WAVELET TRANSFORM IN IMAGE COMPRESSION

by

RICHARD ANDREW MUYSHONDT, B.S.

A THESIS

IN

ELECTRICAL ENGINEERING

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

Approved

December, 1995

ACKNOWLEDGMENTS

Foremost, I would like to thank God for all of His love, understanding, and help. I would also like to express my most heartfelt appreciation to my loving wife Jennifer for all of the love, support, and confidence she gave me while I completed my M.S.E.E. A great, big "thank you" also goes to my parents Enrique and Berta Muyshondt, my parents-in-law John and Dianne Van Heerde, and my brothers and sister for all of their guidance, love, and support. I could not have done this without any of you.

My sincere appreciation also goes out to my graduate advisor Dr. Sunanda Mitra for all of her help, wisdom, and encouragement during my research. Dr. Mitra is not just an outstanding graduate advisor, but is my mentor and friend.

Finally, I also would like to give my gratitude to Dr. Thomas Krile and Dr. Thomas Trost for serving on my graduate committee and for helping me with my thesis.

TABLE OF CONTENTS

ABSTRACT

The past few years have seen a rapid development in the areas of image compression techniques. The evolution in image compression is mainly attributed to the need of rapid and efficient techniques for the storage and transmission of data among individuals. In order to achieve maximal storage and transmission capabilities, different compression algorithms should be compared in order to find an optimal technique for medical image compression. In this research, we studied the performance of different wavelet basis functions and of a Wavelet Transform (WT) image coding algorithm. We characterized the performance of the wavelet coefficients and the coding algorithm by calculating the mean square error, peak signal to noise ratio, and root mean square signal to noise ratio of the reconstructed images. In addition, we compared the WT algorithm to the current JPEG standard. We performed comparisons on standard as well as radiographic images using the above criteria in order to judge the compression characteristics of both techniques. In addition, we show that the Adelson 15 wavelet coefficients perform better than Daubechies 4 and 12 wavelet coefficients for image compression. Furthermore, we show that JPEG and the WT algorithm had comparable performance in standard image compression, but the WT algorithm outperformed JPEG in all the above criteria and proved to be a versatile coding technique for radiographic images.

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Objectives

The medical field has experienced an increased need for fast and efficient coding algorithms due to the rise of telemedicine and the use of digital medical images [1]. This demand has led to the development of several techniques which have revolutionized the area of image compression. However, the development of so many techniques has given rise to the dilemma of deciding which of these methods possess the best properties and potentials for effective coding. This problem is of particular importance in the medical field where the distortion of information may lead to inaccurate diagnosis. Thus, there exists a need to compare the different coding algorithms in order to see the advantages of each technique. Since much attention has recently been gained by the wavelet transform [2, 3, 4, 5, 6], we chose to investigate the properties of this transform-based compression algorithm and compare them to the JPEG standard.

1.2 Chapter Outline

This thesis examines the coding properties of the wavelet transform and compares its performance with the JPEG standard on various types of digital images. Chapter 2 explains the basic terminology in image compression and gives an overview

of different image compression techniques. Chapter 3 discusses the Joint Photographic Experts Group (JPEG) compression technique. Chapter 4 shows an in-depth look at the mathematical properties of the wavelet transform, and Chapter 5 describes the wavelet-based compression algorithm. Chapter 6 explains the performance measurement used to evaluate the wavelet and JPEG-based coding techniques. Chapter 7 contains the results and discussion of this research. Finally, Chapter 8 has the conclusions of this work, and a discussion of future work.

CHAPTER 2

IMAGE COMPRESSION TECHNIQUES

2.1 Terms and definitions

In order to have a good understanding of the wavelet-based image compression techniques, certain terms and definitions need to be introduced. A digital image can be described as a sampled version of an analog picture where each sample point is called a pixel. Each pixel contains a location or coordinate value and scale values that indicate the brightness level and color of the particular pixel. In an eight bit gray level image, only one scale value per pixel is needed. This pixel level value can range between 0 and 255 where 0 is black, 255 is white, and the other numbers represent different shades of gray. On the other hand, a color picture requires more pixel scale values in order to represent its color. The pixel values are usually divided into red (R), green (G), and blue (B), and the image is said to be an RGB color image. Each scale value is also variable between a given range. For example, in a 24 bit color RGB image, each scale value of R, G, and B has values between 0 and 255 where the scale value represents a different shade of that color. By varying the R, G, and B values, one can achieve up to 16 million different colors.

The images used for this thesis were 512x512 8 bits per pixel (bpp) gray level images. The justifications for using this image format are as follows. First, the 512x512 8 bpp images are readily available, and the 512x512 8 bpp is the standard,

accepted format used in the image compression field. Second, the coarser image resolution decreases the computational time and complexity for the compressions. Thus, more compressions could be performed. Finally, the 8 bbp gray level format was chosen over a 24 bit color format since the color format is merely a three layer 8 bbp format where each layer represents a color plane. Since the gray level 8 bpp can be thought of as just having one color plane (e.g., gray plane), its compression can be simply related to a 24 bpp color RGB image.

For the above reasons, the original radiographic images used in this work were down-sampled to a 512x512 resolution. Although most radiographic images are scanned at higher resolutions (e.g., 2kx2.5k) [1], the down-sampled image compressions provide a better comparison with the standard images. In addition, since the higher resolution images are more correlated, the compression ratio of the higher resolution radiographic images should be larger. Thus, the radiographic compression ratios found in this thesis are conservative in nature.

The following is a list of terms and equations used in this thesis.

Compression Ratio (CR): the ratio of the total number of bits needed to code the original image to the total number of bits needed to code the compressed image. Thus, the CR is

$$CR = \frac{B_o}{B_c} \qquad (2.1)$$

where B_o is the total number of bits needed to code the original image data and B_c is the total number of bits needed to code the compressed data [7]. A higher numerical CR value indicates that fewer bits were needed to code the compressed image.

Relative Data Redundancy (R_D): reflection of the amount of redundancy in a given digital image. The R_D is defined as

$$R_D = 1 - \frac{1}{CR}. \qquad (2.2)$$

The redundancy can further be divided into three categories [8]. The first is coding redundancy. This refers to the redundancies that occur when a digital image is coded. For example, the probabilities of certain gray level values in images vary. By taking advantage of these statistical properties of an image, the image can be compressed (e.g., Huffman coding). Another kind of redundancy is interpixel redundancy. This redundancy reflects the predictability of neighboring pixel values. That is to say, it shows how well one can predict the gray level value of a given pixel's neighbors. Techniques such as predictive coding can be used to eliminate this redundancy. The third type of redundancy is psychovisual redundancy. Psychovisual redundancy refers to the information which is relatively less important in visual image processing. The exclusion of this information will not greatly affect the perceptual interpretation of the digital image. Compression is achieved by reducing any of these types of redundancy.

Information I(E): negative log of the probability of an event E. That is

$$I(E) = -\log P(E) \tag{2.3}$$

where P(E) is the probability of event E occurring [8]. Thus, if the event E always occurs (i.e., has a probability equal to one), the information given by E is zero. No information is given to the user by telling him that event E has occurred.

Entropy or Uncertainty (H): the average information per source output. Entropy is defined as

$$H = -\sum_{j=1}^{J} P(a_j) \log P(a_j). \tag{2.3}$$

Larger magnitudes of H indicate that the output source has more entropy or uncertainty [8]. Thus, each output provides the end user with more information.

2.2 Purpose

An average 512X512 8 bpp gray level image requires a total of 262144 bytes for representing its visual information. Although by today's standards the storage of such an image is not unusual, the archiving and transmission of digital images is a problem of immense magnitude. This is of genuine concern in the medical field where the typical digital radiographic image is on the order of 2kX2.5k [1] and requires at least 8 bits per pixel (bpp). In addition, many of these images are represented in color which means an additional requirement of 24 bpp. Thus, in order to store a color

radiographic image, the user must have at least 15 megabytes (MB) of electronic storage space. It is plain to see that the efficient archiving and fast transmission of medical images is problematic by nature.

For the foregoing reasons, the area of image compression has experienced a rapid growth. The objective of image compression techniques is to optimally reduce the number of bits needed to store the information in a digital image. Since most digital images contain a large amount of inter pixel correlation, they utilize a superfluous quantity of data to represent a specific amount of information. By capitalizing on this redundancy, image compression techniques manipulate the digital image data to represent the visual information in the most efficient manner.

However, before analyzing each compression technique, it must be said that compressions may be either lossless or lossy. In a lossless compression approach, the original and reconstructed images are identical. No information is lost during the compression and reconstruction stages. In lossy compression, on the other hand, the coding process makes certain approximations or quantizations in order to better compact the information. Although better compression ratios are achieved, the trade-off is in a reconstructed image which is not identical to the original image in the sense of individual pixel values. However, the information presented may still appear to be visually identical. In fact, herein is the key to achieve higher compression ratios in lossy compression techniques. Its objective is to take any means to reduce the amount of data needed to represent a certain amount of visual information. Its emphasis,

however, is not on the exactness of the pixel values but rather on the exactness of the visual quality of the reconstructed image.

2.3 Pixel Coding

Different approaches exist to effectively reduce the amount of data needed to depict the visual information in an image. The first of these approaches is pixel coding.

2.3.1 Run Length Coding

Run length coding (RLC) achieves compression of an image by decreasing the spatial redundancies associated with that image [9, 10]. In particular, one dimensional RLC codes an image by defining run lengths (RL) of pixel values (PV) where a run length is the number of successive pixels which have the same value. Thus, in order to run length code an image, the image is scanned row by row, and the pixel values of the image are replaced by (RL, PV) [8, 9, 10]. For example, for the following excerpt of a gray level image row

12 16 16 16 16 15 15 15 15 15 250 11 11 11 11 11 11 11 11 2 2 2 2 2 2,

the run length code is represented as

(0,12) (4,16) (5,15) (0,250) (8,11) (6,2).

Since the RLC depends heavily on the spatial redundancy of an image, it is possible to actually increase the size of the stored image. This is especially true for an image that has much detail. To mitigate this possible problem, the following algorithm is used:

1. A run-length is identified by having all of its last 3 most significant bits set to 1. Its lower 5 bits provide a repeat count from 1 to 31 for the byte that follows.

2. If a pixel value has zero run length, then it can be coded as follows:

   a. If its last 3 bits are all set to 1, i.e. >= 224, then it is coded as (11100000,pixel value), i.e., zero run-length followed by pixel value.

   b. Otherwise, it is coded as pixel value. (Ahmed, 1995, p. 400)

Therefore, the actual binary code for

(0,12) (4,16) (5,15) (0,250) ...

would be

PV=12	RL=4	RV=16	RL=5	PV=15	RL=0	PV=250 ...
00001100	11100100	00010000	11100101	00001111	11100000	11111010 ...

where the spaces are only added for the code's readability.

The above 1-D algorithm can be further extended to a 2-D algorithm where the neighboring top, bottom, right, and left pixels are examined for RL. However, in order to decrease computational time, the 1-D algorithm is used in this thesis.

2.3.2 Huffman coding

The pixel values in an image are typically stored as equal length words. For example, in an 8 bpp gray level image, each pixel value is stored in 8 bits (i.e., one

byte). If the probabilities of these pixel values are calculated, it is found that every image has a statistical distribution of pixel values, and thus, a certain entropy. It is possible to exploit this statistical probability and reduce the average number of bits needed to store the pixel values. This decreases the entropy of the image. In 1951, Huffman accomplished this reduction by creating an algorithm which assigns shorter bit codes to pixel values that have higher probabilities and assigns longer bit codes to pixel values that have lower probabilities [8].

The Huffman (or Entropy) coding algorithm assigns the optimal bit length code to each pixel value by following a few simple steps.

1. Calculate the probabilities of every pixel in the image.
2. Arrange the probabilities of the pixels in a descending order.
3. Sum the last two smallest probabilities and rearrange it and the other probabilities in descending order. The summation of probabilities is referred to as a source reduction.
4. Continue performing the source reductions until only two probabilities are left.
5. Assign these last two probabilities the symbols 0 and 1. The choice is arbitrary.
6. Working backwards through one source reduction, add the symbols 0 and 1 to the pre-summed probabilities of the last source reduction. Again, the choice is arbitrary.
7. Continue working backwards one reduction at a time until the original probabilities are reached.

