

LOSSLESS COMPRESSION OF MEDICAL IMAGES

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL
AND ELECTRONICS ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Behiç Fırat KILIÇ
July 1991

TH 515
TA
1637
K55
1991

TA
1634
•K55
1981

B. 8653

LOSSLESS COMPRESSION OF MEDICAL IMAGES

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND
ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF
MASTER OF SCIENCE

By

Behiç Fırat Kılıç

July 1991

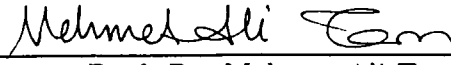
Behiç Fırat Kılıç
Electrical Engineering

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



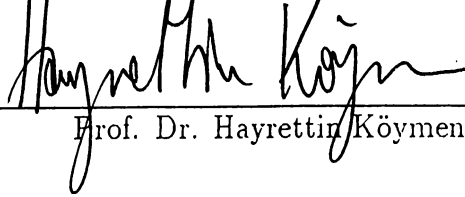
Assoc. Prof. Dr. Ahmet Enis Çetin(Principal Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



Assoc. Prof. Dr. Mehmet Ali Tan

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



Prof. Dr. Hayrettin Köymen

Approved for the Institute of Engineering and Sciences:



Prof. Dr. Mehmet Baray
Director of Institute of Engineering and Sciences

ABSTRACT

LOSSLESS COMPRESSION OF MEDICAL IMAGES

Behiç Fırat Kılıç

M.S. in Electrical and Electronics Engineering
Supervisor: Assoc. Prof. Dr. Ahmet Enis Çetin
July 1991

The digital imaging techniques are used more and more in various diagnostic modalities, including computed tomography (CT), Digital Subtraction Angiography (DSA), etc. As a result, a huge amount of digital images are being generated. Therefore, techniques to compress these images into a more compact form for storage and transmission become more important and anyone involved with the storage of medical images in a Picture Archiving and Communication System (PACS) should first apply compression (lossy or lossless depending on the characteristics of the digitized data) on them. There are image coding methods which reduces the storage size of an image by an amount of 1/10 without any visual degradation. But in some medical applications lossless coding of medical images is necessary.

Multiresolution coding techniques have been used for lossy image and speech coding, in this thesis we developed multiresolution techniques for the lossless image compression. We observed that the use of multiresolution techniques in lossless compression is as advantages as in lossy compression schemes.

ÖZET

MEDİKAL GÖRÜNTÜLERİN KAYIPSIZ OLARAK SAKLANMASI

Behiç Fırat Kılıç

Elektrik ve Elektronik Mühendisliği Bölümü Yüksek Lisans

Tez Yöneticisi: Doç Dr. Ahmet Enis Çetin

Temmuz 1991

Sayısal görüntüleme tekniklerinin tomografi, anjiografi vs. gibi alanlarda her geçen gün daha fazla kullanılmaya başlanmasıyla, çok fazla yer kaplayan medikal görüntülerin saklanması önemli bir sorun olmuştur. Bu görüntülerin daha yoğun bir şekilde saklanmasını sağlayan teknikler de önem kazanmıştır. Bazı geliştirilen *az kayıplı* metodlarla bu görüntülerin kapladığı hafıza 1/10 oranında azaltılabilir. Fakat, medikal alandaki bazı kullanımlardan dolayı *hiç kayıpsız* saklama önemli bir araştırma alanı olarak kalmıştır.

Bugüne kadar çok çözünürlü teknikler *kayıplı saklamalarda* kullanım alanı bulmuştur. Bu tezde ise çok çözünürlü teknikler kayıpsız görüntü saklamasında kullanıldı. Çok çözünürlü tekniklerin, en az kayıplı saklamada olduğu kadar kayıpsız saklama alanında da başarılı olabileceği yapılan deneylerle de gözlendi.

ACKNOWLEDGMENT

My first debt is to Assoc. Prof. Dr. A.Enis Çetin who provided perfect dosage of criticism and unfailing support throughout the project.

A special note of thanks is to my father and mother who are the real supporters of my studies in Bilkent University.

Finally I should like to thank to all my friends, primarily to Erkan Savran and Atilla Malaş, in the University who have one way or another participated, helped or supported me all through my MS work.

-

Contents

1	Introduction	1
1.1	Introduction	1
2	An Overview of Some Multirate Techniques	3
2.1	Introduction	3
2.1.1	Subband Decomposition	4
2.1.2	Pyramidal Decomposition	14
3	Review of Some Waveform Coding Methods	16
3.1	Introduction	16
3.1.1	Transform Domain Coding	17
3.1.2	The Huffman Coding	22
3.1.3	Run-length Coding	24
4	Lossless Multirate Image Coding	26
4.1	Introduction to Subband Coding	26
4.1.1	Coding with nonrectangular subband decomposition . . .	27
4.1.2	Coding with rectangular subband decomposition	30
4.1.3	Pyramidal Coding	31
4.2	Conclusion	34

List of Figures

2.1	$X(e^{jw})$ is the shifted and scaled sum of $X_a(j\Omega)$	4
2.2	Interpolation operation	5
2.3	Decimation operation	6
2.4	Subband decomposition by 2	7
2.5	Ideal subband decomposition structure	7
2.6	Cuix-cunx downsampling structure	10
2.7	Nonrectangular subband decomposition	11
2.8	Reconstruction stage for nonrectangular decomposition	12
2.9	Rectangular subband decomposition	13
2.10	Reconstruction stage for rectangular decomposition	14
2.11	The pyramidal coding stage	15
2.12	Reconstruction stage of the pyramidal coder	15
3.1	Reconstruction of the image from basis images	18
3.2	Zig-zag scan	21
3.3	Huffman codebook for the DCT algorithm	22
3.4	Optimum code assignment procedure	23
3.5	Modified Huffman codebook pair	24

4.1	Code book for low information sub-images	27
4.2	DCT coding structure	28
4.3	Pyramidal coding with lossy DCT	32
4.4	The sample image “Angio”	33
4.5	$1/4^{th}$ of the sample image “Chest” and “Lena”	34

List of Tables

4.1	DCT based results for nonrectangular decomposition	29
4.2	Difference based results for nonrectangular decomposition	30
4.3	DCT based results for rectangular decomposition	31
4.4	Difference based results for rectangular decomposition	31
4.5	Pyramidal compression results	33

-

Chapter 1

Introduction

1.1 Introduction

The digital imaging techniques such as Computed Tomography (CT), Digital Subtraction Angiography (DSA), ultrasound, nuclear medicine and computed radiography are used increasingly in various diagnostic modalities. Therefore, a huge amount of digital images are being generated, for which compression into a more compact form for storage and transmission become more important.

Medical images are usually digitized in large rasters. For instance, a chest X-Ray is scanned in a raster of size 2048 lines and 2048 columns (or 1024×1024 in some cases) with pixel depth of 12 bits and digital subtraction angiography (DSA) images are digital images of size 512×512 (in most machines) with pixel depth ranging from 8 bits to 12 bits. These figures stress the fact that, any type of image consumes a huge amount of storage media. Furthermore, if we consider storing huge amount of medical images in a hospital Picture Archiving and Communication System (PACS) the importance of the data compression becomes even clearer. Hence, anyone involved with the storage of medical images in an archive or the transmission via a telecommunication system should first apply compression (lossy or lossless depending on the characteristics of the digitized data) on them. We believe that the image compression unit must be an integral part of any PACS [2].

Image compression methods can be classified into two main categories: *Lossy (irreversible)* and *lossless (error-free)* coding methods. Although the techniques developed in this thesis can be applied to all kinds of images, we will be mostly concerned with error-free (lossless) coding of medical images due to legal reasons commonly used; lossy image coding methods can not be used in compressing medical images.

Many irreversible image coding techniques have been developed during the past two decades for applications ranging from tele-conferencing to the transmission of digital TV images [7],[8]. Recently, multiresolution image coding techniques become popular [4]-[7]. In gray-scale High Definition Television (HDTV) images a compression ratio of three can be achieved without introducing any visual degradation using multiresolution image coding (in fact some HDTV standard proposals are based on multiresolution image coding) and for other images compression ratios of as high as ten can be achieved by using multiresolution coding methods. However it should be noted that, visually lossless is a term to describe a lossy coding technique in which the reconstructed image is visually indistinguishable. As the name irreversible implies some numerical loss on the image pixel values are introduced in the irreversible coding. In some medical applications any loss however harmless it may seem may not be tolerable. Therefore, in these cases medical images should be compressed by numerically lossless image coding techniques. In this thesis we developed multiresolution lossless image compression methods for medical images.

Most of the lossless techniques are based on a lossy technique plus an error correction unit. So, we will first examine lossy techniques in order to extend the study to lossless cases, as well.

We also review multirate signal processing, transform domain coding, Huffman coding, run-length coding etc. in Chapters 2 and 3. These concepts are extensively used in our coding schemes. In Chapter 4 we present the newly developed multiresolution medical image coding methods and some simulation examples.

Chapter 2

An Overview of Some Multirate Techniques

2.1 Introduction

The key idea of multirate signal processing (MSP) based coding methods is to represent the signal in lower resolutions, and to encode each of these components separately. By dividing the signal into sub-signals we wish to remove redundancies and obtain a number of uncorrelated sub-signals.

These uncorrelated sub-signals can be coded more efficiently than the correlated ones [14], [15]. Indeed, optimum coding can be achieved if the signal can be decomposed into perfectly uncorrelated sub-signals [16].

In our case, the signals will be two dimensional and will be referred as images. The decomposition of images to several sub-images in lower resolutions has the advantage that, the number of bits used to encode a sub-image which corresponds to a frequency component of the original image can be variable. So, the encoding accuracy is always placed wherever it is needed in the frequency domain. In fact, bands with little or no energy may not be encoded at all in some lossy coding techniques.

The rest of this section describes two multirate signal processing techniques which are *subband* and *pyramidal* decomposition methods.

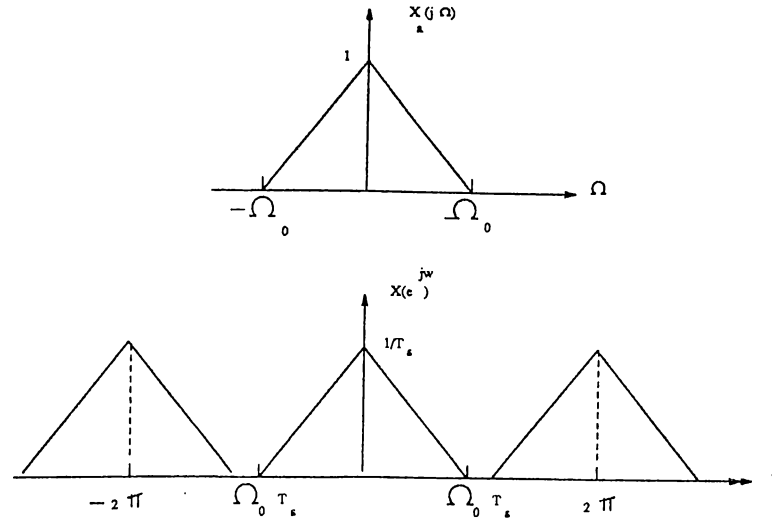


Figure 2.1: $X(e^{jw})$ is the shifted and scaled sum of $X_a(j\Omega)$

2.1.1 Subband Decomposition

Let us assume that we have a bandlimited analogue signal $x_a(t)$ with the continuous time Fourier transform (CTFT) $X_a(j\Omega)$.

$$x_a(t) \xleftrightarrow{CTFT} X_a(j\Omega) \quad (2.1)$$

It is well known from the *Nyquist theorem* that the discrete time Fourier transform (DTFT) of the signal $x_a(nT_s)$ is given as follows¹:

$$x_a(nT_s) \xleftrightarrow{DTFT} X(e^{jw}) = \sum_{k=-\infty}^{\infty} X_a(j\frac{w + 2\pi k}{T_s}) \frac{1}{T_s} \quad (2.2)$$

In Fig.2.1 we plot $X(e^{jw})$ and $X_a(j\Omega)$ where DTFT $X(e^{jw})$ of the analog signal $x_a(t)$ sampled at nT_s intervals is the shifted and scaled sum of it's CTFT $X_a(j\Omega)$.

