IMAGE MIRRORING, ROTATION AND INTERPOLATION IN THE WAVELET DOMAIN

by

THEJU ISABELLE JACOB

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

August 2008

To Bhageerathy Sundaram.

Till we meet again.

ACKNOWLEDGEMENTS

I would like to thank my supervising professor Dr.K.R.Rao for his constant encouragement and support. He has always been very prompt in answering my queries, diligent in correcting my mistakes and in giving further suggestions. I thank Dr.Alan Davis and Dr.Kambiz Alavi for taking interest in my work and for taking time off to be in my defense committee. I also thank Dr.Soontorn Oraintara under whom I took the Wavelets and Filter Banks course beside others; I thank him for the wonderful teacher he has been.

To Pappa and Amma, for being there when it mattered, and for all of the support they provided throughout my degree program, thank you. To my sister, for bringing cheer in her own ways. To my two friends, you know who you are, thank you for the support from across the miles.

July 14, 2008

ABSTRACT

IMAGE MIRRORING, ROTATION AND INTERPOLATION IN THE WAVELET DOMAIN

Theju Isabelle Jacob, M.S.

The University of Texas at Arlington, 2008

Supervising Professor: K. R. RAO

JPEG2000, the international standard for still image compression, uses the wavelet transform. The filters used in JPEG2000 for transformation are the 9/7 Daubechies filters and the 5/3 Le Gall filters. At present, to achieve image mirroring or rotation for images, one has to convert them back to spatial domain and proceed. This thesis presents methods by which it can be implemented in the wavelet domain, without extra computational complexity. To do so, one manipulates the transform domain coefficients so that the condition of perfect reconstruction still holds true.

The thesis also discusses a method to achieve image interpolation, which would result in a higher resolution image from a lower resolution one. One manipulates the wavelet domain coefficients of the image for the same. For comparison purposes, one takes the low frequency component obtained after the wavelet transform of the image, which acts as the low resolution image, try to predict the rest of the components, and then reconstruct the original image. The reconstructed image is then compared with the original image.

TABLE OF CONTENTS

Appendix

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

Z ..... Set of Integers

$\mathbf{R}$ ..... Set of Real Numbers

$L^2(\mathbf{R})$ ..... Space of square integrable functions

$\bigoplus$ ..... Vector Space Addition

$<>$ ..... Inner Product

CHAPTER 1

INTRODUCTION

Image mirroring and rotation in the transform domain allows one to obtain mirrored and rotated images from the transform domain components of the original image. It allows one to do away with the necessity of having to convert the transform domain components of the original image back to time domain first. Initially, it is shown how this can be accomplished for the wavelet transform which is in use in the JPEG2000 image codec. Later on, a new method to obtain a higher resolution image from a lower resolution one, again manipulating the wavelet domain coefficients, is discussed. This chapter discusses the wavelet transform and its relation to filterbanks. Chapter 2 discusses the JPEG2000 image codec. Chapter 3 discusses image mirroring and rotation, while chapter 4 discusses image interpolation. Conclusion is given in chapter 5.

1.1 Wavelet Transform

Wavelets emerged as a means of analysing nonstationary signals. The wavelet transform, in its essence, offers a multiscale view of the signal. A single function known as the wavelet exists, whose contracted version can be used for fine temporal analysis, and dilated version that can be used for fine frequency analysis.

The basis functions in the Fourier transform are complex exponentials. In the case of wavelet transform, the basis functions are obtained by translation and dilation of a single prototype wavelet given by[6]:

$$\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k) \tag{1.1}$$

where j, k ϵ Z. This parameterization of the time or space location by k and the frequency or scale by j turns out to be very effective.

Thus, a function f(t) belonging to $L^2(\mathbf{R})$ can be represented by the series:

$$f(t) = \sum_{j,k} a_{j,k} 2^{j/2}\psi(2^j t - k) \tag{1.2}$$

which can be rewritten using (1.1) as:

$$f(t) = \sum_{j,k} a_{j,k}\psi_{j,k}(t) \tag{1.3}$$

The two-dimensional set of coefficients $a_{j,k}$ is called the discrete wavelet transform (DWT) of f(t). A more detailed form indicating how the $a_{j,k}$'s are evaluated can be written using inner products as:

$$f(t) = \sum_{j,k} < \psi_{j,k}(t), f(t) > \psi_{j,k}(t) \tag{1.4}$$

1.2 Concept of Multiresolution

It can be proved that applying the discrete wavelet transform to a signal is equivalent to applying a discrete set of filters. Consider the case of multiresolution as discussed next. First, a scaling function is defined and then the wavelet is defined

in terms of the same. One define a set of scaling functions in terms of the integer translates of the basic scaling function by:

$$\phi_k(t) = \phi(t-k) \qquad k\epsilon Z \qquad \phi\epsilon L^2 \tag{1.5}$$

Let the subspace of $L^2(\mathbf{R})$ spanned by the above functions be V_0, for all integers k from negative infinity to positive infinity. The size of the subspace spanned by the scaling functions can be changed by changing the time scale of the scaling functions. A two-dimensional family of functions is generated from the basic scaling function by scaling and translation by:

$$\phi_{j,k}(t) = 2^{j/2}\phi(2^j t - k) \tag{1.6}$$

Let V_j be the space of functions spanned by the above. Then, the set of functions $f(t)$ which belong to V_j can be represented as a linear combination of $\phi_{j,k}(t)$. For $j > 0$, the span is larger since $\phi_{j,k}(t)$ is narrower and is shifted in smaller steps, and hence represents finer detail. For $j < 0$, $\phi_{j,k}(t)$ is wider and is shifted in larger steps and so, represent coarser information. The space they span, is smaller. It can be thought of it as follows:

$$V_j \subset V_{j-1} \qquad j\epsilon Z \tag{1.7}$$

One could expand the above to represent a nesting of spanned spaces as:

$$...V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2... \subset L^2 \tag{1.8}$$