The code assignment achieved by this algorithm will approach the entropy of the image [8, 9, 10].

2.3.3 Scalar quantization

Scalar quantization is the process of mapping the pixel values of an image into a smaller number of bins via a parameter called the binsize. Binsize refers to the size or number of pixel values that will be mapped into one bin. For instance, if a binsize of ten is chosen for mapping an 8 bpp gray level image, then the pixel values 0 to 9 would be mapped into bin 0. Similarly, the pixel values 10 to 19 would be mapped into bin 1, 20 to 29 into bin 3, etc. See Figure 2.1 for a pictorial example. Thus, the image would now be represented by 26 bins instead of 256 gray levels. In addition, this type of pixel mapping increases the correlation of the image. Therefore, scalar quantization allows the image to be further compressed.

Excerpt of a 8 bpp gray level image	scalar quantized excerpt with a binsize=10
13 15 17 19 29 26 23 58 55 57 57	1 1 1 1 2 2 2 5 5 5 5

Figure 2.1. Example of scalar quantization of an excerpt 8 bpp gray level image

2.4 Predictive coding

If an image is highly correlated, then the pixels in a particular neighborhood should have similar values. Predictive coding takes advantage of this correlation by coding the difference between the current pixel value and the preceding pixel value.

Thus, if both pixels have similar values, the error to be coded is very small, and the number of bits per pixel needed to code the image is reduced. Algorithms such as differential pulse code modulation (DPCM) and delta modulation (DM) are based on the above concepts and can reduce the coded data of an 8 bpp image to 1-2 bpp with good quality [9, 11]. For an in-depth look at predictive coding, refer to references [8], [9], and [11].

2.5 Transform coding

The goal of transform coding is to decorrelate the image pixels by applying a linear, reversible transform to the image [8, 11]. The transform maps the image pixels to transform coefficients that are then quantized and coded to achieve compression. In general, the quantization of the transform coefficients is performed selectively. That is, transform coefficients which contain the most information are carefully coded while the coefficients that contain the least amount of information are coded coarsely [8].

The typical transforms used for coding are the Karhunen-Loeve transform (KLT), discrete Fourier transform (DFT), discrete cosine transform (DCT), Walsh-Hadamard transform (WHT) [8, 9], and wavelet transform (WT). The KLT is the optimal transform in decorrelating the pixels in an image. However, it is also data dependent and computationally intensive to implement. Although the WHT is simple to implement, the DFT and DCT are the closest in decorrelation capabilities to the KLT [9]. However, since the DCT only computes real coefficients, is not data

dependent, and is less computationally intensive, it is the most commonly used transform in image compression and is the current transform used in the JPEG standard [8].

For an in-depth look at the DCT and WT, see Chapters 3 and 4.

2.6 Other coding methods

Other coding methods such as vector quantization and hybrid coding should be mentioned for completeness.

2.6.1 Vector quantization

Vector quantization (VQ) is similar to scalar quantization except that the mapping is performed on vectors or blocks of pixels rather than on individual pixels. The general VQ algorithm has three main steps. First the image is partitioned into blocks which are usually 2X2 or 4X4 in size. After blocking the image, a codebook which best estimates the blocks of the image is constructed and indexed. Finally, the original image blocks are substituted by the index of best estimate code from the codebook [7, 12, 13].

2.6.2 Hybrid coding

Most compression algorithms achieve maximum compression by combining two or more of the above coding algorithms [11]. The combined coding algorithm is

referred to as a hybrid coding algorithm, and an example of such an algorithm is shown in Figure 2.3. Note that this algorithm contains three steps. However, more or less steps could be included in a general hybrid algorithm [9].

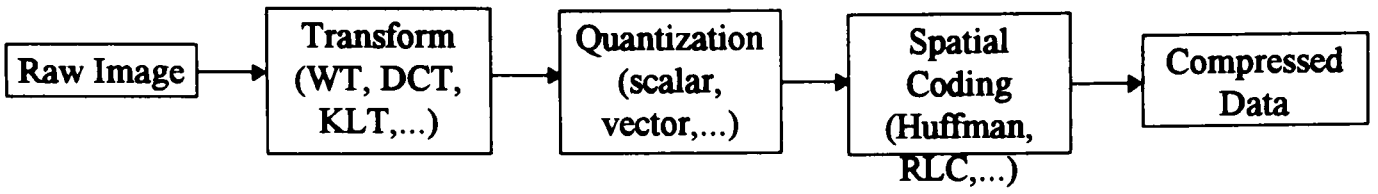


Figure 2.2. Typical hybrid coding steps

CHAPTER 3

JOINT PHOTOGRAPHIC EXPERTS GROUP (JPEG) COMPRESSION

The JPEG compression algorithm was originally created in order to serve as a standard in the image compression area. This compression algorithm was made to be simple and efficient in its implementation. To accomplish these goals, the JPEG algorithm relies on the discrete cosine transform (DCT) to concentrate the energy of an image and decorrelate its pixels. The following two sections will give an overview of the DCT and of the JPEG algorithm.

3.1 Discrete cosine transform

The DCT is an orthogonal transform which has excellent energy compacting capabilities. In 2-D, it is defined as [8, 14, 15]:

$$C(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (3.1)$$

for u,v = 0,1,2, ... ,N-1

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \alpha(u)\alpha(v)C(u,v)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (3.2)$$

for x,y = 0,1,2, ... ,N-1 and

$$\alpha(u) = \sqrt{\frac{1}{N}} \quad \text{for } u = 0, \quad (3.3)$$

$$\alpha(u) = \sqrt{\frac{2}{N}} \quad \text{for } u = 1, 2, \ldots, N\text{-}1, \tag{3.4}$$

$$\alpha(v) = \sqrt{\frac{1}{N}} \quad \text{for } v = 0, \tag{3.5}$$

and

$$\alpha(v) = \sqrt{\frac{2}{N}} \quad \text{for } v = 1, 2, \ldots, N\text{-}1. \tag{3.6}$$

3.2 JPEG compression algorithm

The JPEG algorithm is the current image compression standard. Compression is accomplished in 5 main steps. These are as follows [14, 15]:

1. The original image is partitioned into 8x8 blocks, and each block is transformed using the DCT
2. The DCT coefficients are normalized via a pre-defined normalization array. The normalized coefficients are then uniformly quantized by rounding to the nearest integer.
3. The quantized DC coefficients are encoded with a lossless Differential Pulse Code Modulation (DPCM) scheme.
4. A zigzag coding approach is then used to code the AC coefficients.
5. Huffman coding is utilized to code the above results.

For further details on the JPEG compression algorithm, see Rabbani [14] and Pennebaker [15].

The JPEG compression algorithm was the first standard in the field of image compression. During its conception, the algorithm was fitted to the technology of that time. Thus, since computational power was limited, the compression algorithm used 8x8 block partitions to ease the computational load of the transform. Subsequently, it was widely studied, and the JPEG coding technique was improved to achieve the best compression. In addition, hardware and software are readily available which perform the compression and reconstruction of images in an expedient manner.

Although the JPEG algorithm is simple to implement, quick to perform, and is widely available, it has many disadvantages. In particular, due to the segmentation of the image into 8x8 blocks, the JPEG compressed images contain noise which is called the blocking artifact. The blocking artifact, which is also known as the blocking effect, is present in all JPEG compressed images but is most prevalent in images that have been highly compressed. Since the algorithm relies on the 8x8 partitioning of the image to perform the DCT, it is impossible to circumvent this problem. In addition, the JPEG compression algorithm seems to have a low upper bound on its ability to compress images. Thus, one can only achieve a low maxima in compressing an image. Finally, the JPEG algorithm is not well suited for multiresolution decompositions. This multiresolution capability is of great value for fast and efficient transmission of images from one computer to another where the end user may need only certain resolutions of the digitized image or may need to zoom in and out of the image. It

should be noted, however, that the progressive JPEG algorithm [15] uses progressive transmissions of better quantized images to facilitate rapid transmissions.

CHAPTER 4

WAVELET TRANSFORM

Transform coding of images is performed by the projection of an image on some basis. The basis is chosen so that the projection will effectively decorrelate the pixel values, and thus, represent the image in a more compact form. The transformed (decomposed) image is then quantized and coded using different methods such as scalar and vector quantization, arithmetic coding, run length coding, Huffman coding, and others.

This section gives a brief explanation of the WT. Section 4.1 contains a description of the mathematics used by the WT. Section 4.2 explains how Daubechies defined and found her wavelet coefficients.

4.1 Mathematical Background

The best way to introduce the wavelet transform mathematical derivation is to review the familiar Fourier transform. The continuous time Fourier transform (CTFT = $F(\omega)$) is defined as the inner product or projection of the function f(x) onto the basis function $e^{-i\omega x}$. In mathematical terms, the CTFT [8, 9] is

$$CTFT: F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx = \left\langle e^{-i\omega t}, f(x) \right\rangle. \quad (4.1)$$

where < , > denotes the inner product, x is the location variable, and ω is the frequency variable. The inner product in equation 4.1 denotes the Fourier coefficients. Therefore, a function f(x) can be written exactly as

$$f(x) = \int_{-\infty}^{\infty} F(\omega) e^{i\omega x} d\omega \qquad (4.2)$$

Equation 4.2 says that f(x) is the integration of $e^{i\omega x}$ multiplied or weighed by the appropriate Fourier coefficient.

Although the Fourier transform is a very strong mathematical and analytical tool, it has some properties which are not desirable for image compression. As can be seen from equation 4.2, all past and future time information is needed to accurately reconstruct a signal f(x). If all this time information is not known, the reconstructed signal will only be an approximation of the original. In addition, in order for the Fourier transform not to have an infinite number of coefficients, the signals being transformed must be stationary. Since the Fourier transform can be written as the projection of the signal onto cosines in the real plane and sines in the imaginary plane, discontinuities in a signal would only be represented with many sine and cosine terms. Therefore, the decomposition of a discontinuous signal would contain many coefficients which are spread out over the entire frequency axis [5].

In order to mitigate this coefficient frequency spreading, the CTFT was modified to form the short time Fourier transform (STFT). The STFT, which is sometimes referred to as the windowed Fourier transform, is similar to the CTFT

except for the addition of a window function (w) that limits the existence of the transform. The window function is usually a Gaussian, and its shifting over the time axis results in a time frequency description of the signal. In general, the STFT can be mathematically written as

$$STFT: F(\omega, b) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} w(x - b)\, dx \quad (4.3)$$

where x is the location variable, ω is the frequency variable, w() is the window function, and b is the location shifting parameter for the window function [5].

The STFT eliminates the need for complete past and future signal information. However, the STFT still contains a drawback. The window function used in the transform has a fixed window size. Therefore, it cannot adapt to the characteristics of signals at certain points. That is, it does not give a good description of signals with widely changing frequency spectra.

The continuous time wavelet transform (CTWT) mitigates the limitations of the CTFT and STFT by having a basis function which can be both shifted and dilated or contracted. The CTWT basis function is

$$\psi_{a,b}(x) = a^{-\frac{1}{2}} \psi\left(\frac{x - b}{a}\right), \quad (4.4)$$

where ψ(x) is a zero mean band pass function, and the transform is defined as

$$CTWT(a, b) = a^{-\frac{1}{2}} \int_{-\infty}^{\infty} \psi\left(\frac{x - b}{a}\right) f(x)\, dx = \left\langle \psi_{a,b}(x), f(x) \right\rangle \quad (4.5)$$

where $\Psi_{a,b}$ are the wavelets basis function, a is the scaling factor, and b is the shifting factor [5].