Note that $\Omega = w/T_s$ and $T_s = 1/f_s$ in Fig.2.1. Aliasing occurs if sampling rate is lower than the Nyquist rate.

Having mentioned the frequency domain behavior of a bandlimited signal under sampling we can introduce the concept of *digital decimation* and *digital interpolation*.

The digital interpolation by a factor of **L** is the operation of padding **L-1** zeros between the consecutive samples of the time domain signal $x(n)$ and then

¹where $x_a(nT_s)$ is the samples of $x_a(t)$ taken at nT_s time intervals and n is an integer

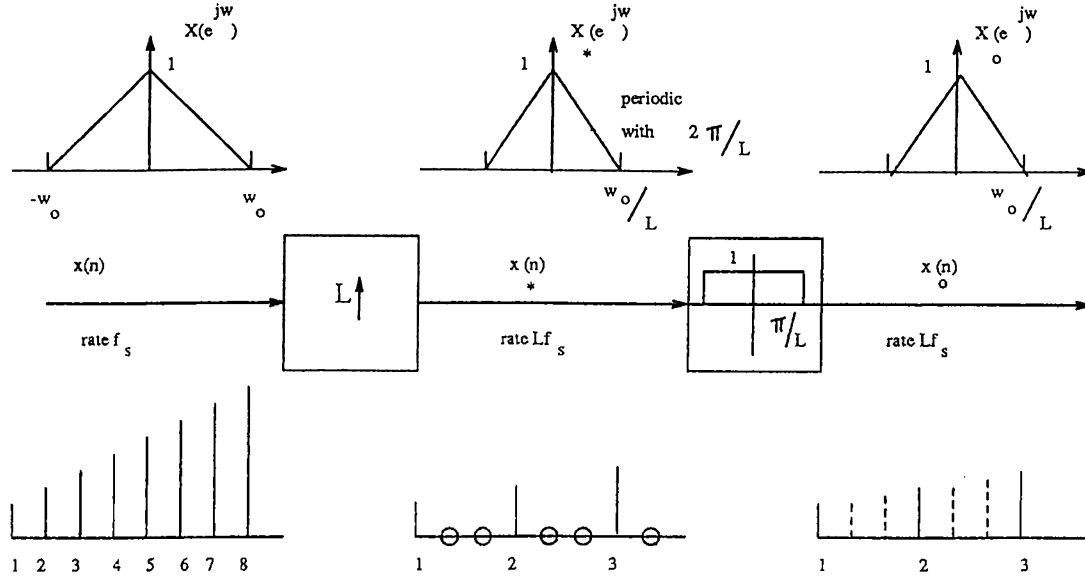


Figure 2.2: Interpolation operation

filtering it with a low-pass filter cut-off at π/L . Padding $L-1$ zeros increases the rate by a factor of L and represented by the symbol $L \uparrow$ to indicate an increase in rate by a factor of L^2 . This increase in the rate causes a proportional redundancy in the frequency content of the signal. The original signal can exactly be extracted from its' aliases by low-pass filtering with cut off π/L . The described operation of interpolation (both in time and frequency domain) can be summarized by the Fig.2.2. Note that in Fig.2.2 the zero padded signal $X_*(e^{jw})$ can be represented in terms of the input signal $X(e^{jw})$ as follows³:

$$X_*(e^{jw}) = X(e^{jLw}) \quad (2.3)$$

The digital decimation is a complementary operation of the digital interpolation. That is, if certain constraints are satisfied a decimated signal can exactly be reconstructed by interpolation. In decimation, rate of the signal is reduced by a factor of M and represented by $M \downarrow$. We filter the signal by a low-pass filter at cut-off π/M and drop $M-1$ samples out of every M samples⁴. Prefiltering by a low-pass filter (sometimes called the *decimation* filter) with cut off at π/M avoids the possibility of aliasing which may occur after decimation. Unlike to interpolation, decimation operation increases the bandwidth of the signal and may cause aliasing. The decimation operation (both in time and frequency domain behavior) can be seen in Fig.2.3.

²This is known as "upsampling"

³It is assumed that $w_0 < \pi$ in Fig.2.2

⁴This is known as "downsampling"

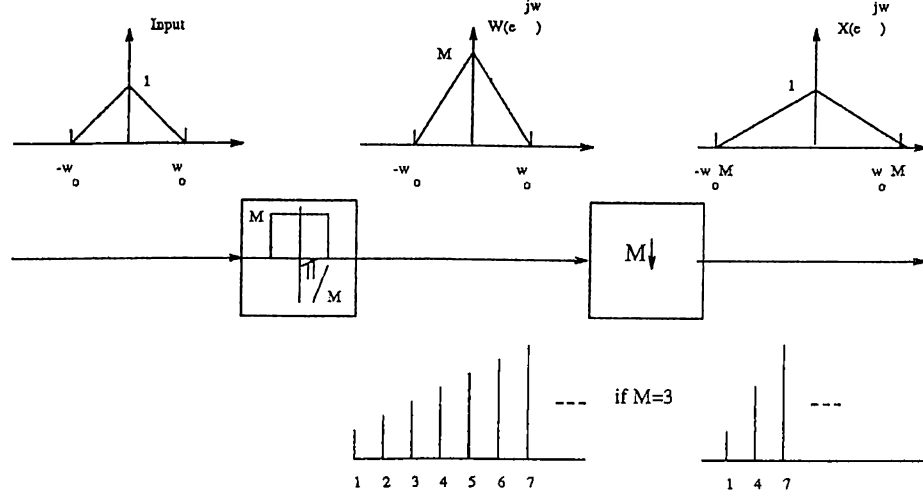


Figure 2.3: Decimation operation

Also note that

$$X(e^{jw}) = \frac{1}{M} \sum_{k=0}^{M-1} W(e^{j\frac{w-2\pi k}{M}}) \quad (2.4)$$

In *subband decomposition* the input signal is decomposed to sub-signals in lower resolutions according to their frequency contents. A typical subband decomposition structure is shown in Fig.2.4.

This structure decomposes the input signal $x(n)$ into two sub-signals x_l and x_u . After decimation by 2 (i.e., $2 \downarrow$), these two sub-signals are fed into the encoder and then to the channel. The receiver decodes the received signal and interpolates it. In this way, the input signal $x(n)$ can be reconstructed at the receiver.

Now let us derive the necessary conditions on the filters (in Fig.2.4) for exact reconstruction of the signal at the receiver part. For this purpose we will exclude the encoder, decoder and the channel. The structure will be as in Fig.2.5.

The formulation of the subband structure (in Fig.2.5) begins as follows:

$$X_l(e^{jw}) = H_l(e^{jw})X(e^{jw}) \quad (2.5)$$

$$X_u(e^{jw}) = H_u(e^{jw})X(e^{jw}) \quad (2.6)$$

from Eq.2.4

$$Y_l(e^{jw}) = \frac{1}{2} [X_l(e^{j\frac{w}{2}}) + X_l(e^{j\frac{w+2\pi}{2}})] \quad (2.7)$$

$$Y_u(e^{jw}) = \frac{1}{2} [X_u(e^{j\frac{w}{2}}) + X_u(e^{j\frac{w+2\pi}{2}})] \quad (2.8)$$

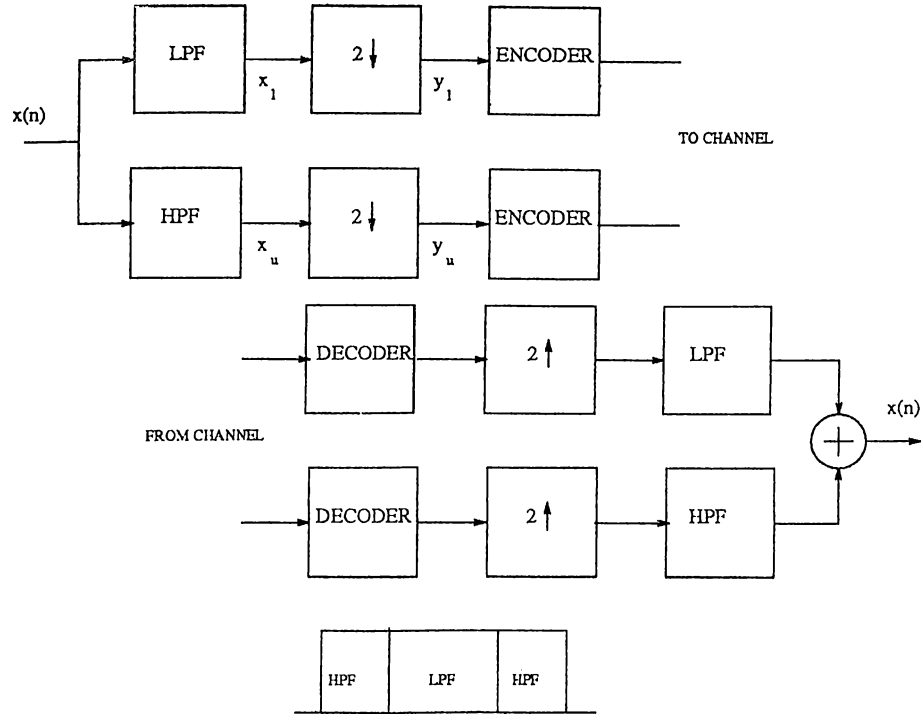


Figure 2.4: Subband decomposition by 2

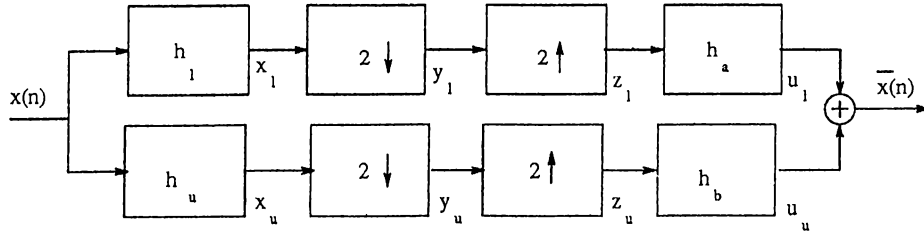


Figure 2.5: Ideal subband decomposition structure

and from Eq.2.3

$$Z_l(e^{jw}) = Y_l(e^{j2w}) = \frac{1}{2}[X_l(e^{jw}) + X_l(e^{j(w+\pi)})] \quad (2.9)$$

$$Z_u(e^{jw}) = Y_u(e^{j2w}) = \frac{1}{2}[X_u(e^{jw}) + X_u(e^{j(w+\pi)})] \quad (2.10)$$