In short, the space that contains high resolution signals will contain low resolution ones as well. One could hence say that the spaces have to satisfy a natural scaling condition:

$$f(t) \ \epsilon \ V_j \qquad \Leftrightarrow \qquad f(2t) \ \epsilon \ V_{j+1} \tag{1.9}$$

The nesting of spaces as is shown in (1.8) implies that if $\phi(t)$ is in V_0, then it is also in V_1, which is in turn spanned by $\phi(2t)$. Hence, one could express $\phi(t)$ in terms of $\phi(2t)$ as:

$$\phi(t) = \sum_n h_0(n)\sqrt{2}\phi(2t-n), \qquad n\epsilon Z \tag{1.10}$$

Now, consider the functions which span the difference between spaces scanned by the various scales of the scaling function. They are the wavelet functions, denoted by $\psi_{j,k}(t)$. The wavelets would be orthogonal to the scaling functions, and so, W_j, the space spanned by wavelets would be the orthogonal complement of V_j in V_{j+1}. They are denoted as follows:

$$< \phi_{j,k}(t), \psi_{j,l}(t) >= 0 \tag{1.11}$$

Here, j,k,l all belong to Z. Again,

$$V_{j+1} = V_j \bigoplus W_j \tag{1.12}$$

$$...W_{-2} \bigoplus W_{-1} \bigoplus W_0 \bigoplus W_1 \bigoplus W_2... = L^2 \tag{1.13}$$

One could also write it as:

$$W_{-\infty} \bigoplus ..... \bigoplus W_{-1} = V_0 \tag{1.14}$$

As the wavelets reside in the space spanned by the next narrower scaling function $W_j \subset V_{j+1}$, they can be represented by a weighted sum of shifted scaling functions as is shown in:

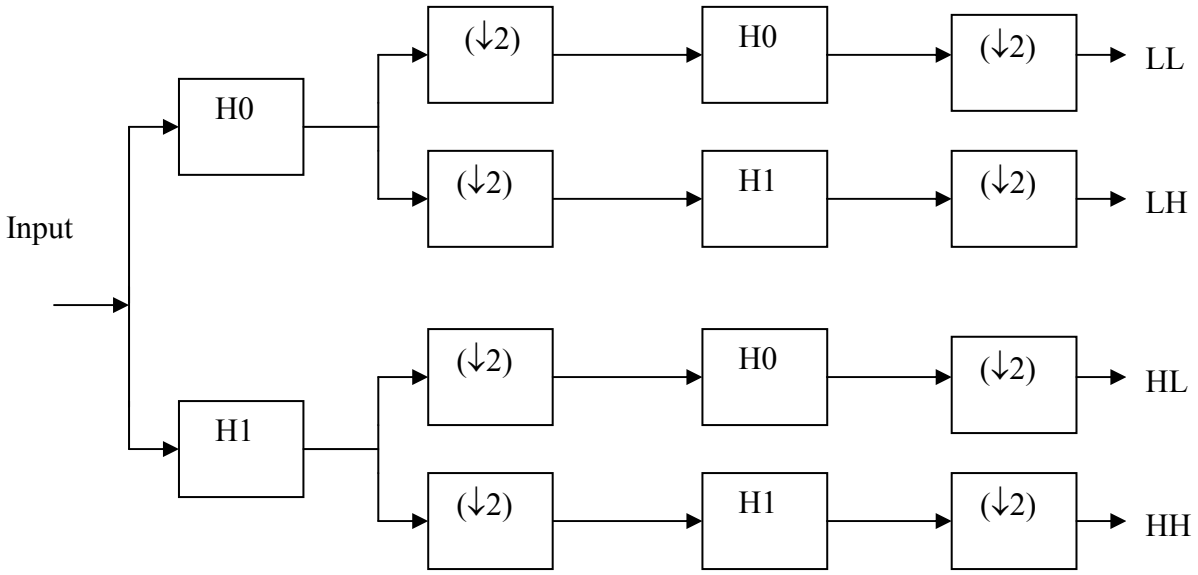

Figure 1.1. Representation of a 2 channel filter bank for analysis of images. H0 and H1 represent low pass and high pass filters respectively.(↓2) represents downsampling by 2.

$$\psi(t) = \sum_n h_1(n)\sqrt{2}\phi(2t-n) \qquad n\epsilon Z \tag{1.15}$$

The coefficients of expansion in (1.10) and (1.15) are the ones which become the filter coefficients in our filter bank.

1.3 Wavelets and Filter Banks

To obtain wavelet domain coefficients of an image, one applies a set of filters to rows and columns of the image, followed by down sampling. To reconstruct the image from the coefficients, one up samples them, and then subject them to a different set of filters which hold a certain relation with the filters on the transmitter side. Analysis can be represented as shown in Fig.1.1.