$\Psi_{a,b}$ are real and oscillatory and fade away when they approach plus or minus infinity [16]. Ψ itself is called the mother wavelet, and the shifted and scaled conglomeration of Ψ form an orthonormal basis which as mentioned above are called wavelets. A close examination of equation 4.5 shows that increasing 'a' causes Ψ to be stretched, and thus, the formed wavelets can act as low frequency windows. On the other hand, decreasing 'a' causes Ψ to be shrunk, and thus, these wavelets act as high frequency windows. Consequently, one can narrow and widen the time-frequency window as well as shift it over the time domain via 'b' in order to adjust to the frequency characteristics of any signals. In addition, due to the decay property of the wavelets, one does not need all past and future signal information in order to accurately represent any signal [2, 3, 4, 5].

In general, any signal f(x) can be represented as

$$f(x) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{0}^{\infty} CTWT(a,b) a^{-\frac{1}{2}} \psi\left(\frac{x-b}{a}\right) \frac{dadb}{a^2} \qquad (4.6)$$

where a, b are continuous and C_Ψ is a constant. Since there are no restrictions on a and b, there exist infinite combinations of possible wavelets. In order to reduce this redundancy, a and b are discretized by letting

$$a = a_0^{\,m} \qquad (4.7)$$

and

$$b = n a_0^{\,m} b_0 \qquad (4.8)$$

where n, m, a_o, and b_o are constants. By substituting these equalities into equation 4.4, the basis functions now become

$$\psi_{m,n}(x) = a_0^{-\frac{m}{2}} \psi\left(a_0^{-m} x - n b_0\right), \qquad (4.9)$$

and the new "sampled" CTWT, now called the continuous time wavelet series (CTWS), is defined as [2, 3, 4, 5, 6]

$$CTWS_{m,n}(x) = d_{m,n} = \int_{-\infty}^{\infty} \psi_{m,n}(x) f(x) dx = \left\langle \psi_{m,n}(x), f(x) \right\rangle \qquad (4.10)$$

and

$$f(x) = \sum_m \sum_n d_{m,n} \psi_{m,n}(x) = \sum_m \sum_n \left\langle \psi_{m,n}(x), f(x) \right\rangle \psi_{m,n}(x). \qquad (4.11)$$

As mentioned above, wavelets are zero mean band pass functions and have zero DC coefficients. Thus, in order to completely represent a function f(x) via a pure wavelet expansion, one needs an infinite number of wavelet resolutions (see equation 4.11) [6]. This is an impracticality in image compression where the goal is to reduce the amount of data needed to represent the total image.

To avoid using an infinite number of coefficients for the wavelet representation, f(x) is represented by a low pass or low resolution version, $f_j(x)$ and a detail version, D. The detail version D is simply a finite wavelet representation. The low pass version, on the other hand, is a result of the inner product between f(x) and a scaling function $\phi(t)$. Thus, if the scaling function weight coefficients are

$$c_{j,n} = \int_{-\infty}^{\infty} f(x) a_0^{-\frac{j}{2}} \phi\left(a_0^{-j}x - nb_0\right) dx = \left\langle \phi_{j,n}(x), f(x) \right\rangle \tag{4.12}$$

and the function at resolution j (i.e., $f_j(x)$) is

$$f_j(x) = \sum_{n=-\infty}^{\infty} c_{jn} a_0^{-\frac{j}{2}} \phi\left(a_0^{-j}x - nb_0\right) = \sum_{-\infty}^{\infty} \left\langle \phi_{j,n}(x), f(x) \right\rangle \phi_{j,n}(x), \tag{4.13}$$

then any arbitrary function f(x) can be represented as [2, 3, 4, 5, 6]

$$f(x) = \sum_{n=-\infty}^{\infty} c_{j,n} a_0^{-\frac{j}{2}} \phi\left(a_0^{-j}x - nb_0\right) + \sum_{m=-\infty}^{j} \sum_{n=-\infty}^{\infty} d_{m,n} a_0^{-\frac{m}{2}} \psi\left(a_0^{-m}x - nb_0\right) \tag{4.14}$$

$$= \sum_{n=-\infty}^{\infty} c_{j,n} \phi_{j,n}(x) + \sum_{m=-\infty}^{j} \sum_{n=-\infty}^{\infty} d_{m,n} \psi_{m,n}(x) \tag{4.15}$$

$$= f_j(x) + D. \tag{4.16}$$

Furthermore, by letting $a_o = 2$ and $b_o = 1$, one can pick special $\psi(x)$ such that for fixed m, $\psi_{m,n}(x)$ and $\phi_{m,n}(x)$ individually form an orthonormal basis. In addition, both $\psi_{m,n}(x)$ and $\phi_{m,n}(x)$ are orthogonal complements of one another, and $\langle \psi_{m,n}(x), f(x) \rangle$ represents the information lost by going to a lower resolution via $\langle \phi_{m,n}(x), f(x) \rangle$ [6].

Now that the wavelet representation of a function has been established by the above, one should be aware of the containment restrictions placed on both $\phi(x)$ and $\psi(x)$. Since the scaling function projects the function f(x) to a lower resolution plane (i.e., resolution j in the above discussion), it is easy to visualize that all of the information in any lower resolution is contained by the top resolution of the scaling function. Thus, the containment restriction on $\phi(x)$ is

$$\phi(x) = 2\sum_{n} h_0(n)\phi(2x-n) \tag{4.17}$$

where $h_o(n)$ are called interscale basis coefficients. Similarly, the containment restriction on $\psi(x)$ is

$$\psi(x) = 2\sum_{n} h_1(n)\phi(2x-n) \tag{4.18}$$

where $h_1(n)$ are the expansion coefficients.

In general, it can be proved that

$$h_n = 2^{\frac{1}{2}}\int \phi(x-n)\phi(2x)dx\ . \tag{4.19}$$

In addition if one defines

$$g_1 = (-1)^1 h_{-1+1} \tag{4.20}$$

then the scaling coefficients $c_{j,n}$ and wavelet coefficients $d_{m,n}$ can be represented by

$$c_{j,n} = \sum_{k} h_{2n-k} c_{j-1,k} \tag{4.21}$$

and

$$d_{m,n} = \sum_{k} g_{2n-k} c_{m-1,k}\ . \tag{4.22}$$

Equations 4.21 and 4.22 are used for the computation of the multiresolution wavelet transform, and h is a low pass filter and g is a band pass filter [2, 3, 4, 5, 6].

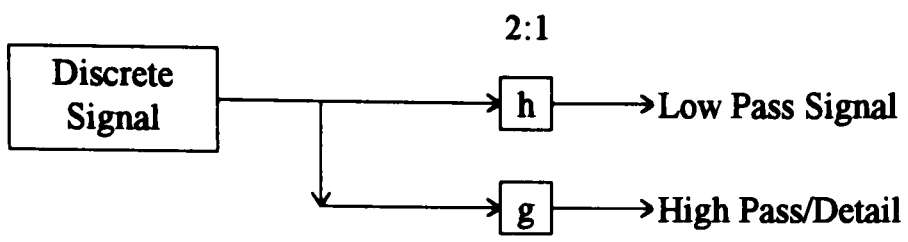


Figure 4.1. Implementation of the 1-D wavelet transfrom

For more detail on multiresolution wavelet transforms, see Mallat [3], Daubechies [2], and Barnard [5].

4.2 Derivation of Daubechies' coefficients

Daubechies found a general algorithm that finds different wavelet coefficients based on two properties. The first property is the orthogonality condition placed on the wavelet filters. Simply stated, the orthogonality condition requires that, given the wavelet filter W, then $WW^{-1} = 1$. The second property defined by Daubechies is that the filter g should have a zero response to a smooth data vector since g is a band pass filter. To get the zero response, g has to have a certain number of vanishing moments.

Using the above properties, Daubechies defined the h and g filters by keeping only the first n coefficients and arranging these coefficients in a matrix as follows:

$$W = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & & & & & & & & \\ g_0 & g_1 & g_2 & g_3 & & & & & & & & \\ & & h_0 & h_1 & h_2 & h_3 & & & & & & \\ & & g_0 & g_1 & g_2 & g_3 & & & & & & \\ & & & & & \cdots & & & & & & \\ & & & & & \cdots & & & & & & \\ & & & & & & h_0 & h_1 & h_2 & h_3 & & \\ & & & & & & g_0 & g_1 & g_2 & g_3 & & \\ h_2 & h_3 & & & & & & & h_0 & h_1 & & \\ g_2 & g_3 & & & & & & & g_0 & g_1 & & \end{bmatrix} \quad (4.23)$$

where the blank matrix entries represent zeros. Due to equations 4.21 and 4.22, W becomes

$$W = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & & & & & & & & \\ h_3 & -h_2 & h_1 & -h_0 & & & & & & & & \\ & & h_0 & h_1 & h_2 & h_3 & & & & & & \\ & & h_3 & -h_2 & h_1 & -h_0 & & & & & & \\ & & & & & \cdots & & & & & & \\ & & & & & \cdots & & & & & & \\ & & & & & & h_0 & h_1 & h_2 & h_3 & & \\ & & & & & & h_3 & -h_2 & h_1 & -h_0 & & \\ h_2 & h_3 & & & & & & & h_0 & h_1 & & \\ h_1 & -h_0 & & & & & & & h_3 & -h_2 & & \end{bmatrix} \quad (4.24)$$

By applying the orthogonality condition, one finds that

$$h_0^2 + h_1^2 + h_2^2 + h_3^2 = 1 \quad (4.25)$$

and

$$h_2 h_0 + h_3 h_1 = 0. \quad (4.26)$$

Since four coefficients were used, one then must set the first two moments of the matrix to equal zero. This condition yields

$$h_3 - h_2 + h_1 - h_0 = 0 \tag{4.27}$$

$$0h_3 - 1h_2 + 2h_1 - 3h_0 = 0. \tag{4.28}$$

Since there are four equations with four unknowns, Daubechies solved them to find her four wavelet coefficients that are referred to as DAUB4. These coefficients are

$$h_0 = \frac{1+\sqrt{3}}{4\sqrt{2}}, \tag{4.29}$$

$$h_1 = \frac{3+\sqrt{3}}{4\sqrt{2}}, \tag{4.30}$$

$$h_2 = \frac{3+\sqrt{3}}{4\sqrt{2}}, \tag{4.31}$$

and

$$h_3 = \frac{1-\sqrt{3}}{4\sqrt{2}}. \tag{4.32}$$

By increasing the number of coefficients and letting n moments go to zero, Daubechies found the values for 6, 8, 10, ..., 2^n wavelet coefficients. For an in-depth look at Daubechies work, the reader should see [2].

CHAPTER 5

WAVELET-BASED COMPRESSION ALGORITHM

This chapter gives a description of the WT-based compression algorithm. Section 5.1 describes the multiresolution representation of a function or image using orthonormal wavelets while Section 5.2 describes the WT coding algorithm.

5.1 Multiresolution or pyramidal decomposition

As discussed in Chapter 4, the WT represents any arbitrary function, f(x), in terms of two sets of basis functions $\psi_{m,n}(x)$ and $\phi_{j,n}(x)$ where

$$f(x) = \sum_{n=-\infty}^{\infty} c_{j,n}\phi_{j,n}(x) + \sum_{m=-\infty}^{j}\sum_{n=-\infty}^{\infty} d_{m,n}\psi_{m,n}(x) \qquad (5.1)$$

and $\psi_{j,n}(x)$ and $\phi_{j,n}$ span orthogonally complimentary spaces [3]. As shown in Chapter 5, the detail or information lost going to the lower approximation of the original function, f(x), by taking the inner product of f(x) with $\phi_{j,n}$, is given by the projection (or inner product) of the original function in the $\psi_{j,n}(x)$ subspace. Thus, the discrete representation of a discrete function $a_{j-1,1}(f)$ at resolution j-1 is

$$c_{j-1,1}(f) = \sum_{n} \left(g_{2n-1} d_{m,n}(f) + h_{2n-1} c_{j,n}(f)\right) \qquad (5.2)$$

where g_{2n-1} and h_{2n-1} are the discrete high and low pass filters, respectively, and $c_{j,n}$ and $d_{m,n}$ are the lower j resolution discrete signal and the details lost by going to the lower resolution signal, respectively [4].