The output is

$$\hat{X}(e^{jw}) = U_l(e^{jw}) + U_u(e^{jw}) \quad (2.11)$$

where

$$U_l(e^{jw}) = H_a(e^{jw})Z_l(e^{jw}) \quad (2.12)$$

$$= H_a(e^{jw})\frac{1}{2}[X_l(e^{jw}) + X_l(e^{j(w+\pi)})] \quad (2.13)$$

$$U_u(e^{jw}) = H_b(e^{jw})Z_u(e^{jw}) \quad (2.14)$$

$$= H_b(e^{jw})\frac{1}{2}[X_u(e^{jw}) + X_u(e^{j(w+\pi)})] \quad (2.15)$$

inserting $X_l(e^{jw})$ and $X_u(e^{jw})$ values (Equations 2.5,2.6) we get

$$U_l(e^{jw}) = \frac{H_a(e^{jw})}{2} [H_l(e^{jw})X(e^{jw}) + H_l(e^{j(w+\pi)})X(e^{j(w+\pi)})] \quad (2.16)$$

$$U_u(e^{jw}) = \frac{H_b(e^{jw})}{2} [H_u(e^{jw})X(e^{jw}) + H_u(e^{j(w+\pi)})X(e^{j(w+\pi)})] \quad (2.17)$$

so the output is

$$\begin{aligned} \tilde{X}(e^{jw}) &= \frac{X(e^{jw})}{2} \overbrace{[H_a(e^{jw})H_l(e^{jw}) + H_b(e^{jw})H_u(e^{jw})]}^{\Delta_1(e^{jw})} \\ &\quad + \frac{X(e^{j(w+\pi)})}{2} \overbrace{[H_a(e^{jw})H_l(e^{j(w+\pi)}) + H_b(e^{jw})H_u(e^{j(w+\pi)})]}^{\Delta_2(e^{jw})} \quad (2.18) \\ \tilde{X}(e^{jw}) &= \frac{X(e^{jw})}{2} \Delta_1(e^{jw}) + \frac{X(e^{j(w+\pi)})}{2} \Delta_2(e^{jw}) \quad (2.19) \end{aligned}$$

By examining Eq.2.19 we observe that, the output is composed of two terms where $X(e^{j(w+\pi)})$ is an aliasing term which should be canceled for exact reconstruction of $X(e^{jw})$ at the output.

In order to reconstruct the input signal at the output of the subband decomposition structure we need to have:

- 1) $\Delta_1(e^{jw}) = 2e^{jk_o w}$ where $e^{jk_o w}$ only introduces a time delay of k_o units; and
- 2) $\Delta_2(e^{jw}) = 0$ to cancel the aliasing term.

There are many filter design algorithms [3]-[5],[10],[11] to satisfy these conditions and hence, reproduce exact replica of $x(n)$ at the receiver.

One such design with FIR filters is obtained as follows:

- Choose $h_l(n)$ of order N (N is even) linear phase FIR low pass filter⁵.
- Choose $h_u(n) = (-1)^n h_l(n)$.⁶

⁵Note $h_l(n) = h_l(N - n - 1) \Rightarrow H_l(e^{jw}) = |H_l(e^{jw})|^2 e^{-jw(\frac{N-1}{2})}$

⁶Note that if $h_l(n)$ is a low pass filter then $h_u(n)$ is a high pass filter

It can be easily shown that $H_u(e^{jw}) = H_l(e^{j(w+\pi)})$ with the above choice of the filters.

- Finally choose

$$H_a(e^{jw}) = 2H_l(e^{jw}) \quad \text{and} \quad (2.20)$$

$$H_b(e^{jw}) = -2H_u(e^{jw}) \quad (2.21)$$

which implies that

$$\Delta_2(e^{jw}) = H_l(e^{j(w+\pi)})H_a(e^{jw}) + H_u(e^{j(w+\pi)})H_b(e^{jw}) \quad (2.22)$$

$$= H_l(e^{j(w+\pi)})2H_l(e^{jw}) + H_u(e^{j(w+\pi)})(-2)H_u(e^{jw}) \quad (2.23)$$

$$= H_l(e^{j(w+\pi)})2H_l(e^{jw}) + H_l(e^{jw})(-2)H_l(e^{j(w+\pi)}) \quad (2.24)$$

$$= 0$$

and

$$\Delta_1(e^{jw}) = H_l(e^{jw})H_a(e^{jw}) + H_b(e^{jw})H_u(e^{jw}) \quad (2.25)$$

$$= H_l(e^{jw})2H_l(e^{jw}) - 2H_u(e^{jw})H_u(e^{jw}) \quad (2.26)$$

$$= 2[H_l(e^{jw})^2 - H_u(e^{jw})^2] \quad (2.27)$$

note that

$$H_l(e^{jw})^2 = |H_l(e^{jw})|^2 e^{-j(N-1)w} \quad (2.28)$$

$$\begin{aligned} H_u(e^{jw})^2 &= |H_u(e^{jw})|^2 e^{-j(N-1)w} \\ &= |H_l(e^{-j(w+\pi)})|^2 e^{-j(N-1)(w+\pi)} \\ &= -|H_l(e^{j(w+\pi)})|^2 e^{-jw(N-1)} \end{aligned} \quad (2.29)$$

$$\implies \Delta_1(e^{jw}) = 2[|H_l(e^{jw})|^2 + |H_l(e^{j(w+\pi)})|^2]e^{-jw(N-1)} \quad (2.30)$$

so if

$$|H_l(e^{jw})|^2 + |H_l(e^{j(w+\pi)})|^2 = 1 \quad (2.31)$$

then we satisfy the exact reconstruction condition.

One dimensional (1-D) subband decomposition structures can be generalized to two dimensions (2-D) in many ways [7]. We now consider two different subband decomposition methods to decompose the image into sub-images at different resolutions. In the first method, we use diamond shaped exact reconstruction filter banks [4],[11] and in the second one we use rectangular exact reconstruction filter banks [5],[9],[10].

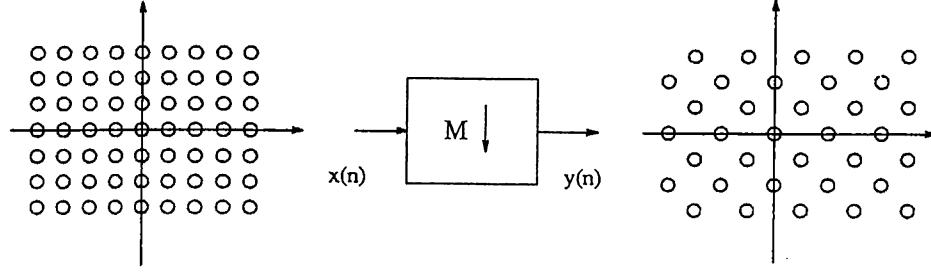


Figure 2.6: Quin-cunx downsampling structure

Nonrectangular Subband Decomposition

The nonrectangular decomposition is a subband decomposition method where nonrectangular filters are used for separation of the input image into sub-images (see Fig.2.7). The simplest way is the separable 2-D filters where the filters are constructed from 1-D filters such that;

$$H_{ij}(w_1, w_2) = H_i(w_1)H_j(w_2) \quad (2.32)$$

But other non-separable 2-D structures can also be used. Apart from filters the decimation and interpolation operations can also be generalized to the 2-D case in the same manner. This time decimation and interpolation factors $\mathbf{M}$ and $\mathbf{L}$, respectively turn into matrices. The sampling rate is either decreased or increased by a factor of $|\det(\mathbf{M})|$ or $|\det(\mathbf{L})|$, respectively. A simple decimation operation by a factor of $\mathbf{M}$ in 2-D space can be seen in Fig.2.6 where

$$\mathbf{M} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.33)$$

In Fig.2.6 $y(\underline{n}) = x(\underline{M} \underline{n})$. This decimation scheme is also known as the “quin-cunx” downsampling where only half of the samples are retained and the rest is dropped [17].

Our nonrectangular subband decomposition scheme consists of two stages. In the first stage the medical image is filtered by two filters. One of these filters is a lowpass filter with a diamond shaped⁷ passband. The other filter is a high pass filter with a diamond shaped stopband. The outputs of these filters are downsampled by a factor of 2 on a line quin-cunx grid. In this way two sub-images are obtained. The sub-image coming from the lowpass filter is decomposed and downsampled once again by the same filterbank. In effect

⁷Among many other possible nonrectangular filter structures “diamond shape” is a specific filter (see Fig.2.7) which we used in our simulations.

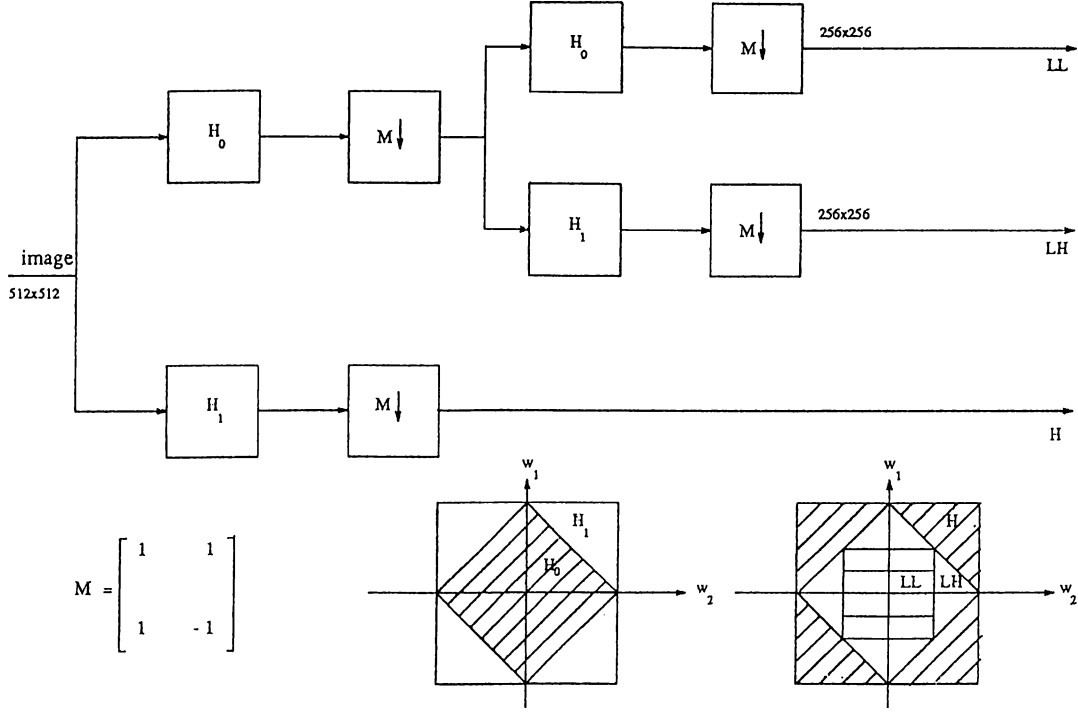


Figure 2.7: Nonrectangular subband decomposition

the procedure splits the image information into three regions in the frequency domain, ideally as shown in Fig.2.7. In this way a low resolution sub-image LL and two difference sub-images (LH and H) are obtained from the original image.

In Chapter 3 we make use of the special characteristics of these sub-images in our coding algorithms to lower bitrate or equivalently to improve data compression.