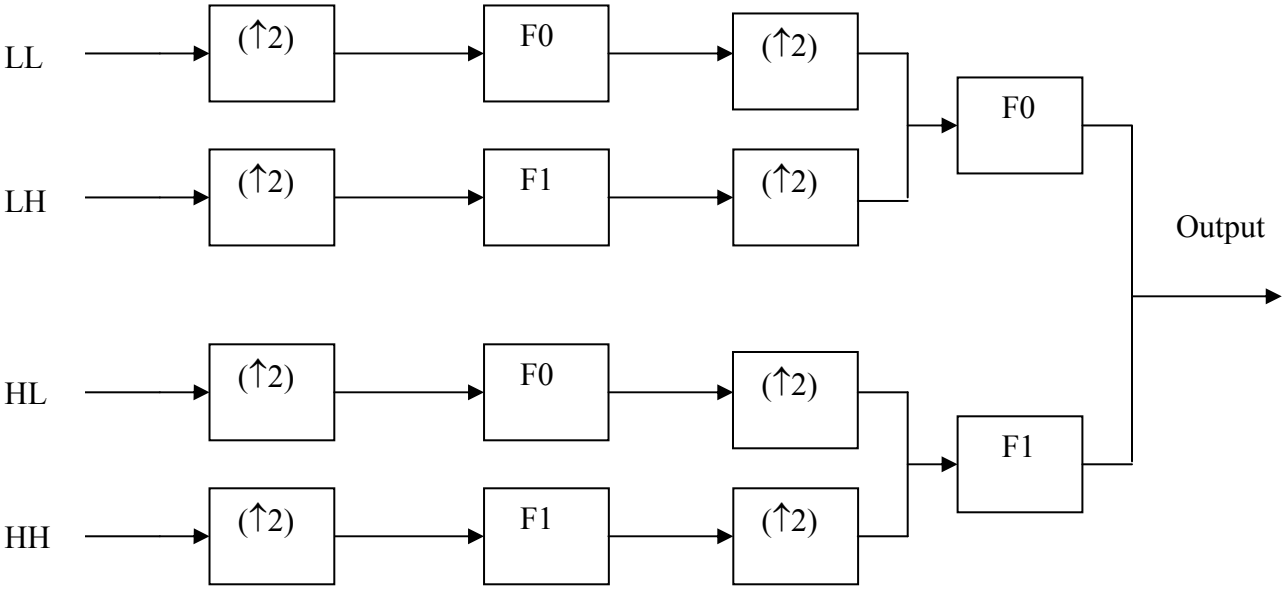

Figure 1.2. Synthesis Filter Bank. F0 and F1 are low pass and high pass filters respectively.(↑2) represents upsampling by 2.

In the top branch of Fig.1.1, the rows are filtered by H0, followed by down sampling of rows by 2. Filtering of columns by H0 and down sampling of columns by 2 would lead to a LL component. Filtering of columns by H1 followed by down sampling of columns by 2 would lead to a LH component. Similarly, filtering of rows by H1, down sampling by 2, followed by filtering by H0 and further down sampling by 2 would produce HL. Filtering of both rows and columns by H1 with down sampling at appropriate places would lead to HH.

The synthesis filter bank is shown in Fig.1.2. Each of the LL, LH, HL and HH components are given as input to the filter bank. Each of them are up sampled along columns by 2. LL is filtered along the columns by F0, up sampled along rows, and then filtered along rows by F0. LH is filtered along columns by F1, up sampled along rows, and filtered along rows by F0. HL is filtered along columns by F0, up sampled along rows, and filtered along rows by F1, while HH is filtered along columns by F1, up sampled along rows, and filtered along rows again by F1.

CHAPTER 2

JPEG 2000 STILL IMAGE COMPRESSION STANDARD

The growth in demand for a better image compression standard gave rise to the JPEG2000 still image compression standard, developed jointly by the International Organization for Standardization (ISO), International Telecommunication Union (ITU) and the International Electrotechnical Commission (IEC). In JPEG2000, the image coding system is not only optimized for efficiency, but also for scalability and interoperability in network and mobile environments. Internet and multimedia applications have become widespread today, and JPEG2000 provides a powerful tool for designers and users equally of networked image applications.

Some of the finer points of the standard, as discussed in [2], are as follows:

- Superior low bit-rate performance
- Continuous tone and bi-level compression
- Lossless and lossy compression
- Progressive transmission by pixel accuracy and resolution
- Region-of-interest (ROI) coding
- Open architecture
- Robustness to bit errors
- Protective image security

The JPEG2000 encoder and decoder are as shown in Fig.2.1. The image data is first subjected to a discrete transform, followed by quantization and entropy coding. At the decoder side, the operations follow the reverse order as those at the

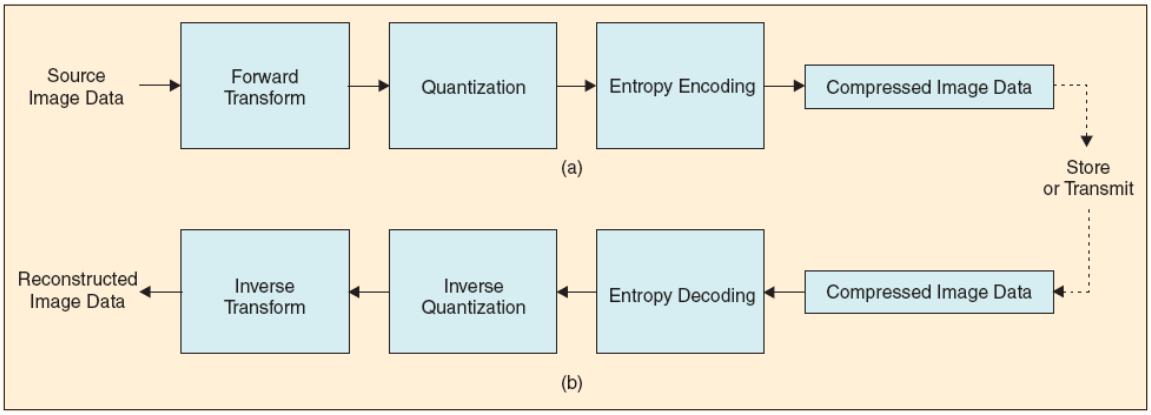

Figure 2.1. General block diagram of JPEG2000 (a)encoder and (b)decoder.[2].

encoder side. Entropy decoding is followed by dequantization, followed by an inverse transform.