It follows that one can represent an arbitrary signal by its lower resolution and the detail lost going to the lower resolution with proper choices of the basis functions. Furthermore, one can extend this concept to a multiresolution approach where the original signal is represented by consecutively decomposing it to lower resolutions, while at the same time storing all the detail information between each consecutive resolution.

This multiresolution approach can be extended to a 2-D image and is quite suitable for progressive transmission of images (see Figure 5.2). In the 2-D multiresolution WT, one must introduce a scaling function $\phi(x,y)$ and a wavelet function $\psi(x,y)$ that are separable. That is

$$\phi(x,y) = \phi(x)\phi(y) \tag{5.3}$$

and

$$\psi(x,y) = \psi(x)\,\psi(y). \tag{5.4}$$

In addition, the combinations of these functions are defined as

$$\phi(x,y)_l = \phi(x)\,\phi(y) \tag{5.5}$$

$$\psi(x,y)_h = \phi(x)\,\psi(y) \tag{5.6}$$

$$\psi(x,y)_v = \psi(x)\,\phi(y) \tag{5.7}$$

$$\psi(x,y)_d = \psi(x)\,\psi(y)\,. \tag{5.8}$$

Using a similar extension of the properties given in Chapter 4, one can now define the filters h and g for the basis functions. The filters are h_r, h_c, g_r, and g_c for

$\phi(x)$, $\phi(y)$, $\psi(x)$, and $\psi(y)$, respectively. The subscripts r and c stand for row and column and indicate the direction in which the filters are applied. [2, 4] So, the low pass and detail images are given by

$$c_1 = h_r h_c c_0, \tag{5.9}$$

$$d_h = d_{1,1} = h_r g_c c_0, \tag{5.10}$$

$$d_v = d_{1,2} = g_r h_c c_0, \tag{5.11}$$

and

$$d_d = d_{1,3} = g_r g_c c_0. \tag{5.12}$$

Computationally, the 2-D WT of an image is taken in two parts. First, the 1-D WT is taken along the image pixel rows by multiplying each row by the appropriate low and high pass filters h and g. The low pass and detail groups are then down sampled by two. The second step in the 2-D WT is accomplished by taking the 1-D WT along the columns of each of the reordered low and high pass filtered groups. This is again accomplished by multiplying both the low pass and high pass filtered groups by the same low pass and high pass filters discussed above. The columns are again down sampled by two. The result of this operation is a decomposition which has

a low pass L image in quadrant one, a vertical error image D_v in quadrant two, a horizontal error image D_h in quadrant three, and a diagonal error image D_d in quadrant four. Thus, the first wavelet decomposition of the image is complete. Further wavelet transformations of the resulting low pass image will result in a multiresolution WT decomposition. See Figures 5.1, 5.2, and 5.3 for a graphical representation of the multiresolution WT results and implementation.

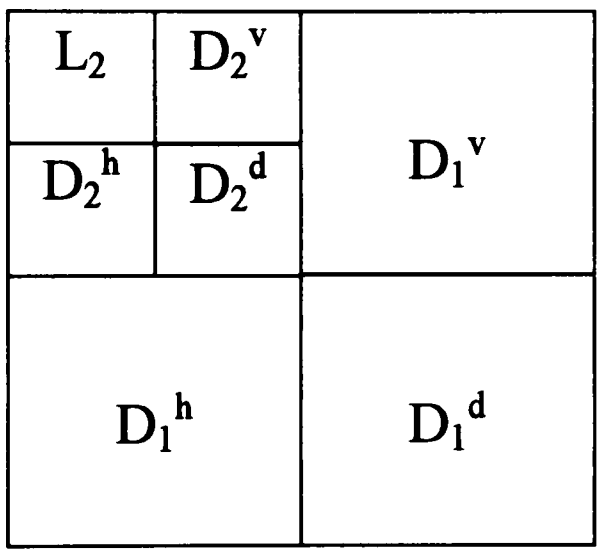


Figure 5.1. Multiresolution or pyramidal decomposition

Figure 5.2. Two level wavelet decomposition of 512X512 8bpp Lena image using Daubechies' 4 wavelet coefficients

5.2 Coding algorithm

The multiresolution wavelet transform packs the image information in the top part of a pyramidal decomposition. We fashioned our multiresolution approach to Mallat's [3] approach where the wavelet transform of the original and then the subsequent low pass versions of the decomposition are taken in order to form the multiresolution pyramid (see Figures 5.1, 5.2, and 5.3).

After carrying out the multiresolution wavelet decomposition, we used scalar quantizers in order to quantize the coefficients of the decomposed image and map them into different bins. Since the detail images contain the high frequency content of the image (i.e., harder to see visually), their coefficients were coarsely quantized by using only a few bins for mapping the pixel values after the WT. However, we applied a more accurate scalar quantizer to the low pass image which simply truncated and mapped, if necessary, the pixel values of the low pass image to values between 0 and 255. Since the low pass image contains the information which is more critical to visual interpretation of the image, this approach maintained good image quality in the reconstructions. Finally, a combination of Run Length Coding (RLC) and Huffman Coding was applied to the coefficients to get the final compression. The number of decompositions (levels in the pyramid) and coarseness of scalar quantizers (i.e., binsize) were varied in order to find the maximum compression ratio that gave reasonable PSNR values (see Figure 5.3).

After comparing the compression capabilities of the Daubechies 4, Daubechies 12, and Adelson 15 wavelet coefficients, we found that Adelson 15 coefficients worked best for compressions requiring more that three levels of decompositions. This is generally the case when higher compression ratios are required since Adelson 15 tap wavelet filter coefficients do not add artifacts as is the case with Daubechies coefficients [2, 14]. Thus, we chose to use a program called Epic which utilizes Adelson 15 wavelet coefficients and optimizes the wavelet compression to perform the Adelson 15 wavelet compressions. The Epic program also uses scalar quantization, run length coding, and Huffman coding to achieve its final image compression.

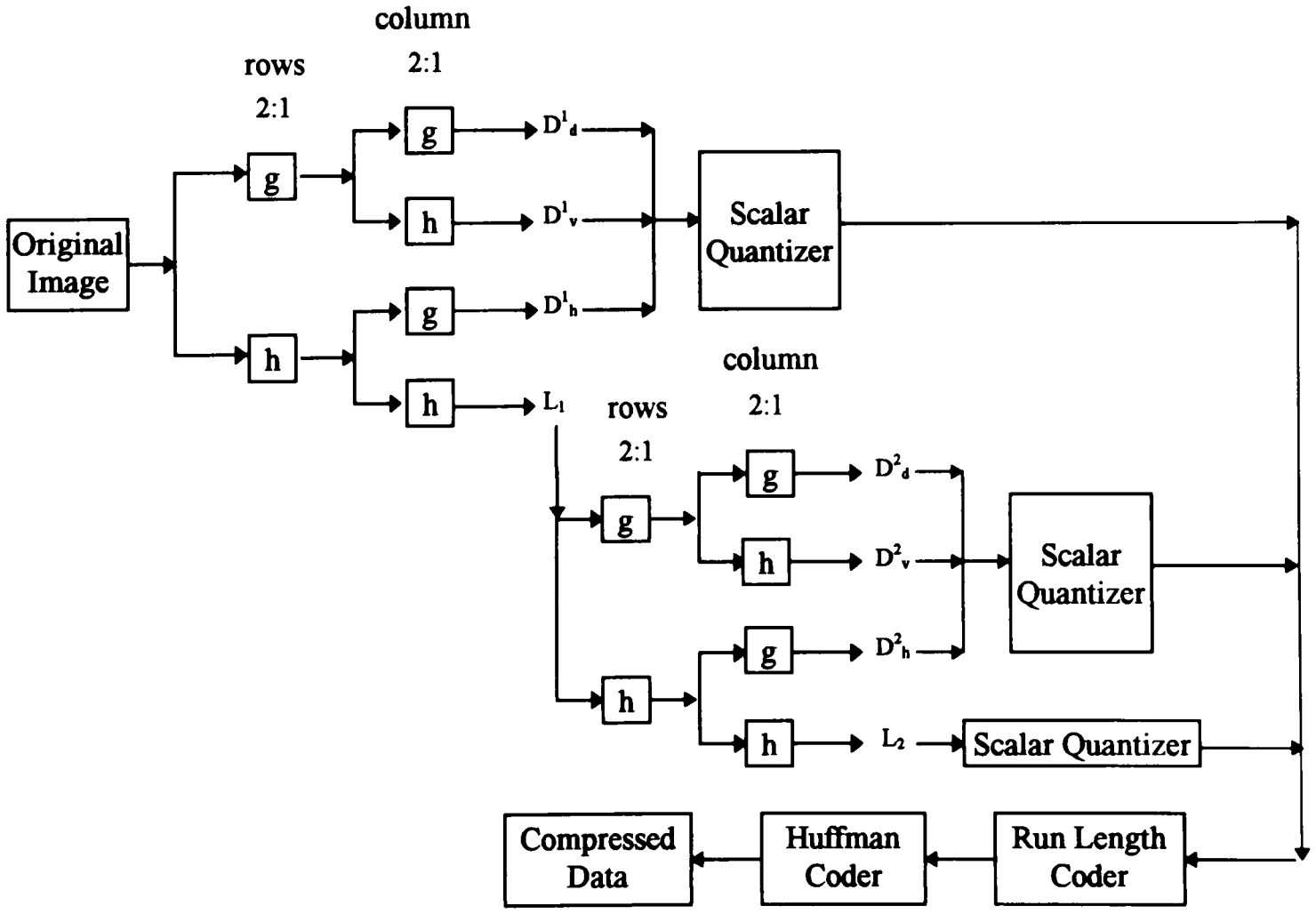


Figure 5.3. Multiresolution WT pyramid coding

CHAPTER 6

PERFORMANCE EVALUATION

Each technique's performance was measured by calculating the mean square errors (MSE), normalized MSE (NMSE), the peak signal to noise ratio (PSNR) in dB, and root mean square signal to noise ratio (SNRrms). The equations which describe these performance evaluation functions are:

$$\mathrm{MSE} = \frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(\hat{x}_{i,j} - x_{i,j})^2}{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(x_{i,j})^2}, \tag{6.1}$$

$$\mathrm{NMSE} = \frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(\hat{x}_{i,j} - x_{i,j})^2}{255^2 * N^2} \tag{6.2}$$

$$\mathrm{PSNR} = 10\log_{10}\frac{255^2}{\frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(x_{i,j} - \hat{x}_{i,j})^2} = 10\log_{10}\frac{1}{\mathrm{NMSE}}, \tag{6.3}$$

and

$$\mathrm{SNR(rms)} = \sqrt{\frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(\hat{x}_{i,j})^2}{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(\hat{x}_{i,j} - x_{i,j})^2}} \tag{6.4}$$

where $\hat{x}_{i,j}$ and $x_{i,j}$ represent the pixel value at location (i,j) of the reconstructed and original image, respectively, and N^2 is the total number of pixels in the image.

It was found that the performance evaluation criteria which best matches the human visual quality of the image was the PSNR. For this reason, a heavy emphasis will be placed on the PSNR over the other techniques as a criterion for performance evaluation.

CHAPTER 7

RESULTS AND DISCUSSION

7.1 Wavelet basis function comparison

In order to compress the images used in this thesis efficiently, the appropriate wavelet coefficients needed to be chosen. To avoid randomly choosing one set of coefficients, the performance evaluation measurements given in Chapter 6 were applied to a small sample of images which were compressed and reconstructed using Daubechies 4, Daubechies 12, and to Adelson 15 tap wavelet coefficients. Tables 7.1 and 7.2 show the results of the compressions performed on the standard Lena image and the cervical spine image.