In order to perform the multiresolution subband decomposition the following IIR filters described in [4] are used:

$$H_i(\omega_1, \omega_2) = \frac{1}{2} + 2(-1)^i e^{j\omega_1} T(\omega_1 + \omega_2) T(\omega_1 - \omega_2), \quad i = 0, 1 \quad (2.34)$$

where

$$T(\omega) = \frac{1}{2} \cdot \frac{1/3 e^{j\omega} + 1}{1/3 + e^{j\omega}} \quad (2.35)$$

is an all-pass section. The filter with subscript $i = 0$ ($i = 1$) is the lowpass (highpass) filter. In order to reconstruct the coded image from the sub-images the synthesis filter bank of [4]

$$Y_i(w_1, w_2) = 2 H_i(-w_1, -w_2), \quad i = 0, 1 \quad (2.36)$$

which provides exact reconstruction is used. The reconstruction structure can be seen in Fig.2.8.

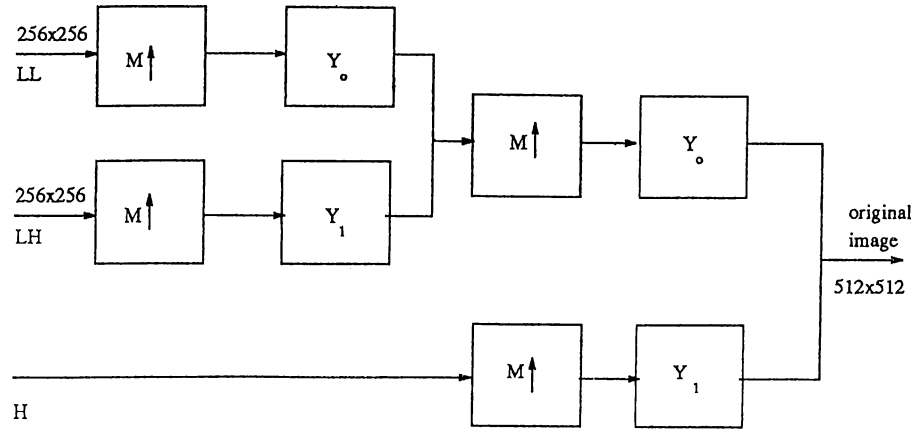


Figure 2.8: Reconstruction stage for nonrectangular decomposition

During this process no loss is introduced by the chosen filter bank of [4]. Hence, by the proper choice of filter bank the frequency content of the image can be split into lower resolutions and then can be reconstructed again without any distortion. This method is called as the nonrectangular subband decomposition.

Rectangular Subband Decomposition

The second subband decomposition method which we consider is the well known rectangular decomposition method [3],[6]. It is another subband decomposition method in which exact reconstruction filters of rectangular shape (see Fig.2.9) are used.

A splitting procedure similar to the *nonrectangular* case is applied to the image. But this time, the filters in the structure are rectangular and instead of applying a tree like structure for decomposition we use the structure shown in Fig.2.9. The image is split into 4 sub-images instead of three as in the case of nonrectangular subband decomposition.

In our work, we use the separable IIR⁸ filter bank of [5] which introduce no loss to the image. The filters used are

$$H_{ij}(\omega_1, \omega_2) = H_i(\omega_1)H_j(\omega_2), \quad i, j = 0, 1 \quad (2.37)$$

where

$$H_i(\bar{\omega}) = \frac{1}{2}[1 + (-1)^i e^{j\omega} T(2\omega)] \quad (2.38)$$

⁸Infinite impulse response

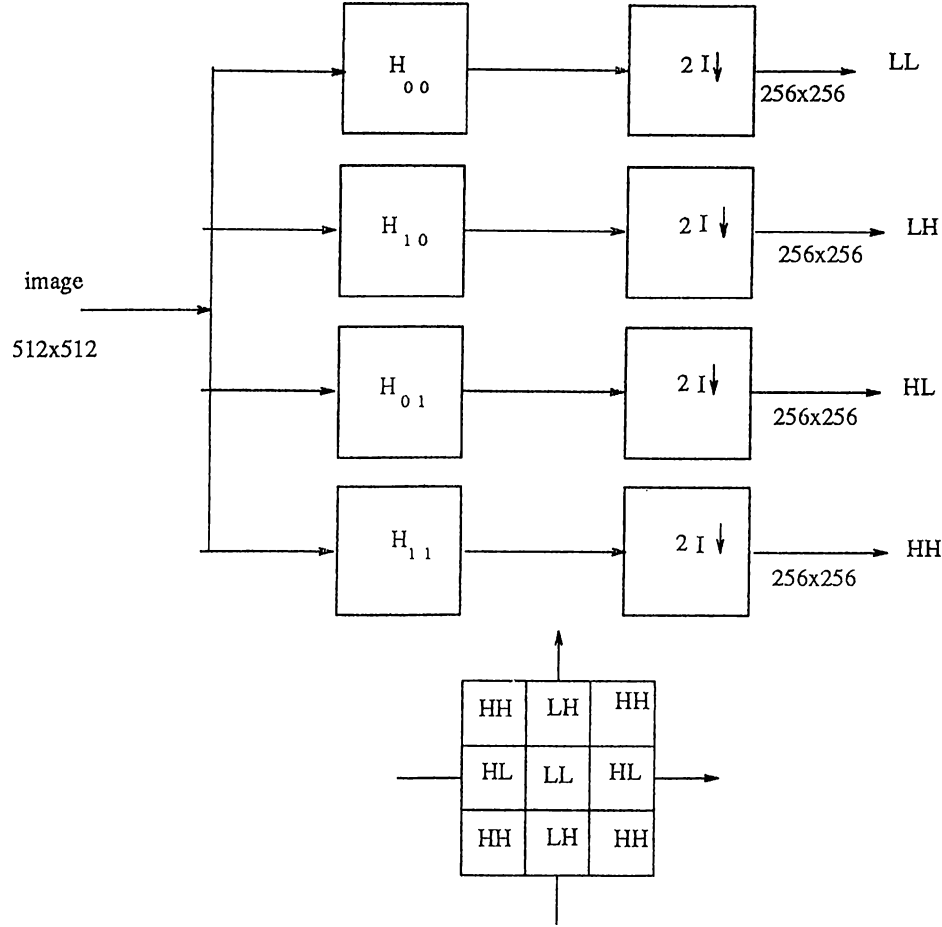


Figure 2.9: Rectangular subband decomposition

and

$$T(2\omega) = \frac{1/3e^{j2\omega} + 1}{1/3 + e^{j2\omega}} \quad (2.39)$$

The low resolution sub-image (LL in Fig.2.9) carries the highest information among other sub-images LH, HL, HH. Later on, this property will be exploited to improve the compression ratios of the images.

Reconstruction of the original image from the sub-images is carried out by the synthesis filter bank of [5]

$$Y_i(\omega) = \frac{1}{2}[1 + (-1)^i T(-2\omega)], \quad i = 0, 1 \quad (2.40)$$

with the reconstruction structure of Fig.2.10.

These decompositions of the image into sub-images (either rectangular or nonrectangular) enable us to make use of particular code-books to encode particular frequency ranges. As it will be discussed in detail later on, this improves data compression.

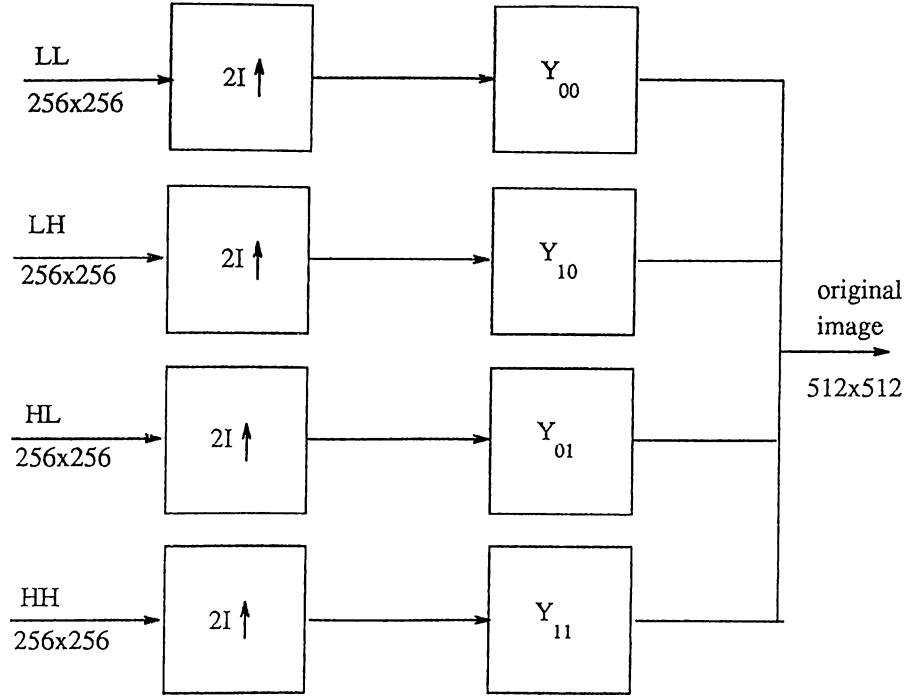


Figure 2.10: Reconstruction stage for rectangular decomposition

2.1.2 Pyramidal Decomposition

Pyramidal decomposition is another multirate signal decomposition scheme. A typical pyramidal coding structure is shown in Fig.2.11.

The image $x(\mathbf{n})$ (where $x(\mathbf{n}) = x(n_1, n_2)$) is filtered and down sampled, on a rectangular grid, to $x_1(\mathbf{n})$ where the number of pixels are reduced by a factor of 4 during the process⁹. The image $x_1(\mathbf{n})$ corresponds to the low frequency range hence, carries more information. After it is being coded this quantized and encoded sub-image $x_b(\mathbf{n})$ is then upsampled and interpolated to form $x_2(\mathbf{n})$ (which is of the original image size). Then the error image $\tilde{x}(\mathbf{n})$, is obtained by subtracting the reconstructed image $x_2(\mathbf{n})$ from the original image $x(\mathbf{n})$, i.e. ,

$$\tilde{x}(\mathbf{n}) = x(\mathbf{n}) - x_2(\mathbf{n}) \quad (2.41)$$

This completes one full cycle through the pyramidal structure. The error image $\tilde{x}(\mathbf{n})$, is either entered to a similar structure once more or transmitted to the receiver after being quantized and coded.

Pyramidal decomposition structure produces samples (5/4 times) more than the input image samples whereas in subband decomposition structures

⁹H is a separable 2-D Lagrange filter of length 7 and $2\mathbf{I} \downarrow$ is a rectangular downsampling matrix $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

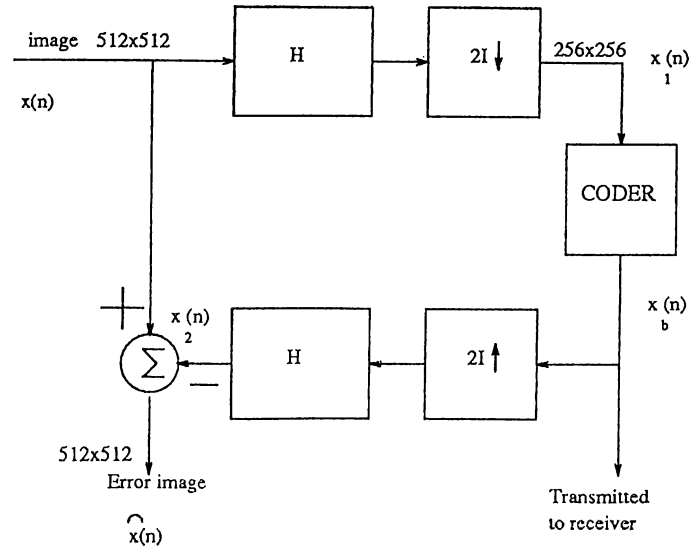


Figure 2.11: The pyramidal coding stage

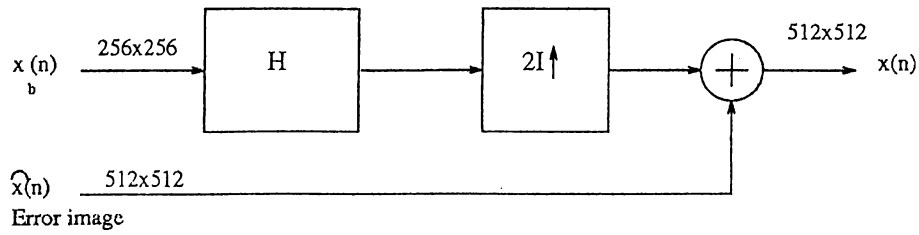


Figure 2.12: Reconstruction stage of the pyramidal coder

the number of output samples are equal to the input samples. The pyramidal structure is suitable for compression purposes as the error sequence $\tilde{x}(n)$ which is of the original image size carries very low information.