One could briefly outline the steps involved in the entire process as follows [2]:

• The source image is first decomposed into components.

• The components are further decomposed into rectangular tiles. The tile component thus forms the basic component of the reconstructed image.

• Wavelet transform is applied to each tile, thereby decomposing the tile into different resolution levels.

• The resolution levels consist of subbands of coefficients that describe the frequency characteristics of local areas of the tile components.

• The subbands of coefficients are quantized and collected into arrays of code blocks.

• The bit planes of coefficients in a code block are entropy coded.

• The encoding can be done to allow certain regions of interest to be coded at a higher quality than the background.

• Markers are added to the bit stream for the purpose of error resilience.

• The code stream has a main header at the beginning that describes the original image and various decomposition and coding styles which are used to reconstruct the

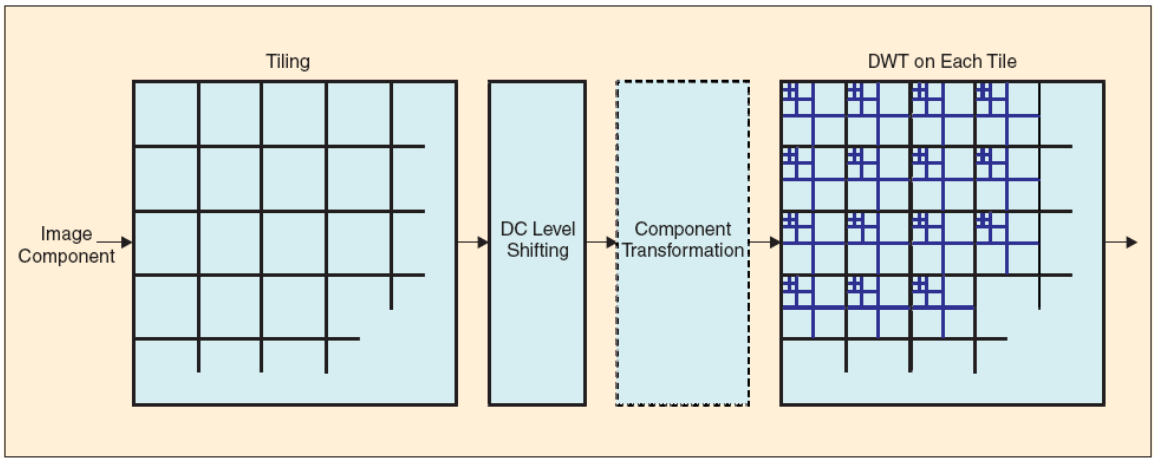

Figure 2.2. Tiling, DC-level shifting, color transformation and DWT of each image component.[2].

image as desired.

One could, on including dc-level shifting (which is subtraction of a constant value from all image pixels) and color transformation for color images (which is transformation between various color subspaces) depict the same process as shown in Fig.2.2.

2.1 JPEG2000 Filters

When wavelet transform is applied to the image blocks, they are decomposed into a number of subbands. These subbands contain coefficients that describe the horizontal and vertical spatial frequency characteristics of the original image block. For the forward transform, the 1-D set of samples is decomposed into low pass and high pass samples. While the low pass samples present the low resolution version of the original image component, the high pass samples present the residual version of the original image component, which is required for its perfect reconstruction.

Table 2.1. Daubechies 9/7 Analysis and Synthesis Filter Coefficients[2]

	Analysis Filter Coefficients	
i	Low-Pass Filter	High-Pass Filter
0	0.6029490182363579	1.115087052456994
±1	0.2668641184428723	-0.5912717631142470
±2	-0.07822326652898785	-0.05754352622849957
±3	-0.01686411844287495	0.09127176311424948
±4	0.02674875741080976	
	Synthesis Filter Coefficients	
i	Low-Pass Filter	High-Pass Filter
0	1.115087052456994	0.6029490182363579
±1	0.5912717631142470	-0.2668641184428723
±2	-0.05754352622849957	-0.07822326652898785
±3	-0.09127176311424948	0.01686411844287495
±4		0.02674875741080976

The transform can be irreversible or reversible. The default irreversible transform is the Daubechies 9/7 filter, whose analysis and synthesis coefficients are given in Table 2.1. The default reversible transform is given by means of Le Gall 5/3 filters, the coefficients of which are given in the Table 2.2. As one could see, they are symmetric filters.

The scaling functions for each of the filters discussed are shown in Fig.2.3.

Two filtering modes are supported in the JPEG2000 standard: convolution based and lifting based. The signal should be extended periodically for either of the modes to be implemented. The extension takes care of the correct filtering operation at the boundaries of the image. The number of samples by which the boundary is to be extended is actually dependent on the filter length.