Table 7.1. CR and PSNR for different wavelet decompositions and wavelet coefficients of 512X512 8 bpp Lena image (binsize=10)

	Daub4		Daub12		Adelson15	
Level	CR	PSNR	CR	PSNR	CR	PSNR
1	1.92	39.05	1.86	38.75	1.80	37.48
2	3.25	39.57	3.01	39.54	4.31	38.52
3	3.02	38.56	3.07	38.54	6.15	38.72

Table 7.2. CR and PSNR for different wavelet decompositions and wavelet coefficients of 512X512 8 bpp cervical spine image (binsize=10)

	Daub4		Daub12		Adelson15	
Level	CR	PSNR	CR	PSNR	CR	PSNR
1	5.54	46.96	4.61	39.69	1.99	40.96
2	9.10	41.71	10.62	45.41	7.27	44.89
3	6.35	34.93	14.39	44.86	19.10	45.91

Figures 7.1, 7.2, 7.3 and 7.4 show a graphical comparison of the CR and PSNR of both the Lena and cervical spine compression.

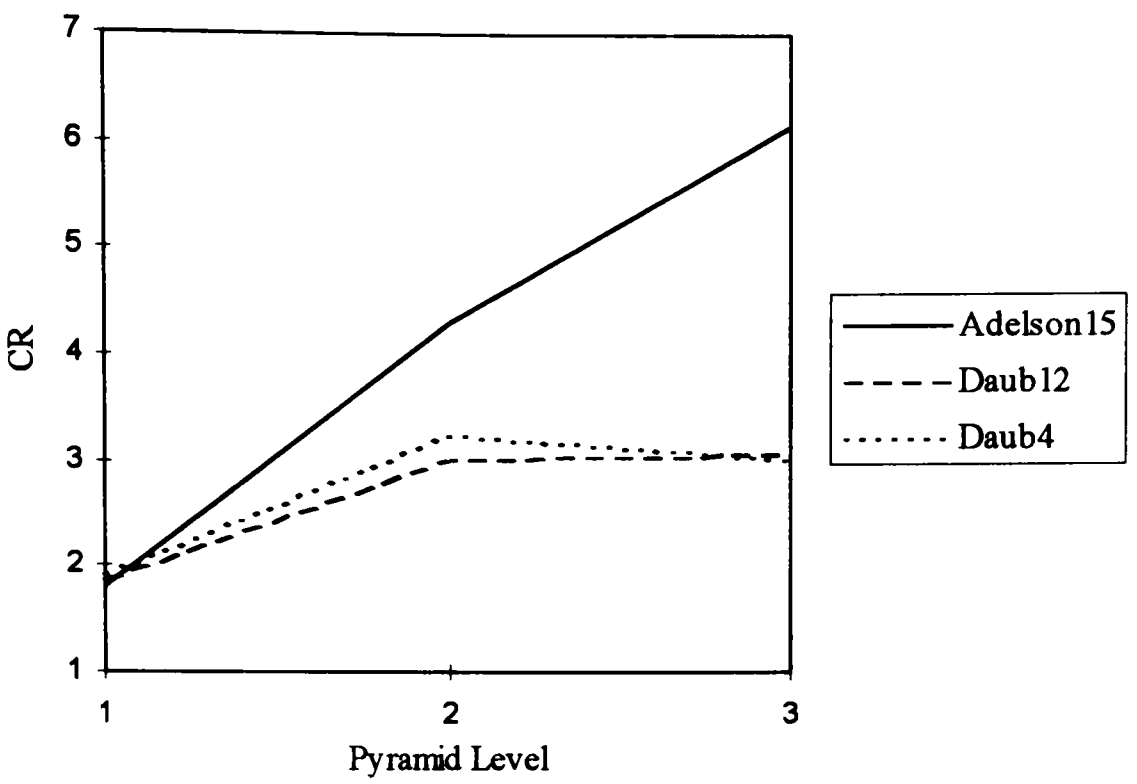


Figure 7.1. CR versus level of decomposition for 512X512 8 bpp Lena image using a binsize=10

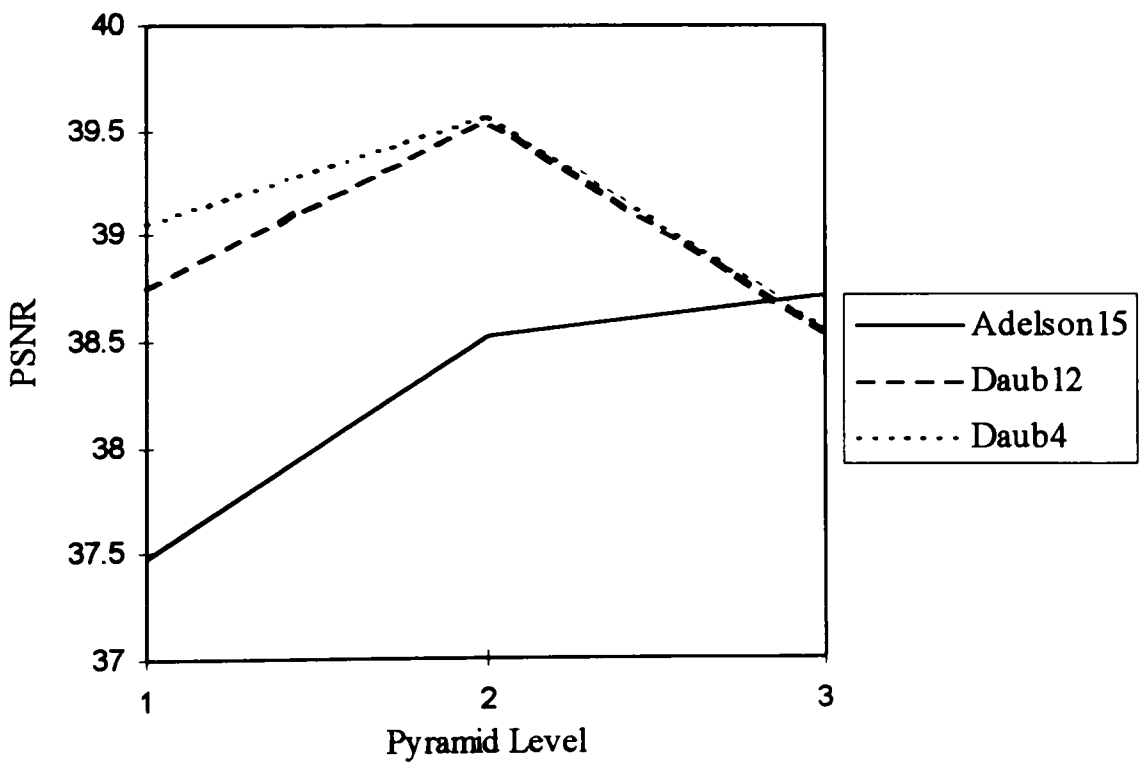


Figure 7.2. CR versus level of decomposition for 512X512 8 bpp Lena image using a binsize=10

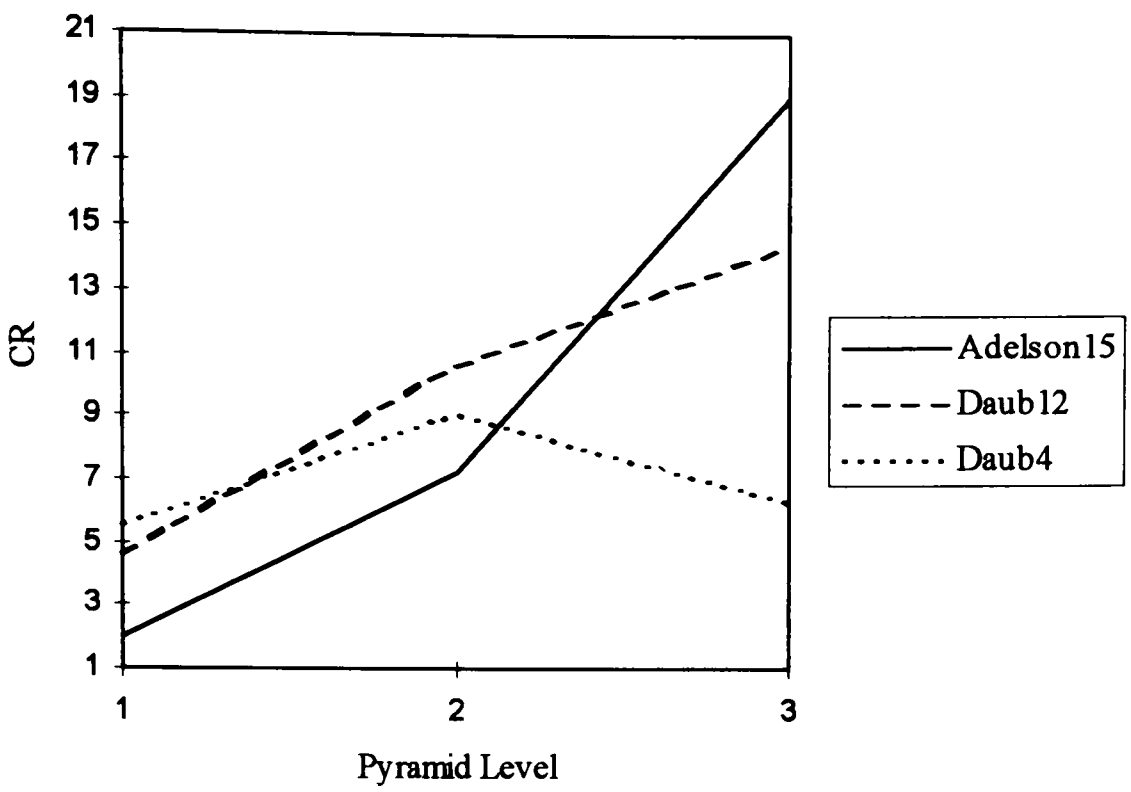


Figure 7.3. CR versus level of decomposition for 512X512 8 bpp cervical spine image using a binsize=10

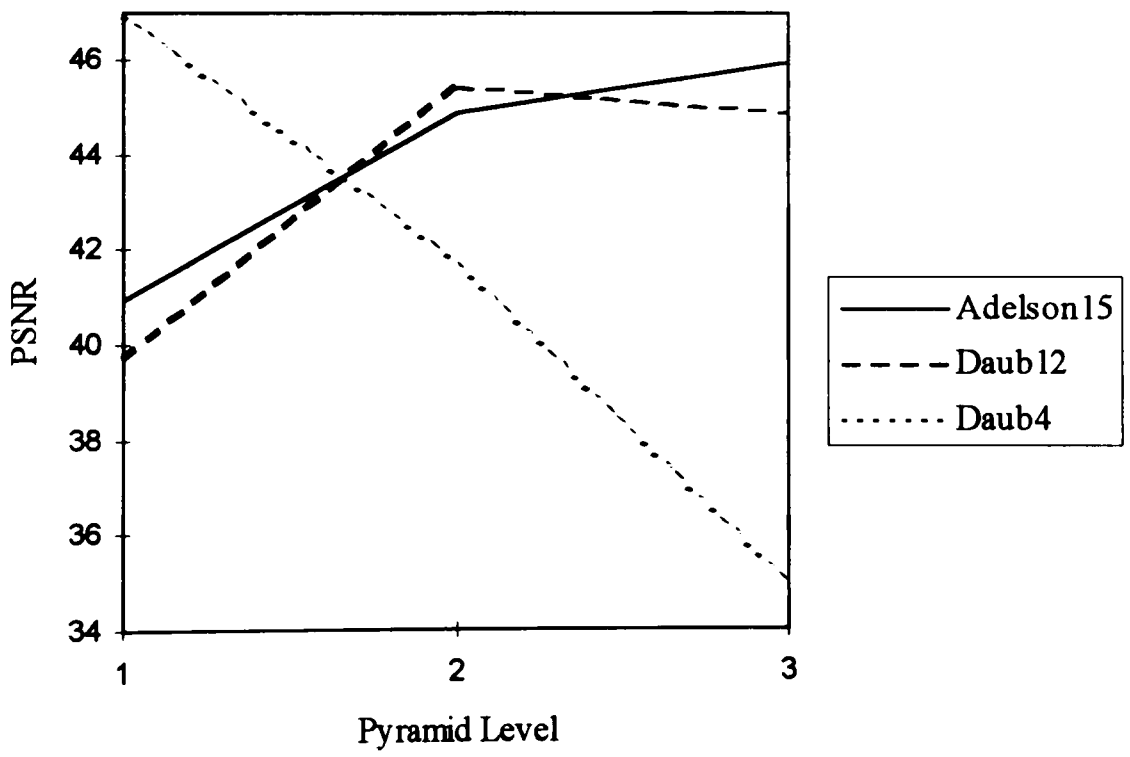


Figure 7.4. PSNR versus level of decomposition for 512X512 8 bpp cervical spine image using a binsize=10

As can be seen from these Figures 7.1, 7.2, 7.3, and 7.4, the Daubechies coefficients perform better than Adelson's coefficients when one level decomposition of the Lena or one and two level decompositions of the cervical spine were used for compression. However, for the 3 level decomposition of the cervical spine as well as for the two and three level decomposition for the Lena, Adelson's coefficients produced a better CR as well as superior PSNR. Since the purpose of the wavelet algorithm is to achieve a high CR with a high PSNR, Adelson's 15 wavelet coefficients are the optimal coefficients out of the three compared. Thus, for the rest of this research, they will be used in the wavelet compression algorithm.