The receiver part of the pyramidal coder is shown in Fig.2.12. The reconstruction is error free without considering the transmission errors that may occur in the channel.

Chapter 3

Review of Some Waveform Coding Methods

3.1 Introduction

The main purpose of any image coding method is to reduce the redundancies in the image as much as possible so that one either communicates in lower bit rates or stores the image in less memory. We try to assign symbols to pixels in an image, in such a way that we achieve this goal.

Image coding methods can be classified in two main categories: *irreversible (lossy) coding* and *lossless coding*. Irreversible coding methods add distortion to the image during the compression of the data which can not be exactly recovered afterwards. That is, some amount of information is lost forever. The amount of information loss during the process also determines the amount of compression. The more information (visual quality) one loses the more compression one gets. Depending on the technique used, the relation between these two criteria differ. That is, for one method one loses too much from visual quality to reach a certain compression ratio where as for another one does not lose that much from visual quality (information) to reach the same compression ratio. The aim is to achieve visually lossless images with minimum loss of information.

In lossy image compression visual quality is a vital criteria. Hence there has to be an objective measure to decide on the distortion introduced to the image. Although there is not yet such a mathematical expression which is in full correlation with the human visual system, there are still expressions that somehow give a measure of the quality of the restored image. This way, we use both subjective measures as well as the objective ones to indicate the visual quality of the image.

Some very common objective distortion measures are

$$M.S.E. = \frac{\sum [x(n) - y(n)]^2}{N^2} \quad (3.1)$$

$$MSE_{\sqrt{}} = \sqrt{\frac{\sum [x(n) - y(n)]^2}{N^2}} \quad (3.2)$$

$$MSE_{db} = -20 \log_{10} \left(\frac{\sqrt{\frac{\sum [x(n) - y(n)]^2}{N^2}}}{x(n)_{max}} \right) \quad (3.3)$$

where $x(n)$ and $y(n)$ are input and output images to the image compression scheme, respectively and N is the dimension of the images. The summations are over all image and $x(n)_{max}$ is the maximum gray-level value of the input image.

In lossless coding, however, there is no loss at all. That is all the gray-level values of the image are restored back exactly. The problem in lossless coding is to propose such a method that a considerable compression can be achieved while no distortion is introduced at all.

In the next sections we introduce some coding strategies which we extensively use in our image compression schemes.

3.1.1 Transform Domain Coding

The term *image transform* usually refers to a class of unitary matrices for representing images [7]. Just as a 1-D signal can be represented by an orthogonal series of *basis functions*, an image can also be represented in terms of a discrete set of basis arrays called *basis images*. These basis images can be generated by unitary matrices.

A unitary transform is represented by a matrix, A . The transform $\underline{v}$ of a 1-D sequence $\{ u(n), 0 \leq n \leq N-1 \}$ is given by

$$v = Au \implies v(k) = \sum_{n=0}^{N-1} a(k, n)u(n), \quad 0 \leq k \leq N-1 \quad (3.4)$$

where $A^{-1} = A^{*T}$ (as in all unitary transforms) then

$$u = A^{*T}v \implies u(n) = \sum_{k=0}^{N-1} v(k)a^{*}(k, n), \quad 0 \leq n \leq N-1 \quad (3.5)$$

The columns of A^{*T} are called the basis vectors of A . Depending on the transformation used these basis vectors differ.

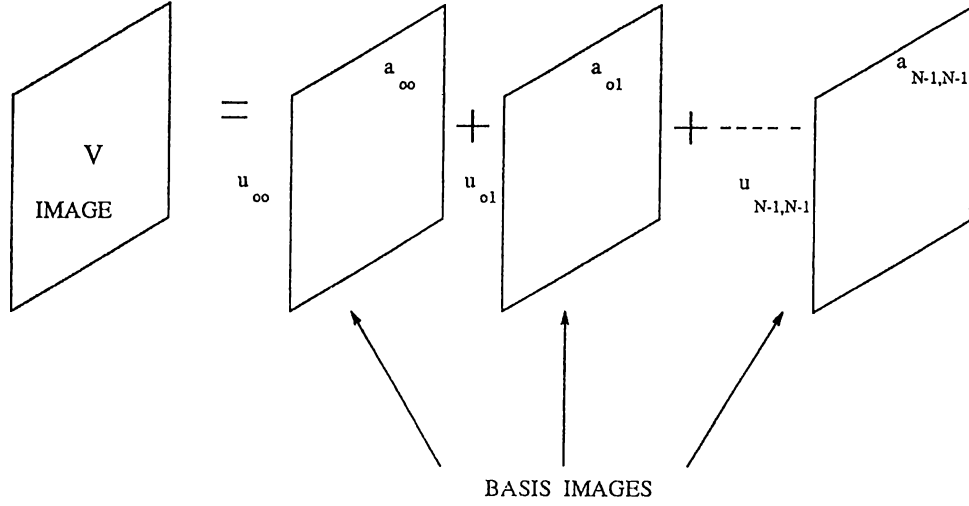


Figure 3.1: Reconstruction of the image from basis images

In 2-D the situation is quite similar and an image of size $N \times N$ can be represented by the transformation of the form

$$v(k, l) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u(m, n) a_{k,l}(m, n), \quad 0 \leq k, l \leq N-1 \quad (3.6)$$

and

$$u(m, n) = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} v(k, l) a_{k,l}^*(m, n), \quad 0 \leq m, n \leq N-1 \quad (3.7)$$

where $a_{k,l}(m, n)$ is a set of complete orthonormal discrete basis functions satisfying the properties

i) **Orthonormality** $\sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a_{k,l}(m, n) a_{k',l'}^*(m, n) = \delta(k - k', l - l')$

ii) **Completeness** $\sum_{k=0}^{N-1} \sum_{l=0}^{N-1} a_{k,l}(m, n) a_{k,l}^*(m', n') = \delta(m - m', n - n')$

Most of the unitary transforms used in image processing are separable that is.

$$a_{k,l}(m, n) = a_k(m) b_l(n) \triangleq a(k, m) b(l, n) \quad (3.8)$$

where $\{ a_k(m), k = 0, \dots, N-1 \}$ $\{ b_l(n), l = 0, \dots, N-1 \}$ are 1-D complete orthonormal sets of basis [7].

The 2-D transformations can be summarized as the representation of an image by the scaled sum of basis images. That is basis images form such a set that if they are scaled and summed properly they can construct any image. The situation can be seen in Fig.3.1. where u_{ij} 's are the scaling coefficients.

In transform coding, we only transmit the scaling coefficients and not the image. The receiver reconstructs the original image from the basis images by

only using the scaling coefficients. By storing (or transmitting) the relatively large scaling coefficients, in magnitude, of the basis images we can remove the redundancies in the original image. For instance, a very small coefficient in magnitude implies a very small information for that specific basis image. Hence, we usually drop or give less emphasis to such coefficients in coding. In this way, we improve our compression results without much distorting the visual quality.

A transformation should have the ability to represent an image by as less scaling coefficients as possible¹. In image coding *Discrete Cosine Transform* is the most popularly used transform which is described next.

Discrete Cosine Transform

The $N \times N$ discrete cosine transform $F(u, v)$ of an image $f(j, k)$ is defined as

$$F(u, v) = \frac{4C(u)C(v)}{N^2} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} f(j, k) \cdot \cos \left[\frac{(2j+1)u\pi}{2N} \right] \cdot \cos \left[\frac{(2k+1)v\pi}{2N} \right] \quad (3.9)$$

for $u, v = 0, 1, \dots, N-1$ where

$$C(w) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } w=0 \\ 1 & \text{for } w=1, 2, \dots, N-1 \end{cases} \quad (3.10)$$

The inverse transform is given by

$$f(j, k) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v) \cdot \cos \left[\frac{(2j+1)u\pi}{2N} \right] \cdot \cos \left[\frac{(2k+1)v\pi}{2N} \right] \quad (3.11)$$

for $j, k = 0, 1, \dots, N-1$.

DCT is a separable transform and it has very high energy compaction property for images with high correlations among neighboring pixels [7].

As it is well known Karhunen-Loeve transform (KLT) is the optimum transform in terms of energy compaction. However, KLT is dependent on the statistics of the input signal hence it has a signal dependent coefficient matrix and not very convenient for most practical image processing applications. DCT has a constant transform matrix (independent of the input signal statistics) so it is suboptimum however, much easier to implement. DCT is a very good

¹This is equivalent to having high energy compaction property

approximation to the KLT for the 1st order highly correlated Markov sequences. An $N \times N$ DCT matrix is very close to the KLT matrix of a first order stationary Markov sequence of length N whose covariance matrix is given by

$$R = \begin{bmatrix} 1 & \rho & \rho^2 & \dots & \rho^{N-1} \\ \rho & 1 & \rho & & \rho^{N-2} \\ \vdots & \vdots & & & \vdots \\ \rho^{N-1} & \rho^{N-2} & & \rho & 1 \end{bmatrix} \quad (3.12)$$

when the correlation parameter ρ is close to unity [7]. Hence, it is widely accepted and even used in some image transmission standardization proposals [18]. Fast DCT algorithms can also be implemented by using FFT algorithms. DCT of a vector of N elements can be calculated in $O(N \log_2 N)$ operations via an N -point FFT [19] which we used this property in our DCT implementations.

The adaptive DCT coder that we implemented for image compression purposes performs transform domain coding. The algorithm is a modified version of the study by [1] and by itself a lossy coding scheme.

We first divide the image in small sub-blocks of typical size 8×8 pixels then take the DCT of each sub-block and put a threshold (T) on the transform domain coefficients. This threshold can be adjusted to eliminate small amplitude coefficients which do not carry much information. The coefficients with amplitudes less than the predefined threshold (T) are made zero, where the others are scaled by the threshold amount T . In this way the number of zero amplitude coefficients are increased and longer run-lengths are obtained². While doing so we do not touch the luminance value of the sub-block which is the (0,0)'th coefficient of the DCT transformed sub-block because this value determines the intensity of the “blocking effect” which is easily observed visually (Mean values of the consecutive blocks change, if the luminance value is harshly quantized). Blocking effects occur due to the quantization of the transform domain coefficients and appears as small blocks of different intensities on the image. It destroys the visual quality and mean square error of the image. In order to avoid “blocking effect” we do not threshold the luminance value (this is the dominating factor in “blocking effect”) of the transformed sub-block and quantize it to full range which is 9 bits.