The filtering operation is performed by buffering the entire image data and then applying the filters along the horizontal and vertical directions. After transformation, all of the coefficients are quantized. The quantization operation is lossy unless the

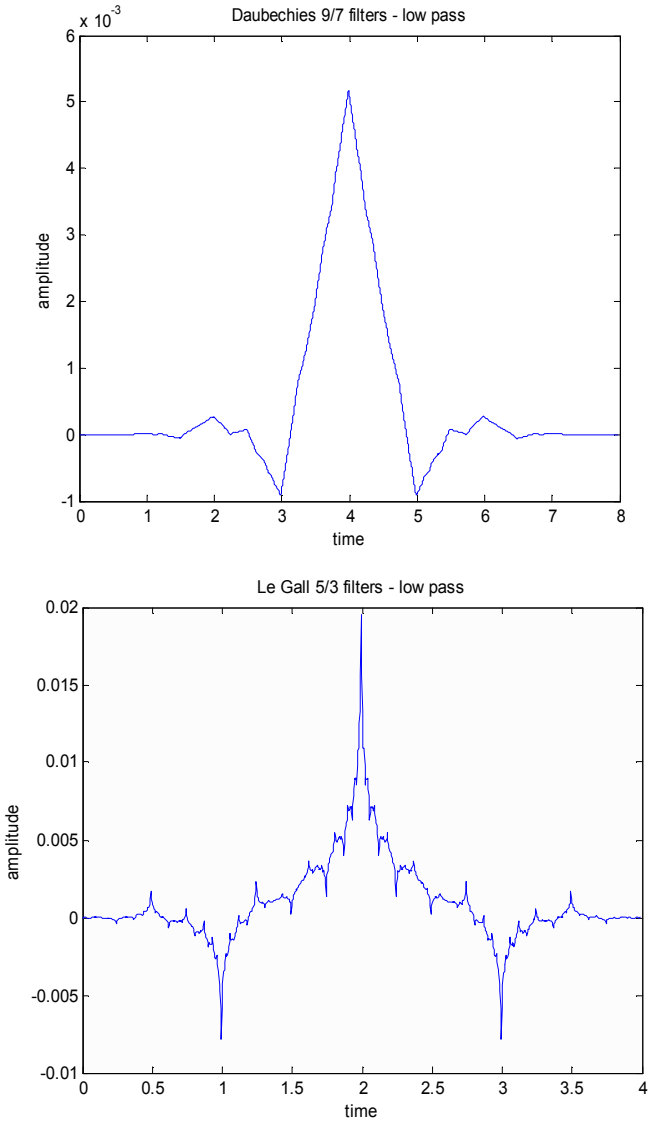

Figure 2.3. Scaling function shown for Daubechies 9/7 and Le Gall 5/3 filters.

quantization step is 1 and the coefficients are integers, as is the case in the reversible Le Gall 5/3 filters. Entropy coding, which follows the quantization step, is achieved by means of an arithmetic coding system. The following chapter discusses image mirroring and rotation in the wavelet domain.

Table 2.2. Le Gall 5/3 Analysis and Synthesis Filter Coefficients[2]

	Analysis Filter Coefficients	
i	Low-Pass Filter	High-Pass Filter
0	6/8	1
±1	2/8	-1/2
±2	-1/8	
	Synthesis Filter Coefficients	
i	Low-Pass Filter	High-Pass Filter
0	1	6/8
±1	1/2	-2/8
±2		-1/8

CHAPTER 3

IMAGE MIRRORING AND ROTATION

Image mirroring and rotation has been studied and implemented before in the literature [3,4,12,14]. The cases found in the literature deal with transforms such as the cosine transform and sine transform. This chapter deals with the wavelet transform as applied to the JPEG 2000 image codec[2].

This chapter begins with a discussion of the property of perfect reconstruction, followed by the key idea behind achieving image mirroring and rotation in the wavelet domain.

3.1 Property of Perfect Reconstruction

One says that a filter bank obeys the property of perfect reconstruction, when it recovers the input signal exactly. This section discusses the conditions for perfect reconstruction. Recall from chapter 1 that there were filters H_0 and H_1 on the analysis side (Fig.1.1), and F_0 and F_1 on the synthesis side (Fig.1.2). Here, a subscript of 0 denotes low pass filters, while a subscript of 1 denotes high pass filters. If the operators of upsampling and downsampling were not present, the condition of perfect reconstruction, with a l-step delay would, in the z-domain, result in [7]:

$$F_0(z)H_0(z) + F_1(z)H_1(z) = z^{-l} \tag{3.1}$$

An overall delay can be expected, as each individual filter is causal.

Now consider the upsampling and downsampling operators which introduce aliasing. In the z domain, v = ($\downarrow$2)x and u = ($\uparrow$2)x would result in

$$V(z) = \frac{1}{2}[X(z^{1/2}) + X(-z^{1/2})] \quad and \quad U(z) = X(z^2) \tag{3.2}$$

Therefore, downsampling followed by upsampling w = ($\uparrow$2)($\downarrow$2)x would result in W(z) = $\frac{1}{2}$(X(z) + X(-z)). The transform of ($\uparrow$2)($\downarrow$2)H_0x is $\frac{1}{2}(H_0(z)X(z) + H_0(-z)X(-z))$. The function turns out to be even. The aliasing term in the function $H_0(-z)X(-z)$ gets multiplied by $F_0(z)$ in the synthesis step. This term has to cancel the alias term $F_1(z)H_1(-z)X(-z)$ from the second channel. The condition for alias cancellation can be thus put across as [7]:

$$F_0(z)H_0(-z) + F_1(z)H_1(-z) = 0 \tag{3.3}$$

To summarize, consider a signal x(n) passed through the filter bank. After being passed through the low pass filter bank H_0, one gets H_0(z)X(z). Downsampling produces:

$$\tfrac{1}{2}[H_0(z^{1/2})X(z^{1/2}) + H_0(-z^{1/2})X(-z^{1/2})]$$

Upsampling produces:

$$\tfrac{1}{2}[H_0(z)X(z) + H_0(-z)X(-z)]$$

Multiplying by $F_0(z)$ produces the output from the low pass channel. Similarly, the output from the high pass channel can be obtained by replacing $H_0(z)$ by $H_1(z)$ and $F_0(z)$ by $F_1(z)$. Hence :

low pass channel: $\frac{1}{2}F_0(z)[H_0(z)X(z) + H_0(-z)X(-z)]$

high pass channel: $\frac{1}{2}F_1(z)[H_1(z)X(z) + H_1(-z)X(-z)]$

On combining the output from the above two channels, one gets:

$$\frac{1}{2}[F_0(z)H_0(z) + F_1(z)H_1(z)]X(z) + \frac{1}{2}[F_0(z)H_0(-z) + F_1(z)H_1(-z)]X(-z) \quad (3.4)$$

For perfect reconstruction to hold true for a two-channel filter bank, two conditions are thus needed :

$$F_0(z)H_0(z) + F_1(z)H_1(z) = z^{-l} \quad (3.5)$$

$$F_0(z)H_0(-z) + F_1(z)H_1(-z) = 0 \quad (3.6)$$

3.2 Product Filters

A product filter is defined as follows[7]:

$$P_m(z) = F_m(z)H_m(z), \qquad m = 0, 1. \quad (3.7)$$

For the purpose of alias cancellation, suppose one choses $F_0(z) = H_1(-z)$ and $F_1(z)$ = -$H_0(-z)$. In this case, it can be found that $P_1(z)$ = -$P_0(-z)$.

$$P_1(z) = -H_0(-z)H_1(z) = -H_0(z)F_0(z) = -P_0(-z) \quad (3.8)$$

The reconstruction equation can thus be written as[1] $P_0(z)$ - $P_0(-z) = 2z^{-l}$. Since the left side of the equation is odd, hence the factor l is also odd. On normalizing $P_0(z)$ by z^l, one gets P(z) = $z^l P_0(z)$. Thus P(-z) = $-z^l P_0(-z)$. The reconstruction equation is thus $P_0(z)$ - $P_0(-z) = 2z^{-l}$, which when multiplied by z^l, becomes:

$$P(z) + P(-z) = 2 \tag{3.9}$$

That is, the product filter should thus be a half band filter.

3.3 Mirroring and Rotation

It is known that after filtering and downsampling, one would have the following the frequency domain:

$$\frac{1}{2}[X(z^{1/2})H_m(z^{1/2}) + X(-z^{1/2})H_m(-z^{1/2})] \qquad m = 0, 1 \tag{3.10}$$

Suppose one flips the obtained sequence in the time domain, one would get the following in the z domain, following the z transform property.

$$\frac{1}{2}[X(z^{-1/2})H_m(z^{-1/2}) + X(-z^{-1/2})H_m(-z^{-1/2})] \qquad m = 0, 1 \tag{3.11}$$

It is found that (3.11) is actually what one would receive if one were to filter the reverse sequence using reverse analysis filters. Suppose one wants to obtain the reverse

[1]The 2 or any constant factor on the right side of the expression comes into the picture as a scaling factor. It does not change anything else in the reconstruction conditions.

sequence as the final output; in that case, one could reverse the synthesis filters and proceed with the synthesis. This is equivalent to saying that reversing the input and the filter would produce the output in reverse.

Now consider the case where the analysis and synthesis filters are symmetric, as is the case of JPEG2000 9/7 and 5/3 filters. For symmetric filters, the coefficients in forward and reverse order are the same, and hence, from the property of Z transforms, it is known that

$$H_m(z) = H_m(z^{-1}) \qquad m = 0, 1 \tag{3.12}$$

Hence, on replacing $H_m(z^{-1/2})$ in (3.11) by $H_m(z^{1/2})$, one obtains

$$\frac{1}{2}[X(z^{-\frac{1}{2}})H_m(z^{\frac{1}{2}}) + X(-z^{\frac{-1}{2}})H_m(-z^{\frac{1}{2}})] \qquad m = 0, 1 \tag{3.13}$$

(3.13) is now the same as filtering the reverse sequence by analysis filters.

Now, let LL, LH, HL and HH be the four components obtained by passing the image information through the two channel filter bank, each component denoting the output of a particular branch in the filterbank. Now, in the synthesis bank, let LL', LH', HL' and HH' denote the branches which accept the four components, LL, LH, HL and HH respectively. Let

$$J = \begin{bmatrix} 0000....1 \\ 000....10 \\ 00....100 \\ 0....1000 \\ 1....0000 \end{bmatrix}$$

be a reverse identity matrix of the same dimension as LL, LH, HL and HH components. Then, one can form the mirrored and rotated images as is discussed next.

In each of the figures, the synthesis filter bank(SFB) has four nodes where it accepts LL, LH, HL and HH components. In the case of horizontal mirroring, one simply, reverses each of the LL, LH, HL and HH components and apply it to the synthesis filterbank - i.e., post multiply each component matrix by J and then give it to the filterbank. For vertical mirroring, on the other hand, premultiply each component by J instead of postmultiplying.

For image rotation by 90 degrees, post multiply by J, take the transpose, and then apply to SFB. One point to be noted is that, here, the[2] $(LH * J)^T$ is given to HL' and $(HL * J)^T$ is given to LH'. For 270 degrees, similar set of steps are adopted - the only difference is that here one premultiplies by J instead of post multiplying.