Before continuing to the compression results, however, one more feature of the coefficients should be pointed out. All of the coefficients were able to give both higher CR and PSNR for the radiographic image compressions. This is attributed to the higher correlation of the pixels in the cervical spine image. Although no quantitative values were tabulated for the particular images, visual inspection of both images reveals that the radiographic images contain far less detail than the Lena images. Thus, the radiographic image contains more redundancy and should yield higher compression ratios.

7.2 Standard Image Compression

After choosing Adelson's wavelet coefficients to form out basis, the WT-based compression technique was applied to the standard 512x512 8 bpp Lena image.

Although the original intent of this thesis was to apply the WT compression algorithm to radiographic images which contain a great amount of correlation, the Lena image was chosen for the initial compressions since it is one of the standards used in image compression. In addition, since the Lena image will generally be less correlated than the highly redundant radiographic images, the compression results of the Lena will give a good baseline comparison.

Table 7.3. WT compressed Lena image (512X512 8 bpp)

Levels	Binsize	CR	PSNR(dB)	MSE(x10^-4)	SNR(rms)
5	100	50.93	29.30	41.40	15.53
5	80	42.08	30.22	33.60	17.25
5	60	32.96	31.33	26.00	19.60
5	40	22.84	32.92	18.00	23.58
5	35	20.00	33.47	15.88	25.10
5	30	17.48	34.09	13.70	26.96
5	25	14.91	34.84	11.60	29.36
5	20	12.24	35.74	9.40	32.61
5	11	7.33	38.33	5.18	43.91
5	6	4.29	41.44	2.54	62.78
5	1	1.85	53.59	0.15	254.54

Table 7.3 shows the CR, PSNR, MSE and SNR(rms) obtained from the WT compression of the Lena image using 5 multiresolution levels and different binsizes. The WT compression scheme gave a wide range of attainable compression ratios. It should be pointed out, however, that although CRs of 50.9 and higher could be achieved, the visual quality of the reconstructed image was progressively poorer.

In order to get a better perception of the WT performance, it was decided to compare the WT compression scheme to the current JPEG standard. So, the 512x512

8 bpp Lena image was compressed using JPEG. The results of these compressions are given in Table 7.4.

Table 7.4. JPEG compressed Lena image (512X512 8 bpp)

Quality Factor	CR	PSNR(dB)	MSE(x10^-4)	SNR(rms)
10	32.84	30.46	32	17.68
20	22.13	33.07	18	23.68
30	17.37	34.42	13	27.87
40	14.60	35.29	11	30.82
50	12.64	35.97	9	33.33
60	10.97	36.61	7.8	35.89
70	9.04	37.49	6.3	39.71
80	7.01	38.71	4.8	45.72
90	4.49	41.01	2.8	59.55
100	1.63	58.51	0.05	446.75

By comparing Tables 7.3 and 7.4, one can see that the WT and the JPEG compressions performed comparably for the Lena image. Both gave approximately equal PSNR for similar CR.

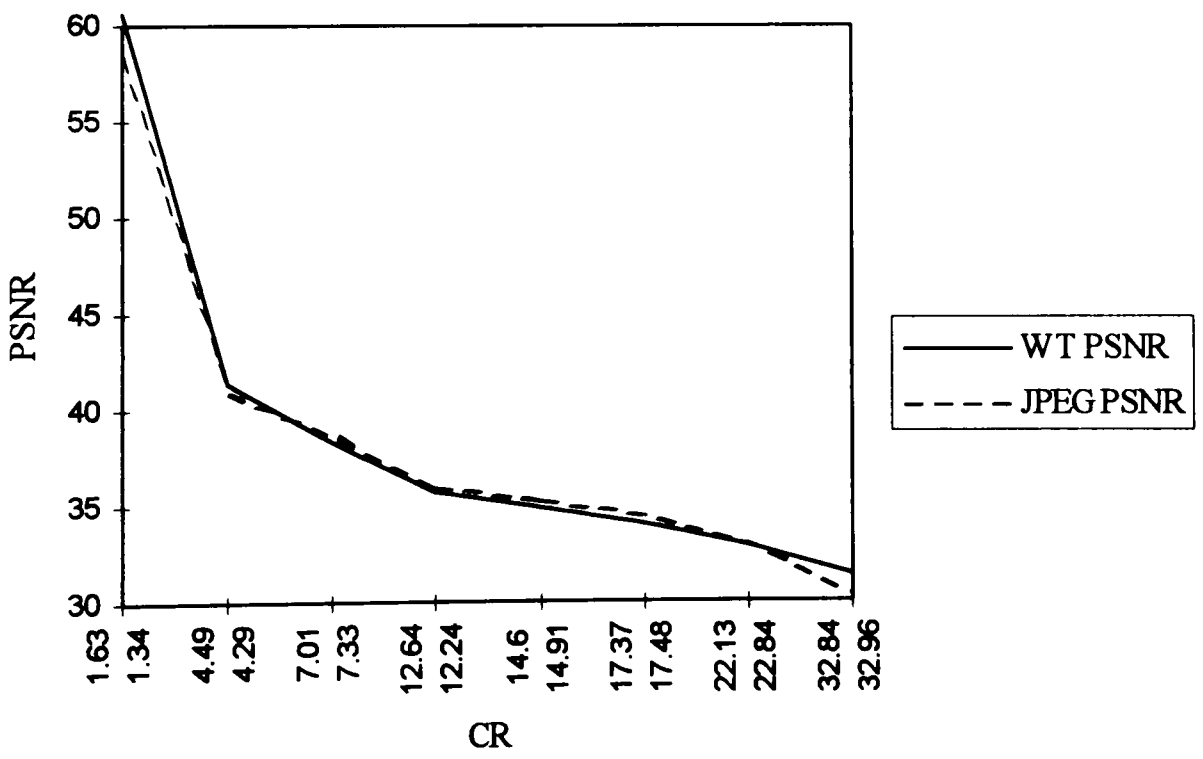


Figure 7.5. Comparison of the WT and JPEG PSNR at similar CRs for the 512X512 8 bpp Lena image

Figure 7.5 contains a plot of the WT and JPEG coding's PSNR versus their corresponding CRs for the Lena image. The left CR number of each subset on the x-axis is the JPEG CR while the right CR of each subset is for the WT coding. As can be seen from Figure 7.5, the WT and JPEG compression algorithm's compressions for the Lena image closely followed each other in the PSNR sense.

Figure 7.6, 7.7, and 7.8 show some of the wavelet and JPEG reconstructed images that were compressed to similar CRs. Note that the images are very similar visually.

7.3 Medical Image Compression

A cervical spine image provided by the National Library of Medicine was used to test the WT compression capabilities on images which are highly correlated. After the WT compressions were done, the cervical spine image was compressed with the JPEG algorithm. As stated in Section 7.2, the purpose of the JPEG compression was to compare the WT performance to the current JPEG standard.

Both techniques gave good compression results up to a compression ratio of around 35. However, two notable differences were seen. First, the wavelet transform was able to achieve higher compression ratios than JPEG at similar PSNRs (see Tables

(a)

(b)

Figure 7.6. 512X512 8 bpp Lena image (a) original and (b) WT CR 42.08

(b)

(b)

Figure 7.7. 512X512 8 bpp Lena (a) JPEG CR 32.84 and (b) WT CR 32.96

(a)

(b)

Figure 7.8. 512X512 8 bpp Lena (a) JPEG CR 7.01 and (b) WT CR 7.33

7.5 and 7.6 and Figures 7.9, 7.10, 7.11, and 7.12). For example, the wavelet decomposed image was compressed to 75.6:1 while the JPEG image was compressed to 34.4:1 at PSNRs of 43.24 dB and 44.06 dB, respectively. It was also apparent that the JPEG images that were compressed at a higher CR contained a high amount of blocking noise. Although the WT compressed images also had some noise, this noise was not visually apparent until much higher compression ratios (see Figures 7.10, 7.11, and 7.12). On the other hand, at lower compression ratios of around 5:1, JPEG slightly outperformed the WT. However, it should be noted here that the better PSNRs at the lower compression ratios were only slightly greater than the WT's PSNRs, and visually, the images were very similar.

Table 7.5. WT compressed cervical x-ray image (512X512 8 bpp)

Levels	Binsize	CR	PSNR(dB)	MSE(xe-4)	SNR(rms)
5	100	197.25	36.29	9.30	32.84
5	80	170.92	37.89	6.44	39.45
5	60	144.04	39.06	4.90	45.15
5	40	107.92	41.07	3.09	56.85
5	30	86.43	42.42	2.27	66.42
5	25	75.61	43.24	1.88	72.96
5	18	57.77	44.17	1.52	81.16
5	12	38.94	45.55	1.10	95.22
5	11	35.55	45.89	1.02	99.04
5	10	32.15	46.19	0.95	102.42
5	9	29.03	46.60	0.87	107.39
5	5	16.59	49.05	0.49	142.44
5	1	5.36	57.53	0.07	377.94

Table 7.6. JPEG compressed cervical spine image (512X512 8 bpp)

Quality Factor	CR	PSNR(dB)	MSE(xe-4)	SNR(rms)
10	57.69	36.11	9.69	32.10
20	45.90	40.55	3.49	53.60
30	38.50	42.49	2.23	66.97
40	34.35	44.06	1.56	80.27
50	31.34	45.72	1.06	97.08
60	28.49	46.60	0.87	107.43
70	24.19	47.07	0.78	113.48
80	19.33	49.66	0.43	152.84
90	12.66	52.58	0.22	213.97
100	5.26	60.37	0.04	524.62