Rest of the transform domain coefficients are then quantized by the user defined quantization step (S). While doing so we quantize the small amplitude coefficients in better precision regardless of the user defined quantization step (S). This in return preserves the visual quality of the image.

²Run-length concept will be explained in Section 3.1.3

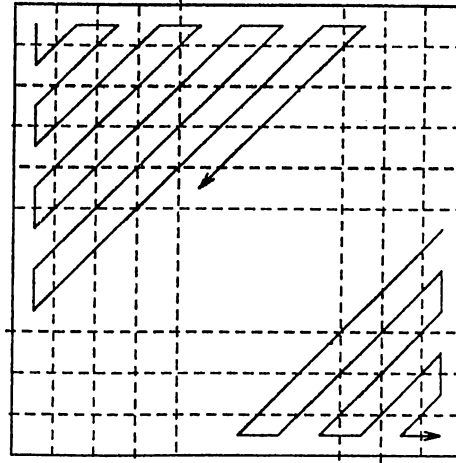


Figure 3.2: Zig-zag scan

A *modified Huffman* codebook and a *run-length* codebook (explained in Sections 3.1.2, 3.1.3) are then adaptively constructed according to the quantization step S .

The quantized block of coefficients are “zig-zag” scanned as seen in Fig.3.2. This zig-zag scanning structure optimizes the number of run-lengths of zeros and end-of-blocks. (During zig-zag scanning “zero-bridging³” is also performed). Because among many functions which model the probability density of the cosine transform coefficients $F(u, v)$, the Laplacian density has shown to provide the best fit [1]. This function can be written as

$$p(x; u, v) = \frac{1}{\sqrt{2}\sigma(u, v)} \exp \left[\frac{\sqrt{2}|x|}{\sigma(u, v)} \right] \quad (3.13)$$

where $\sigma(u, v)$ denotes standard deviation of a coefficient.

After run-lengths of zeros are determined and other coefficients are zig-zag scanned a predefined range is variable length coded, and the rest is fixed length coded (detailed explanation of the coding procedure is given in Section 3.1.3).

The performance of the coder is quite good in terms of mean square error and subjective evaluations. For instance, a compression ratio of 10 can be achieved without any visual degradation (subjective measure) for some images including the well known “Lena”. Typical codebook pairs used for coding transform domain coefficients and run-lengths are given in Fig.3.3.

³Zero-bridging is explained in the Section 3.1.3

AMPLITUDE	NUMBER OF CODE BITS	HUFFMAN CODES
-6	8	00000000
-5	8	01100000
-4	7	0000001
-3	5	01101
-2	4	0111
-1	1	1
1	3	001
2	5	00001
3	6	011001
4	7	0110001
5	8	00000001
6	8	01100001
EOB	4	0001
RL PREFIX	3	010
[-38,-38]	2+6	11+6 BITS
[255,-255]	3+9	101+9 BITS
RUN-LENGTH		
1	3	011
2	4	0101
3	4	0011
4	5	01000
5	5	10010
6	5	01001
7	5	10001
8	5	10011
9	5+9	00010+9 BITS

Figure 3.3: Huffman codebook for the DCT algorithm

3.1.2 The Huffman Coding

Let us assume that a discrete memoryless source U has a K letter alphabet $a_1, a_2, \dots, a_k$ with probabilities $P(a_1), P(a_2), \dots, P(a_k)$. Each letter can be represented by a codeword from a prescribed codebook⁴. Let the number of different symbols in the codebook be D (which is 2 in binary case; 0 and 1) and the number of symbols in the codeword corresponding to a_k be n_k (i.e., if $a_k=0111$ then $n_k=4$).

The average number of symbols per source letter $\bar{n}$ is defined as follows:

$$\bar{n} = \sum_{k=1}^K P(a_k) \cdot n_k \quad (3.14)$$

Any useful codebook must be uniquely decodable. We can intuitively argue that, if we can assign longer codewords to source letters with less probability

⁴Codebook is a list of codewords used in coding

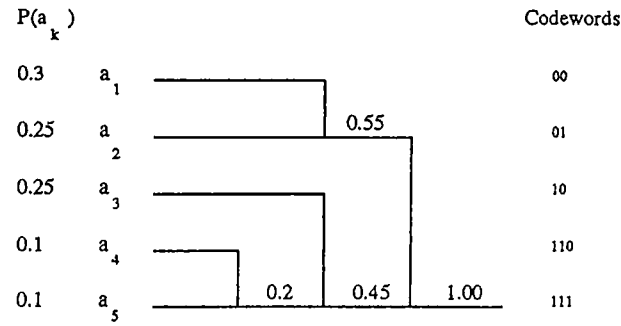


Figure 3.4: Optimum code assignment procedure

and the shorter codewords to source letters with high probability then we can represent our alphabet with less bits than the codebook which assigns fixed length codewords to each symbol. This is meaningful as most of the time the more probable source letters occur and since we assign shorter codewords to them, on the average we decrease the number of symbols transmitted.

In information theory, the well known concept of “source coding” says that [14]

$$\bar{n} \geq H(U) \quad (3.15)$$

at all times, where $H(U)$ is the entropy of the source U and given by

$$H(U) = \sum_{k=1}^K P(a_k) \log_2 \frac{1}{P(a_k)} \quad (3.16)$$

That is, the minimum possible symbol assignment is lower bounded by the entropy (H) of the source. By proper code assignment we try to achieve the entropy value of the source alphabet.

The optimum code assignment for source letters can be achieved by Huffman coding. In this assignment algorithm source letters are given source symbols beginning from the least probable two symbols, in a tree like structure. The coded two symbols and their probabilities are allied and a new source alphabet is formed. Then the same procedure, coding two least likely symbols, is applied recursively until we come up with one source letter. An example is given in Fig.3.4. A nice property of the Huffman coding is that, the output codewords are prefix conditioned (instantaneous). In other words, no codeword is a prefix of another codeword. Hence, decoding can be accomplished in real time without any storage requirement.

In our compression (coding) schemes we use codebooks designed by the Huffman coding algorithm. But sometimes the number symbols increase so

AMPLITUDE	NUMBER OF CODE BITS	HUFFMAN CODES
-6	8	0000000
-5	8	0110000
-4	7	0000001
-3	5	01101
-2	4	0111
-1	1	1
1	3	001
2	5	00001
3	6	011001
4	7	0110001
5	8	00000001
6	8	01100001
EOB	4	0001
RL PREFIX	3	010
[38, 38]	2 + 6	11 + 6 BITS
[255, 255]	3 + 9	101 + 9 BITS
RUN-LENGTH		
1	3	011
2	4	0101
3	4	0011
4	5	01000
5	5	10010
6	5	01001
7	5	10001
8	5	10011
9	5 + 9	00010 + 9 BITS

AMPLITUDE	NUMBER OF CODE BITS	HUFFMAN CODES
[16, 16]	1 + 5	1 + 5 BITS
[255, 255]	2 + 9	01 + 9 BITS
RL PREFIX	3	000
EOB	3	001
RUN-LENGTH		
1	2	00
2	2	10
3	4	1100
4	4	1101
5	5	11100
6	5	11101
7	5	11110
8	5	11111
9	2 + 9	01 + 9 BITS

Figure 3.5: Modified Huffman codebook pair

much that instead of paying the cost of a very long and complicated Huffman codebook, we pay the price of a little worse compression by making use of a *modified Huffman codebook*. This time, source letters within the predefined range (especially the high probable ones) are variable length coded and the remaining source letters are fixed length coded. This modification does not degrade the compression ratios (equivalently average bit per source letter) more than 5% in most cases [8]. Two such modified codebooks which we also used in our compression schemes is shown in Fig.3.5.

3.1.3 Run-length Coding

If a certain source symbol occurs consecutively in a sequence of symbols, then it is more efficient to encode this repeated symbol sequence in one codeword. For instance in a binary image sequence like facsimile, there is a great potential for intraframe redundancy reduction by means of a subsequent operation called *run-length coding*. In a text the binary symbol 0 (1) representing black (white) occurs with long runs so that consecutive runs of zeros or ones can be coded.

In gray-level images (2-D sequences) although there are more than 2 levels, in smooth regions we expect to have runs of pixels. Especially, in medical images there are frequent runs of constant gray-level values which can be encoded by run-length coding.

Run-length coding must be an essential part of any compression unit since it is very effective in reducing the primary redundancies (Other methods such as transform coding, subband coding is necessary to reduce secondary redundancies). A simple but popular gray-level lossless image coding is the *shift and subtract* scheme, in which we shift the image by one pixel to the right or left and subtract from itself to make use of repetitions in different gray-level values [20]. In this way, neighboring pixels with the same gray-level value subtract to zero and we obtain a new sequence with frequent runs of zeros⁵.

If “zero” valued samples are repeated consecutively then they can be encoded together by means of run-length coding. For this purpose we can reserve a special prefix called the *run-length prefix* and encode the number of repetitions of zero valued samples by either fixed length or variable length codes. The receiver recognizes run-length prefix and decodes the codeword attached to it as the number of repetitions of zeros.

Run-length coding is very convenient in eliminating the redundancies in smoothly varying images. Medical images are smoother than the typical images of natural scenes. Because of this higher compression ratios can be achieved in medical image coding than the coding of natural scenes.

We can further improve the results by the concept of *end-of-block (EOB)* and *zero-bridging (ZB)*.

EOB is a reserved codeword to indicate that runs of zeros continues till the end of encoded block. This saves lots of bits especially in DCT coding which will be explained in Chapter 4.

ZB can also be used if we perform lossy coding. The aim is to smooth out the isolated variations in the gray-level values and increase the number of run-lengths and EOB's. In zero-bridging, an isolated non-zero sample in between two zero valued samples is bridged over and converted to zero. In this way, we increase the number of run-lengths with the cost of losing some information which can not be recovered forever.

⁵Note that by subtracting the image from itself we increase the range of encoding from 8 bits to 9 bits

Chapter 4

Lossless Multirate Image Coding

4.1 Introduction to Subband Coding

In this chapter, we present our lossless multiresolution medical image coding methods. In Chapters 2 and 3 we summarized background material which is necessary for our lossless coding schemes.

In subband decomposition, either *rectangular* or *non-rectangular*, most of the information is localized in the low frequency range which is denoted by LL in Figs.2.7,2.9. This is due to the natural characteristics of the ordinary images. Especially, for medical images which have very little high frequency contents, the sub-images LH, HL, HH, H contain very little information.

LL sub-image is a lower resolution version of the original image. In typical images including medical images, this sub-image contains most of the information content of the original image. The rest of the sub-images carry detail information and by themselves do not even clearly represent the original image.

In order to obtain a good compression ratio, the low frequency portion sub-image (LL) of the original image should be encoded very efficiently. A good compression in this range effects the overall compression ratio significantly.

In many subband coding schemes, because of the importance of the LL sub-image, DCT based compression schemes are applied and usually no visual degradation is introduced to the LL sub-image. In detail images LH, HL, HH, H where there is less information than LL other techniques such as PCM with run-length coding which are computationally less costly, are used.

In the following Sections 4.1.1, 4.1.2, 4.1.3 the coding of three sample images

AMPLITUDE	NUMBER OF CODE BITS	HUFFMAN CODES
[1,4]	1+2	1+2 BITS
[0,128]	2+7	01+7 BITS
RL PREFIX	3	000
EOB	3	001
RUN-LENGTH		
1	2	00
2	2	10
3	4	1100
4	4	1101
5	5	11100
6	5	11101
7	5	11110
8	5	11111
9	2+9	01+9 BITS

Figure 4.1: Codebook for low information sub-images

with different methods is explained. The “Lena” image, although it is not a medical image, is used to compare our results with the ones reported in the survey of M. Rabbani and P.W. Melnychuk [8].