For 180 degrees on the other hand, one simply has to premultiply and post multiply by J, each of the four components, and apply them to the filterbank.

[2]The asterisk * denotes matrix multiplication operation

3.4 Results

The scheme outlined above was applied to the standard images,the Lena and the Girl images. The output aftter mirroring and rotation operations, in comparison with the original figure is as shown in this section in Fig.3.6 and Fig.3.7.

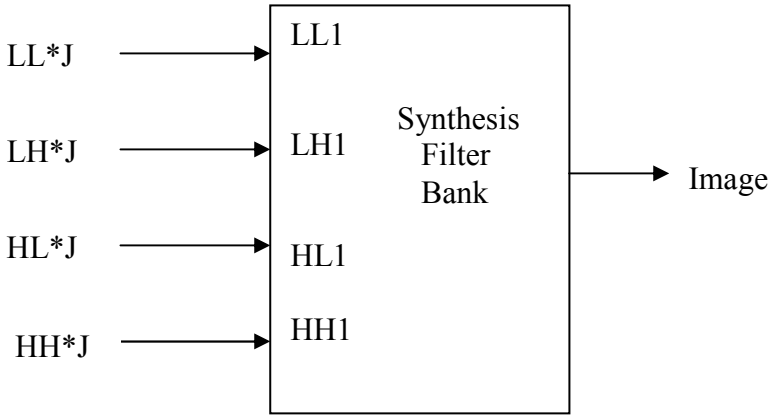

Figure 3.1. Horizontal mirroring.

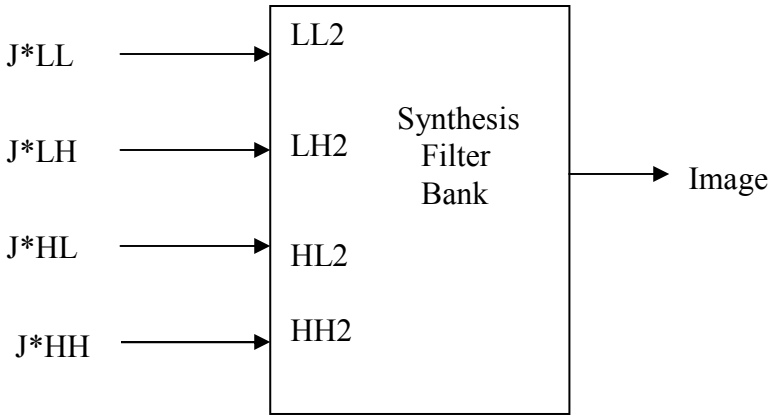

Figure 3.2. Vertical mirroring.

The topic of image mirroring and rotation was been discussed so far. The next section discusses image interpolation, or resizing of images.

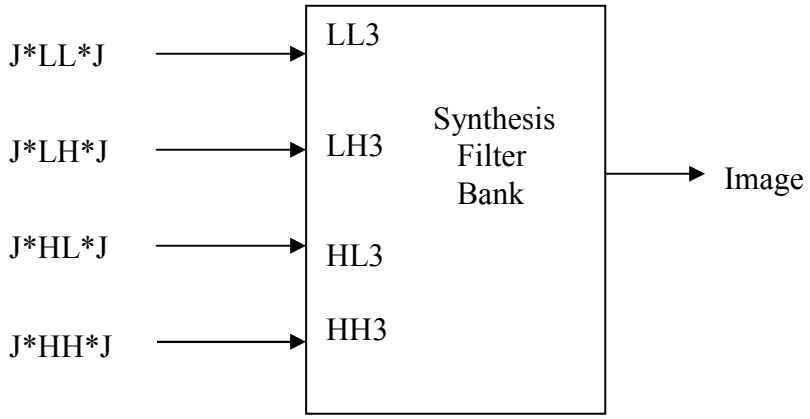

Figure 3.3. Rotation by 180°.

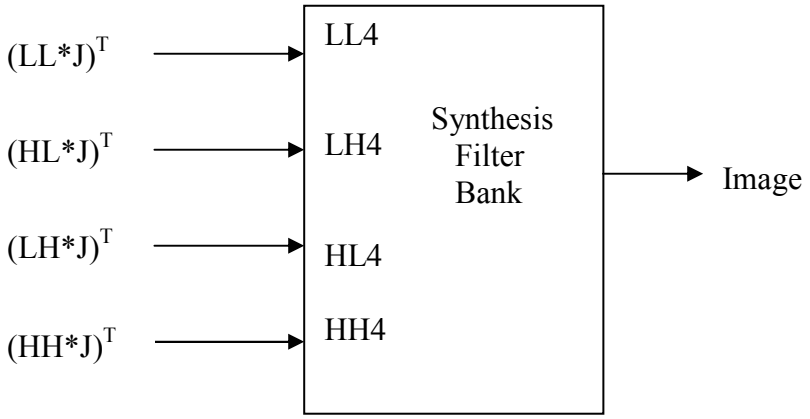

Figure 3.4. Rotation by 90°.

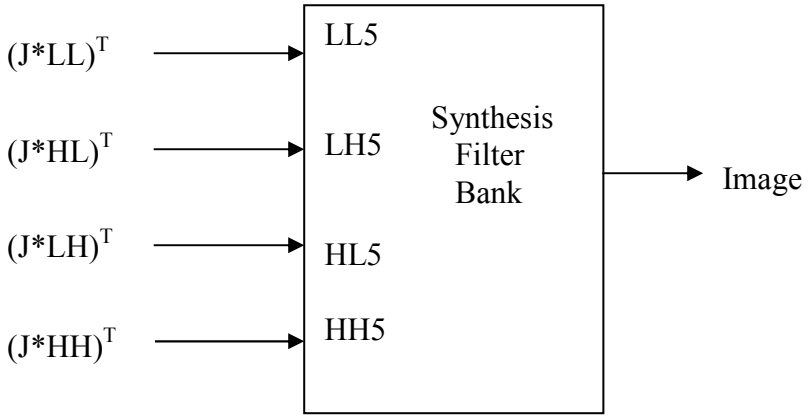

Figure 3.5. Rotation by 270°.