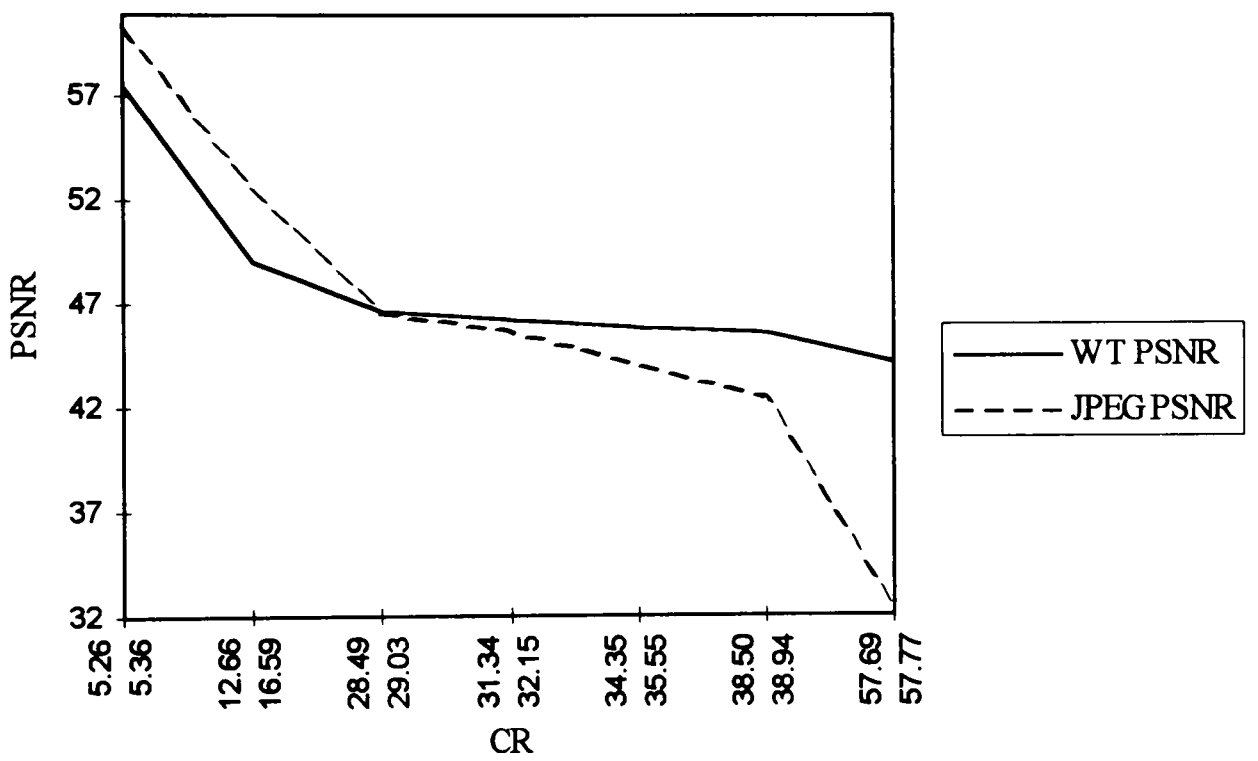
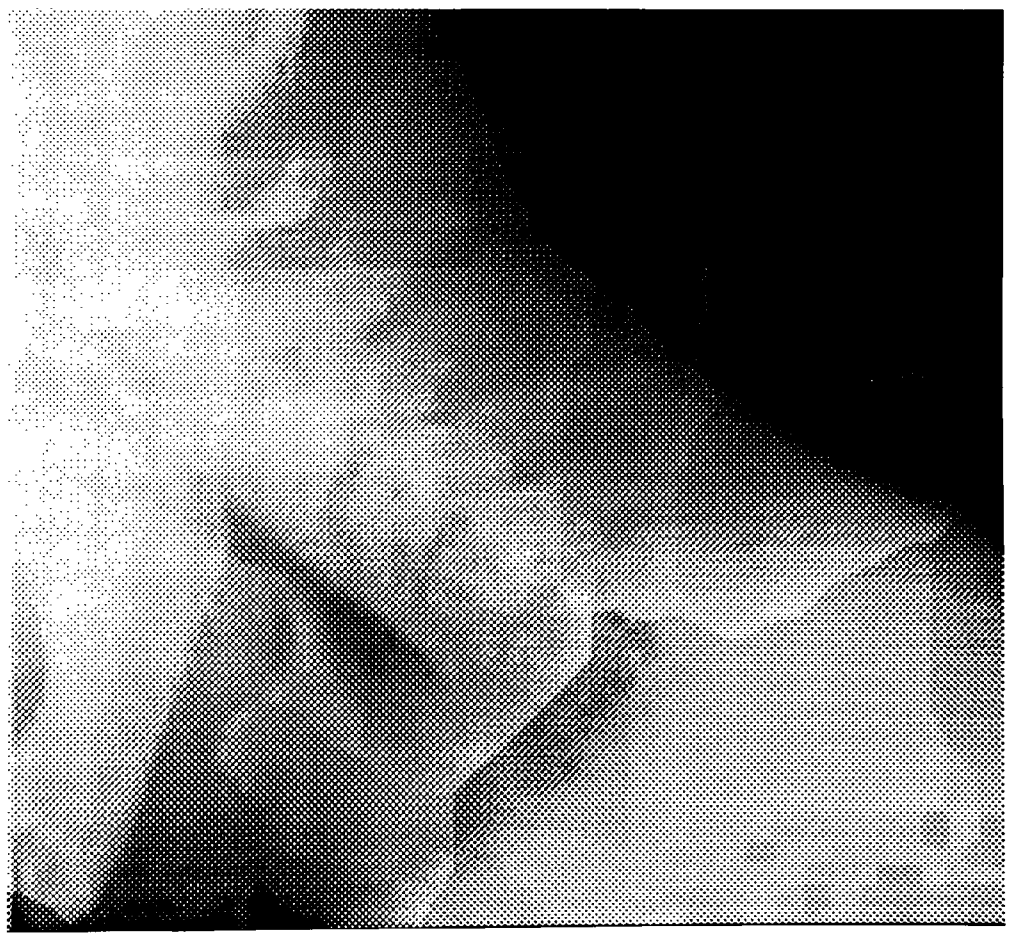


Figure 7.9. Comparison the WT and JPEG PSNR at similar CRs for 512X512 8 bpp cervical spine image

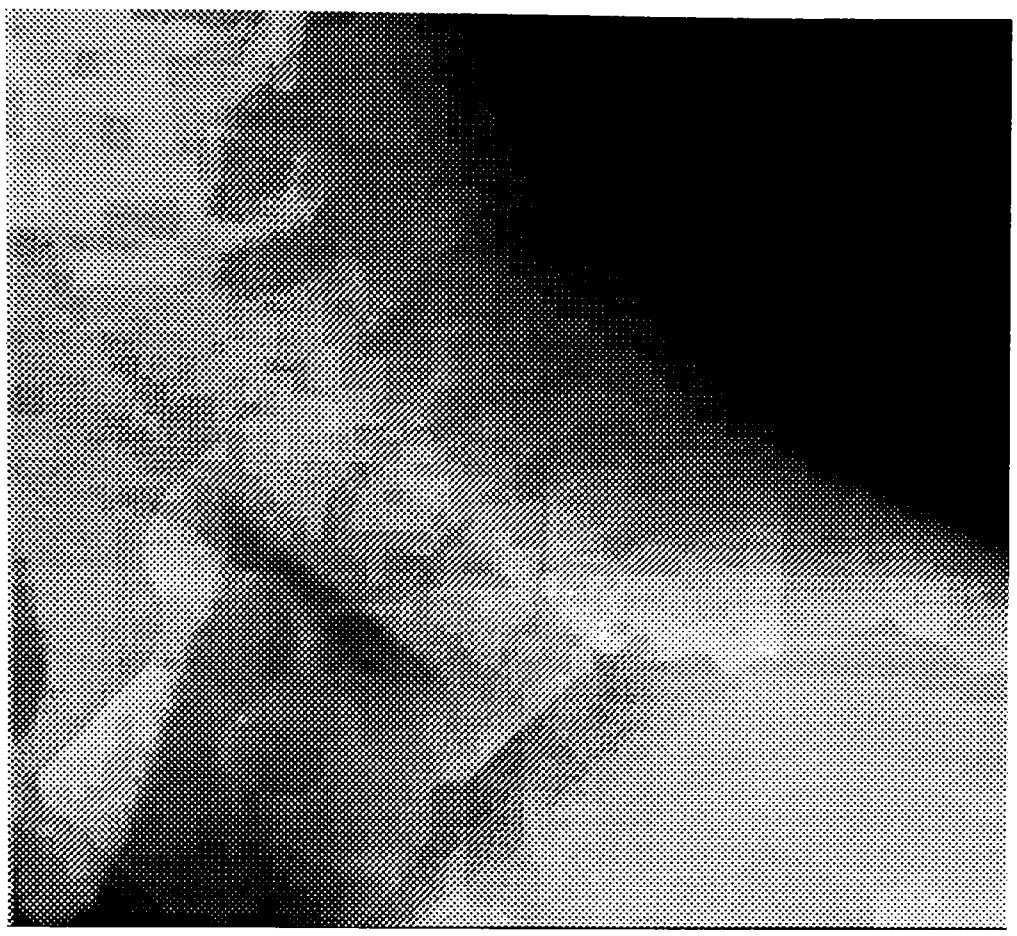
Figure 7.9 contains a plot of the WT and JPEG coding's PSNR versus their corresponding CRs for the cervical spine image. The left CR number of each subset is the JPEG CR while the right CR of each subset is for the WT coding. This graph clearly shows the higher PSNR attained by the WT over JPEG at similar CR above about 30. It can also be seen that in the CR range of around 29 to 35, both JPEG and WT have similar performance while JPEG has slightly better PSNR at lower CR. However, it should be pointed out that although JPEG has better PSNRs at the lower CRs, the comparisons are made with slightly different CRs. So, it can be concluded that JPEG will only have a very small advantage over WT coding at very low CRs in the PSNR sense.

(a)

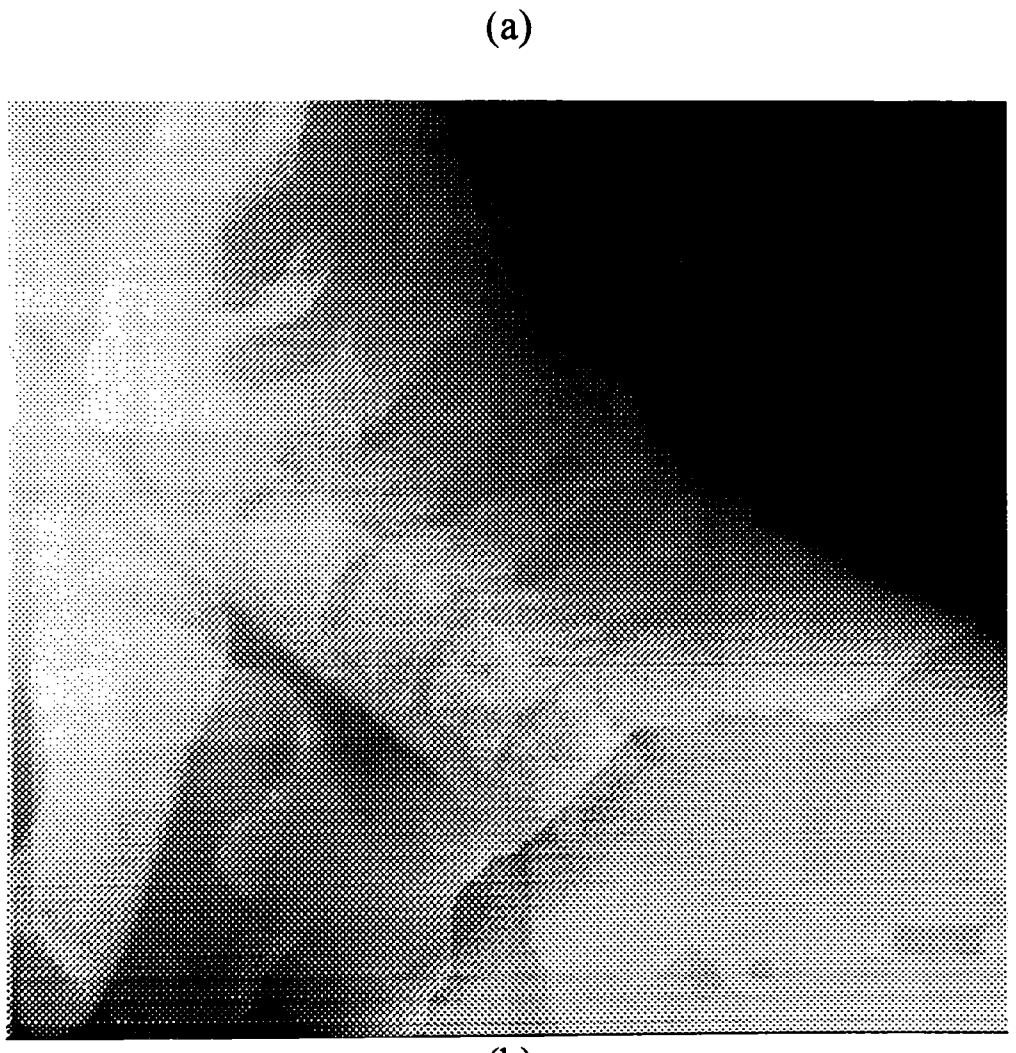


(b)