4.1.1 Coding with nonrectangular subband decomposition

In nonrectangular decomposition the image is split into 3 sub-images via diamond shaped filters (The decomposition procedure and the filters used are explained in detail in Section 2.1.1). These 3 images are referred as LL, LH and H (see Fig.2.7). The rest of the work is to apply appropriate coding to each of these sub-images.

The sub-images LH and H contain very low information and full of zero valued samples. So we can reach very high compression ratios (in the order of 10’s) by run-length coding of these sub-images. As usual the sub-images LH, H do not carry much information and dominated with zero samples. Hence, we can very efficiently compress them by run-length coding the zero samples and variable length coding the non-zero samples.

A specially designed run-length codebook suitable for these sub-images (LH and H) is used. The codebook which we used is shown in Fig.4.1. In order to reduce the complexity of the coder without drastically decreasing the compression ratio, we applied the modified Huffman code book design technique which was explained in Section 3.1.2.

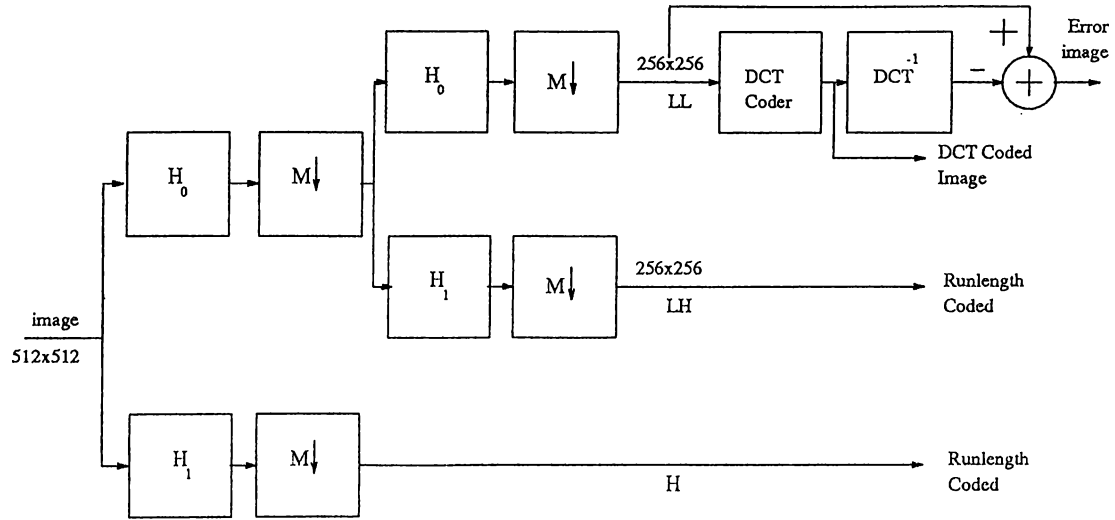


Figure 4.2: DCT coding structure

In our modified Huffman codebook we divided the source symbols into four main categories

- 1) The zero valued samples till the end of block which are encoded by EOB code;
- 2) Consecutive zero valued samples which are encoded by variable length codes after the run-length prefix (RL-pre);
- 3) Samples with low amplitudes which are encoded by fixed length codes after the low amplitude prefix; and
- 4) Samples with high amplitudes which are encoded by fixed length codes after the high amplitude prefix.

The assignment of these prefixes are made according to the frequency of occurrence of such samples. That is, if low amplitude samples are the most probable ones then we assign short prefix to this range and etc.

The sub-image LL is encoded by two different algorithms. These algorithms are “DCT based coding” and “difference coding”.

DCT based coding of LL sub-image: Commonly used DCT based coding schemes are lossy methods due to the quantization performed in the transform domain.

	HL	H	DCT	Error	R	H
Lena 512×512	2.82	3.01	5.28	1.49	2.13	2.82
Angio 512×512	3.61	3.97	4.67	1.87	2.61	3.46
Chest 1024×1024	6.11	4.53	17.56	1.71	3.21	5.04

Table 4.1: DCT based results for nonrectangular decomposition

By retaining the error image, the difference between the original and the DCT coded images, the coding scheme becomes lossless and in this way the information loss introduced by DCT compression is compensated. The overall bits used to encode LL is the sum of the bits used for the encoding of “error image” and the “DCT coding”. However, because of the high quality compression of the DCT the “error image” do not carry much information and dominated with consecutive zero samples so it is run-length coded as in the case of LH and H sub-images.

At this point, we should note that DCT compression is an adaptive scheme and can compress the same image to varying ratios by the change of user-dependent inputs such as the threshold (T) and the quantization step (S). This change also effects the “error image” statistics, as the information loss due to DCT compression depends on the constants T and S . Hence, an optimum point between DCT compression and “error image” compression can be found.

We performed many experiments with the structure of Fig.4.2 using the codebook of Fig.4.1 for sub-images LH, H and obtained results which are presented in Table 4.1.

In the table **R** value is the realized total compression ratio and the **H** is the theoretical entropy value¹. HL, H are realized compression ratios for subbands HL and H respectively. *DCT* and *Error* are the realized compression ratios for the DCT algorithm and error image respectively, for sub-image LL.

Difference coding of LL sub-image: This method is easier to implement than the DCT based method for coding the LL sub-image. This can be the shift and subtract method which was mentioned in Section 3.1.3. This time LL sub-image is shifted by one pixel to the right and subtracted from itself, and resulting difference image is variable length and run-length coded. The results of the shift and subtract operation is presented under LL column. In this way we achieve the results of Table 4.2.

¹One should note that the values in Tables 4.1-4.5 are the compression ratios not the average bits and the results of the DCT algorithm is obtained with codebook of Fig.3.3

	LL	HL	H	R	H
Lena 512×512	1.20	2.82	3.01	2.16	2.53
Angio 512×512	1.55	3.61	3.97	2.81	3.25
Chest 1024×1024	1.48	6.11	4.53	3.10	4.50

Table 4.2: Difference based results for nonrectangular decomposition

We obtained similar results for all the medical images we tried although there may be images where these two methods differ a lot from each other. The difference coding is faster and easier to implement than the DCT based coding besides there is no need to code the “error image” as the method is lossless from the start. However, DCT compression achieves very high compression ratios in the case of lossy coding (with out considering the “error image”), and in situations where some loss can be tolerable it is superior to any other method.

4.1.2 Coding with rectangular subband decomposition

The coding scheme with rectangular decomposition is very similar to the “coding with nonrectangular decomposition” which was explained in the previous section. This time decomposition of the image into lower resolutions is done by rectangular exact reconstruction filters described in Section 2.1.1 and the image is split into four sub-images which will be referred as LL, LH, HL and HH (see Fig.2.9).

As usual the sub-images LH, HL, HH do not carry much information and dominated with zero samples. Hence, we can very efficiently compress them by run-length coding the zero samples and variable length coding the non-zero samples using a similar codebook of the previous coding scheme².

The problem of coding the LL sub-image is handled as in the case of previous section. To code the LL sub-image we again use the same methods that we used in Section 4.1.1: *DCT based coding* and *Difference coding*.

DCT based coding of LL sub-image: We code the LL sub-image by using the lossy DCT algorithm and compensate the information loss by the “error image” as in Section 4.1.1. The encoder extracts the “error image” by the structure defined in Fig.4.2. DCT coefficients are coded by the algorithm explained in Sections 4.1.1 and 3.1.1 is run length coded as usual due to low

²Coding with nonrectangular decomposition

	LH	HL	HH	DCT	Error	R	H
Lena 512×512	3.40	2.69	4.69	5.27	1.50	2.30	3.10
Angio 512×512	3.83	3.75	12.86	4.70	1.85	2.94	3.89
Chest 1024×1024	4.35	8.34	62.93	17.48	1.72	3.98	6.04

Table 4.3: DCT based results for rectangular decomposition

	LH	HL	HH	LL	R	H
Lena 512×512	3.40	2.69	4.69	1.20	2.34	2.76
Angio 512×512	3.83	3.75	12.86	1.53	3.17	3.65
Chest 1024×1024	4.35	8.34	62.93	1.47	3.82	5.39

Table 4.4: Difference based results for rectangular decomposition

information nature of it. These two images (“DCT coded” and “error image”) together reconstructs the original LL sub-image exactly. We did similar experiments on the same images which are shown in Fig.4.4 and, 4.5 and the results presented in Table 4.3 are obtained.

The results are about 14.8% better than the results of “nonrectangular decomposition”.

Difference coding of LL sub-image: The same procedure of the Section 4.1.1 is applied, that is we shift and subtract the LL sub-image from itself and variable length and run-length code the “difference image”. The results of the performed experiments is described in Table 4.4.

We note that both in rectangular and nonrectangular subband decompositions the LL sub-image coding results are the same. This is due to the fact that in both cases LL is rectangularly lowpassed and downsampled version of the original image. So, the main difference between the two methods come from the coding of other sub-images such as LH, HL etc.

4.1.3 Pyramidal Coding

Pyramidal coding is a multirate coding scheme as described in Section 2.1.2. The image $x(n)$ is low pass filtered and down sampled by $2I \downarrow$ to $x_b(n)$. The 2-D low pass filter is a separable anti-aliasing filter and obtained by the multiplication of 1-D low pass filters as explained in Equation 2.32. The down

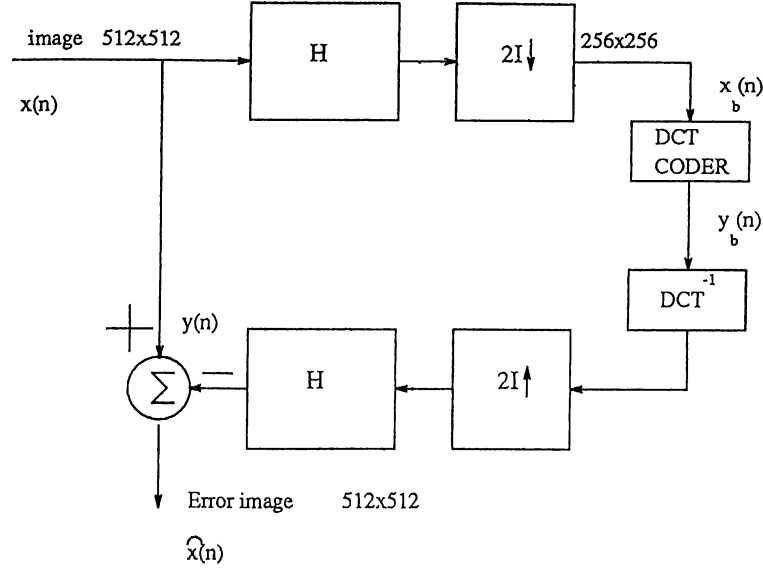


Figure 4.3: Pyramidal coding with lossy DCT

sampling matrix is

$$2I = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \quad (4.1)$$

and hence the resolution of the image is reduced by 4 ($|det(2I)|$). This in return also reduces the size of the image by 4. In other words an image $x(n)$ of size 512×512 reduces to an image $x_b(n)$ of size 256×256 as shown in Fig.4.3.

