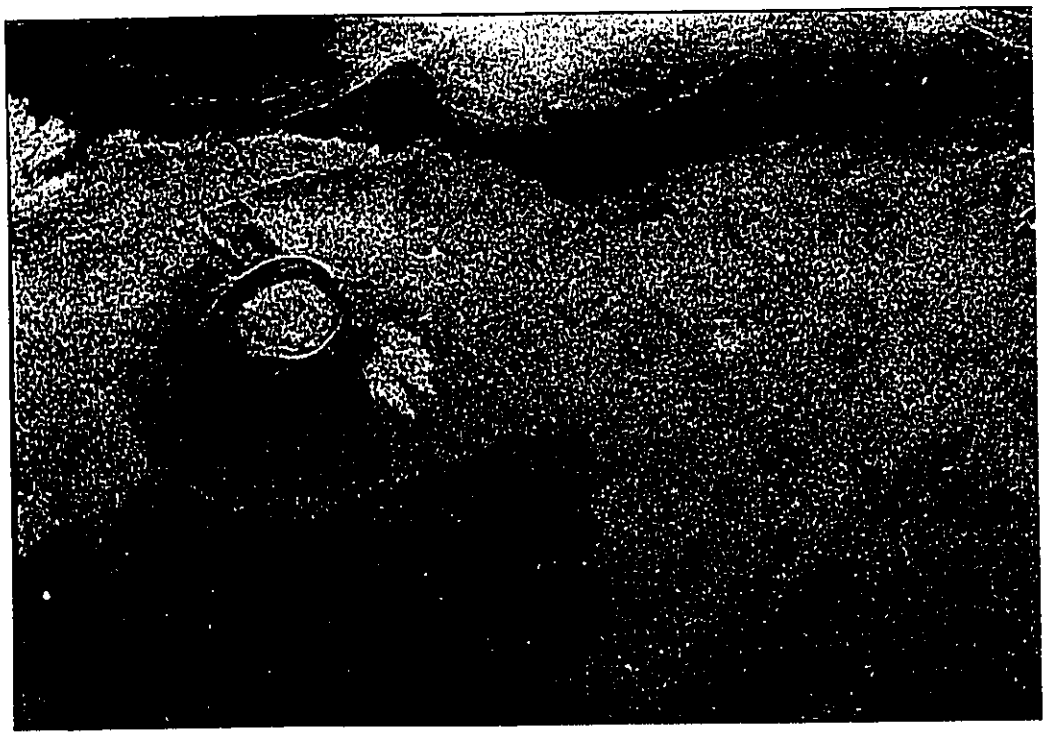
Airport (512 x 512)



Sailboat (512 x 512)

Safsaf Oasis (512x512)

Death Valley, California (480x768)



Galapagos Island (480x704)