Figure 7.10. 512X512 8 bpp cervical image (a) original and (b) WT CR 144

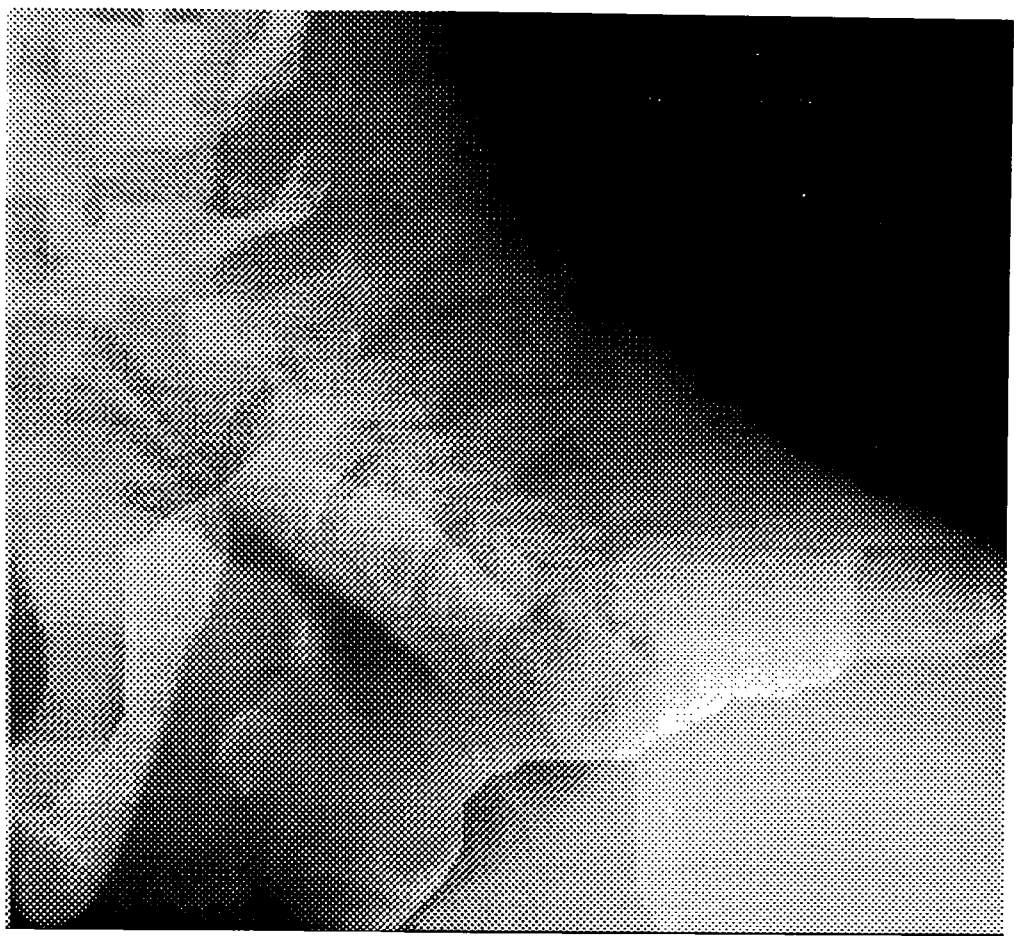


(a)

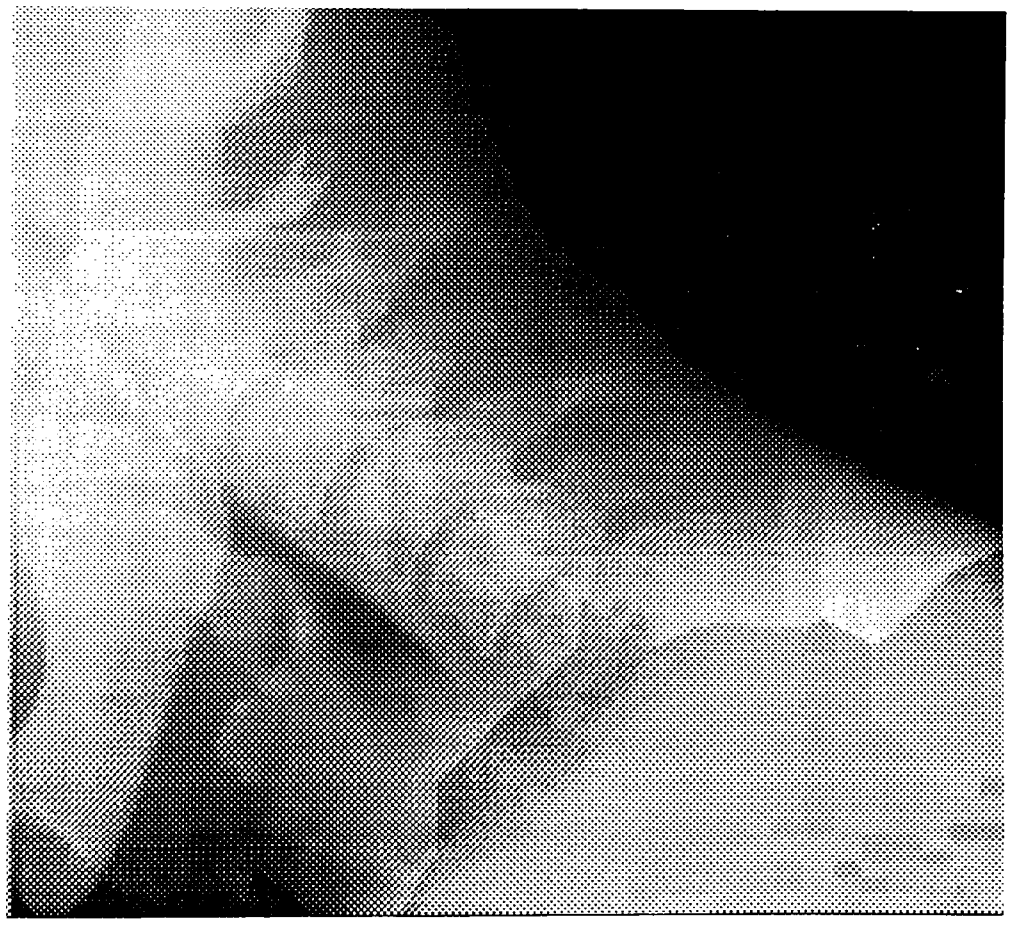


(b)

Figure 7.11. 512X512 8 bpp cervical (a) JPEG CR 58 and (b) WT CR 58



(a)



(b)

Figure 7.12. 512X512 8 bpp cervical image (a) JPEG CR 19 and (b) WT CR 17

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

In this thesis, we presented a comparison of the compression capabilities of WT coding and JPEG. This work showed that, to achieve higher compression ratios, WT coding requires a multi-level wavelet decomposition. Thus, Adelson 15 wavelet coefficients are best suited for this type of compression and were chosen over Daubechies 4 and 12 wavelet coefficients. In addition, the thesis shows that the WT coding algorithm and the JPEG standard performed comparably in the PSNR sense for images such as the Lena. These images have a high information content and both algorithms are able to decorrelate the information in a similar manner. The WT coding algorithm, however, was able to achieve higher compression ratios than JPEG. This is mainly attributed to the lack of an upper bound limitation in the WT code.

JPEG had slightly better compression results than WT coding at very low CRs in the PSNR sense when radiographic images were used. WT coding, however, clearly outperformed JPEG in high image compression ratios. As anticipated, JPEG introduced a highly noticeable amount of blocking noise at the higher compression ratios. These effects were of course not present with the WT coding. However, some visible degradation occurred at compression ratios above 35.

The range of compression ratios yielding reconstructed images with good visual quality when WT coding is used varies with specific image characteristics.

However, preliminary results indicate that for most radiographic images, the usable compression ratios are much higher for the WT than for the JPEG. Thus, WT coding should have an advantage in radiographic image compression and in telemedicine.

Further work should be done in this area. In order to achieve an optimal WT compression algorithm, a thorough comparison of all wavelet coefficients should be performed. Included in these coefficient tests, a wide range of images such as radiographic, standard, highly correlated and uncorrelated, gray level and color images need to be used to give a complete analysis. This study should reveal the best wavelet coefficients to use for each type of image and desired CR. The coding algorithm should also be varied to include a better scalar quantization scheme where the scalar quantization is varied from one pyramid level to the next. This variation should achieve maximal compression ratios with the best PSNR. Other quantizations schemes such as vector quantization should also be investigated to see the effect on the WT compressions. Finally, the wavelet reconstructed radiographic images need to be visually evaluated by ROC tests and by practicing physicians. The physician's opinion of the quality of the images is critical since ultimately he is the individual who has to base his medical opinion on the given information.

REFERENCES

[1] Stephen Wong, Loren Zaremba, David Gooden, and H.K. Huang, "Radiologic image compression - a review," *Proc. of the IEEE*, vol 83, no. 2, pp. 194-219, Feb. 1995.

[2] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol 41, pp. 909-996, Nov 1988.

[3] Stephane G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol 11, no. 7, pp 674-693, July 1989.

[4] Marc Antonini, Michel Barlaud, Pierre Mathieu, and Ingrid Daubechies, "Image coding using wavelet transform," *IEEE Trans. on Image Proc.*, vol. 1, no. 2, pp. 205-220, April 1992.

[5] H. J. Barnard, *Image and Video Coding Using Wavelet Transforms*, Ph.D. Dissertation, Technische Universiteit Delft, May 1994.

[6] Ali N. Akansu, "Wavelets and Filter Banks," *IEEE Circuit and Devices*, pp 14-18, November 1994.

[7] A.V. Rao, M. Thomas, S. Mitra, and M.E. Parten, "Compression of high-resolution images with visually transparent techniques," SPIE, Proc. on Applications of Digital Image Processing, San Diego, July 1994.

[8] Rafael C. Gonzales and Richard E. Woods, *Digital Image Processing*, New York: Addison-Wesley Publishing Company, 1992.

[9] Anil K. Jain, *Fundamentals of Digital Image Processing*, Englewood Cliff, NJ: Prentice Hall, 1989.

[10] M. A. Sid-Ahmed, *Image Processing - Theory, Algorithms, & Architectures*, New York: McGraw-Hill, Inc., 1995.

[11] R. J. Clarke, *Transform Coding of Images*, London: Academic Press, 1985.

[12] N.M. Nasrabadi, R.A. King, "Image coding using vector quantization: a review," *IEEE Trans. on Communications*, vol COM-36, no. 8, pp. 955-971, Aug. 1988.

[13] Y. Linde, A. Buzo, R. Gray, "An algorithm for vector quantizer design," *IEEE Trans. on Communications*, vol. COM-28, no. 1, pp. 84-95, Jan. 1980.

[14] Majid Rabbani and Paul W. Jones, *Digital Image Compression Techniques*, Washington DC: SPIE Optical Engineering Press, 1991.

[15] Pennebaker, "JPEG Technical Specification, Revision 8," Working Document No. JTC1/SC2/WG10/JPEG-8-R8 (Aug. 1990).

[14] Paul W. Jones, Scott Daly, Roger S. Gaborski, and Majid Rabbani, "Comparative Study of Wavelet and DCT Decompositions with Equivalent Quantization and Encoding Strategies for Medical Images," *SPIE, Proceedings on Medical Imaging 1995*, Vol 2431, pp. 571-582.

PERMISSION TO COPY

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Texas Tech University or Texas Tech University Health Sciences Center, I agree that the Library and my major department shall make it freely available for research purposes. Permission to copy this thesis for scholarly purposes may be granted by the Director of the Library or my major professor. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my further written permission and that any user may be liable for copyright infringement.

Agree (Permission is granted.)

Richard Muyshondt — 11-1-95

Student's Signature — Date

Disagree (Permission is not granted.)

_____ — _____

Student's Signature — Date