This low resolution sub-image is very similar to the LL sub-image of the previous subband coding methods. So it carries a lot of information in many typical images. We code $x_b(n)$ by the DCT algorithm. The compressed image is denoted by $y_b(n)$ in Fig.4.3. DCT coding introduces some loss to this low resolution image $y_b(n)$. We upsample the DCT coded sub-image $y_b(n)$ by $2I \uparrow$ and low pass filter it with the same filter of the down sampling block, to $y(n)$ whose size is the same as the size of the original image. The reconstructed image $y(n)$ is not an exact replica of the input image $x(n)$ due to DCT coding and multirate signal processing operations. We obtain an “error image” $\hat{x}(n)$ which is the difference between the original image and the reconstructed image. The “error image” carries very little information and dominated with consecutive zero samples so, can be very effectively run-length coded. In lossy image coding the “error image” is coded by a dead-zone quantizer to increase the efficiency of the run-length coding. In our lossless coding method to achieve perfect reconstruction at the receiver we do not quantize the “error image”.

The “error image” is dominated with low amplitude samples and many consecutive zeros, hence we use “modified Huffman” codebook concept for coding the nonzero samples. i.e., Coding of the “error image” is carried out as

	DCT	Error	R	H
Lena 512×512	2.87	1.48	1.31	1.81
Angio 512×512	5.13	1.71	1.58	2.10
Chest 1024×1024	17.56	1.68	1.64	2.41

Table 4.5: Pyramidal compression results

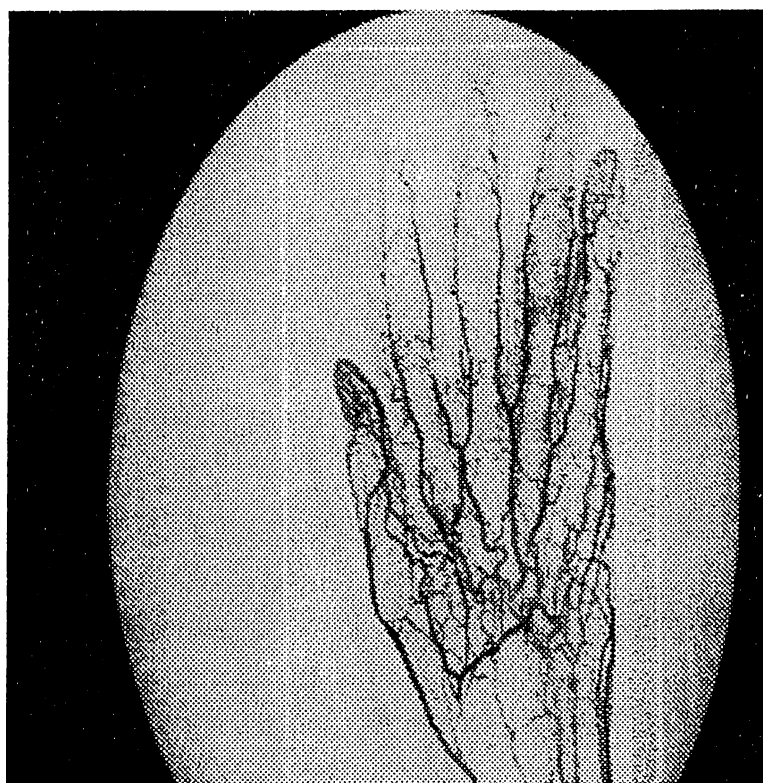


Figure 4.4: The sample image “Angio”

the coding of detail images LH, HH, etc. of subband decomposition structures.

The results obtained with the pyramidal coding algorithm are summarized in Table 4.5

Although the results are promising they are no better than the subband coding results. By comparing Tables 4.1-4.4 with Table 4.5 we observe that pyramid coding method is inferior to the subband decomposition structures.

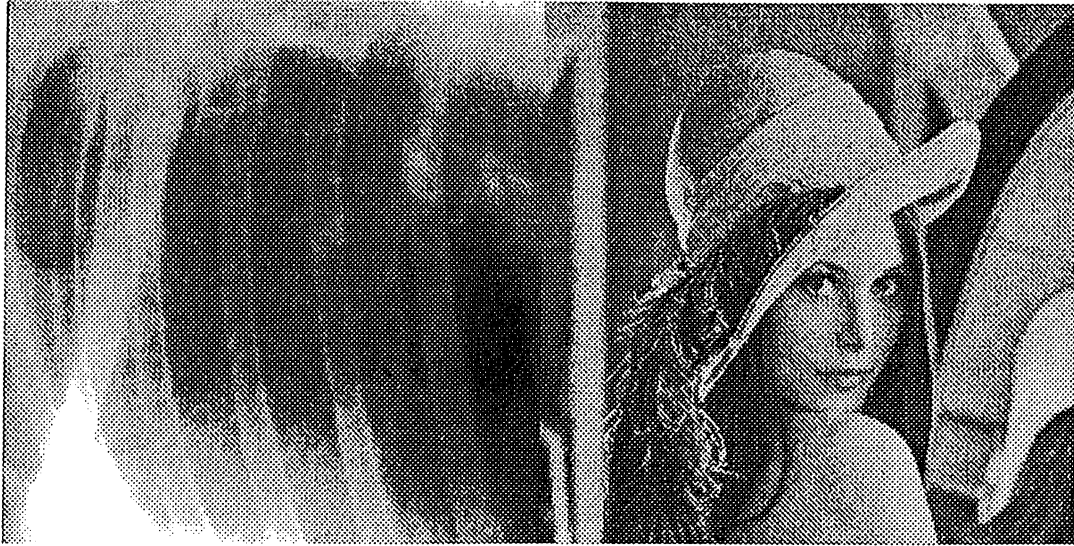


Figure 4.5: $1/4^{th}$ of the sample image “Chest” and “Lena”

4.2 Conclusion

Picture Archiving and Communication Systems (PACS) have recently been deployed in radiology departments. A PACS requires huge amounts of data to be stored in different storage media. The capacity of the memory of a PACS must be used as efficient as possible. The compression of medical images must be carried out in order to reduce the memory size. There are image coding methods which reduces the storage size of an image by an amount of $1/10$ without any visual degradation. But in some medical applications lossless coding of medical images is necessary. Although lossless coding can not achieve high compression ratios as lossy coding, any PACS should have the capability of storing medical images without introducing any loss.

Multiresolution coding techniques have been used for lossy image and speech coding, in this thesis we developed multiresolution techniques for the lossless image compression. We observed that the use of multiresolution techniques in lossless compression is as much advantageous as in lossy compression schemes. The main application area of the new technique is the coding of medical images.

The most popular multiresolution signal representation method, *subband decomposition*, plays a key role in our compression algorithms. We applied two different subband decomposition methods, rectangular and nonrectangular, in conjunction with DCT coding to compress the medical images. Although the compression schemes are adaptive and image dependent, we observe that the rectangular decomposition results are a little better than the nonrectangular decomposition.

The results presented in Tables 1 to 4 in Chapter 4 are about 6.3% superior than the ones recorded in the work of M. Rabbani and P.W. Melnychuk [8]. The well known image “Lena 512×512” is used to compare our results with the previous studies of the same work reported in the survey of [8]. For this specific image the best result is obtained in the difference coding of the rectangular decomposition in Table 4 which is 11.4% better than the best result of [8].

Another popular multiresolution method “pyramidal decomposition” is also considered. The results of Table 5 are inferior to the subband coding based methods (Tables 1 to 4). This is due to the fact that pyramidal decomposition produces more samples than the number of samples of the image at each decomposition stage.

The results presented in this thesis are obtained with fixed code books of Fig.3.5 and 4.1 and can be improved about 5% by use of an adaptive code book. But adaptive code book assignment has the disadvantage of extra information for decoding at the receiver.

Arithmetic coding of LH, HL, HH and H sub-images can be considered to improve the results because arithmetic coding can produce results which arbitrarily closely approach the entropy value .

The algorithms proposed in this thesis can easily be extended to the 3-D compression of medical images by only adding an extra motion detection algorithm.

References

- [1] Wen-Hsiung Chen, William K. Pratt, "Scene Adaptive Coder", *IEEE Trans. Commun.*, Vol. Com-32, pp 225-232, March 1984.
- [2] H.K. Huang, Shih-Chung B. Lo, "Radiological image compression using error-free, irreversible two dimensional direct-cosine-transform coding techniques", in *J. Opt. Soc. Am. A*/Vol.4, No.5, May 1987.
- [3] M. Vetterli, "Filter Banks allowing for perfect reconstruction", *Signal Processing*, Elsevier Science Publishers, North Holland, pp.1-26, 1986.
- [4] R. Ansari, A. E. Cetin, S. Lee, "Subband Coding of images using Non-rectangular Filter Banks", *Applications of Digital Image Processing XI*, SPIE, 1988, pp.315-323.
- [5] R. Ansari, Didier Le Gall, "Advanced television coding using exact reconstruction Filter Banks" in the book by J. Woods, Bellcore, Morristown.
- [6] J. Woods, Sean O'Neil, "Subband Coding of images". *IEEE Trans. on ASSP*, Vol-34, Oct 1986.
- [7] A.K. Jain, "Fundamentals of Digital Image Processing", *Prentice Hall, Englewood Cliffs*, NJ 07632, 1989.
- [8] M. Rabbani, Paul W. Melnychuk, "Survey of lossless image coding techniques", *Hybrid Imaging Division, Photographic Res. Lab.*, Eastman Kodak Company, Rochester, New York.
- [9] D.J. Le Gall, H.P. Gaggioni and C.T. Chen, "Transmission of HDTV Signals under 140 Mbits/s using Subband Decomposition and Discrete Transform Coding", *Proc. Second Int. Workshop Sig. Proc. of HDTV*, L'Aquila, Italy, Feb.29-March2, 1988.
- [10] R. Ansari, A. Fernandez and S.H. Lee, "HDTV Subband/DCT Coding Using IIR Filter Banks: Coding Strategies", *Proc. SPIE Conference on Visual Communication and Image Processing*, Philadelphia, PA, vol.1199, 1291-1302, Nov.1989.

- [11] R.Ansari, C.Guillemot, "Exact Reconstruction Filter Banks using Diamond FIR Filters", *Proc. 1990 Bilkent Int.Conf.*, Ankara, Turkey, pp.1412-1424, July 1990.
- [12] R.E.Crochiere, "Subband Coding", *The Bell System Technical Journal*, Vol.60, No.7, Sept.1981, USA.
- [13] J. Woods, *Subband Image Coding*, Kluwer Academic Publishers, 1990.
- [14] R.G. Gallager, "Information theory and reliable communication", *John Wiley and Sons*, 68-26850, 1968.
- [15] Wozencraft and Jacobs, "Principles of communication engineering", *John Wiley and Sons*, 65-16429, 1965.
- [16] J.H. Shapiro and A.S. Willsky, "Stochastic processes and systems", *MIT EEE dept lecture notes*, Spring 1987.
- [17] D.E. Dudgeon and R.M. Mersereau, "Multidimensional digital signal processing", *Prentice-Hall, Inc., Englewood Cliffs, New Jersey*, 07632, 1984.
- [18] Didier le Gall, "MPEG Standards", *Communications of the ACM*, April 1991, Vol.34, No.4, pp 47-57.
- [19] M.J. Narasimha and A.M. Peterson, "On the computation of discrete cosine transform", *IEEE Trans. Com.*, COM-26, No6, June 1978, pp 934-936.
- [20] R.J. Schalkoff, "Dynamic or time-varying image analysis", *John Wiley and Sons Inc., Canada*, 89-30954, 1989.