Figure 3.6. Lena image with 9/7 Daubechies filter coefficients. Top left - horizontal mirroring, Top center - rotation by 180°, Top right - rotation by 270°. Bottom left - vertical mirroring, Bottom right - rotation by 90°.

Figure 3.7. Girl image with 5/3 Le Gall filter coefficients. Top left - horizontal mirroring, Top center - rotation by 180°, Top right - rotation by 270°. Bottom left - vertical mirroring, Bottom right - rotation by 90°.

CHAPTER 4

IMAGE INTERPOLATION

The aim of image interpolation is to provide a higher resolution image from a lower resolution one. Many of the solutions to this problem rely on a statistical approach[1]. A prominent statistical approach [1], makes use of a Hidden Markov Tree model [10,11]. In [1], the wavelet coefficients at various scales are considered to be occupying the nodes of the Hidden Markov Tree (HMT). A hidden state value, determined based on the significance of a coefficient, actually dictates the value of the coefficient.

A parent child relationship is assumed to exist between the various nodes of the tree. Once the state of the parent coefficient is known, the probability density function (pdf) of the child coefficients is determined, which would be given by a Gaussian mixture model. Once the pdf is known, the child coefficients are generated randomly. Estimation of the parameters of the pdf is another interesting aspect of the solution discussed in the paper [1].

4.1 Interpolation error modelling

In the method proposed here, one initially takes an image and decompose it into LL, LH, HL and HH bands. One then takes the LL component to be the low resolution image, and try to predict the rest of the components, and then reconstruct the original image. The original image is then compared with the reconstructed image to draw conclusions about the effectiveness of the algorithm.

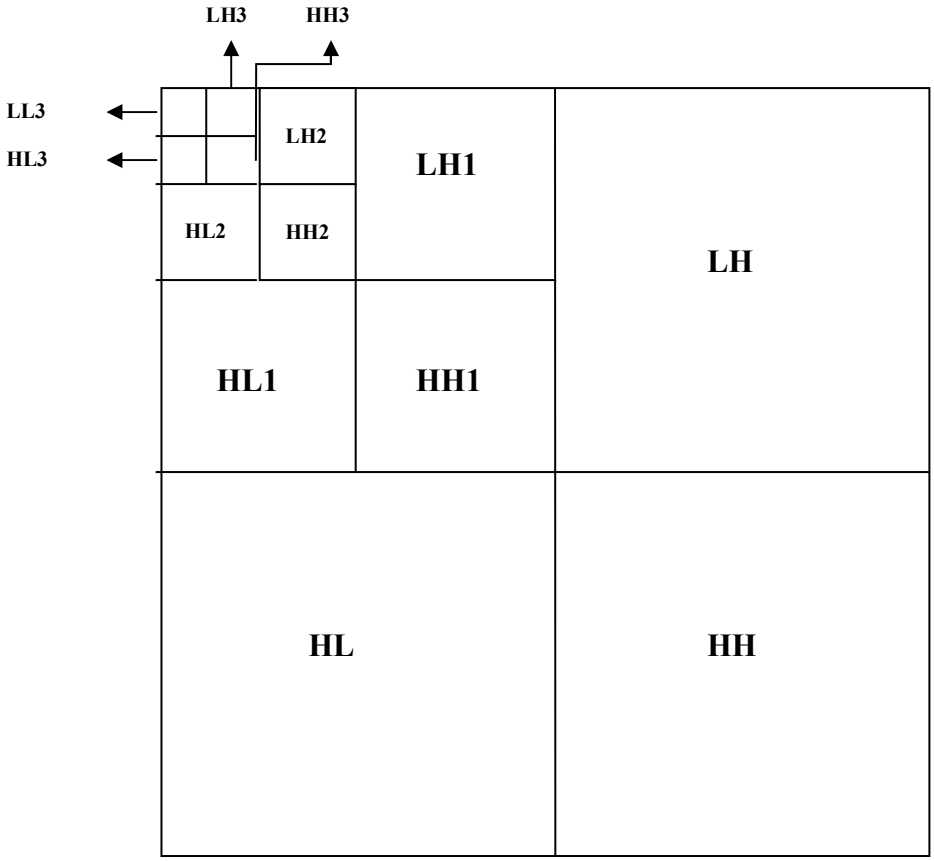

Figure 4.1. Decomposition of image into various levels.

In order to obtain the LH, HL and HH components from LL, LL decomposed into further three levels. Decomposing LL, one obtains level one components, say, LL_1, LH_1, HL_1 and HH_1. LL_1 is then further decomposed to obtain level two components, say LL_2, LH_2, HL_2 and HH_2. LL_2 is decomposed to LL_3, LH_3, HL_3 and HH_3 (Fig.4.1). One then proceeds to interpolate the lower level components to obtain the higher level ones, and then compute the interpolation error for each of the components at each level.

For example, suppose that one applies bilinear interpolation to LH_3 to obtain LH_2', which is of the same dimension as LH_2. Compare it with LH_2 and compute interpolation error. Let the error information be named as LH_3 interpolation er-

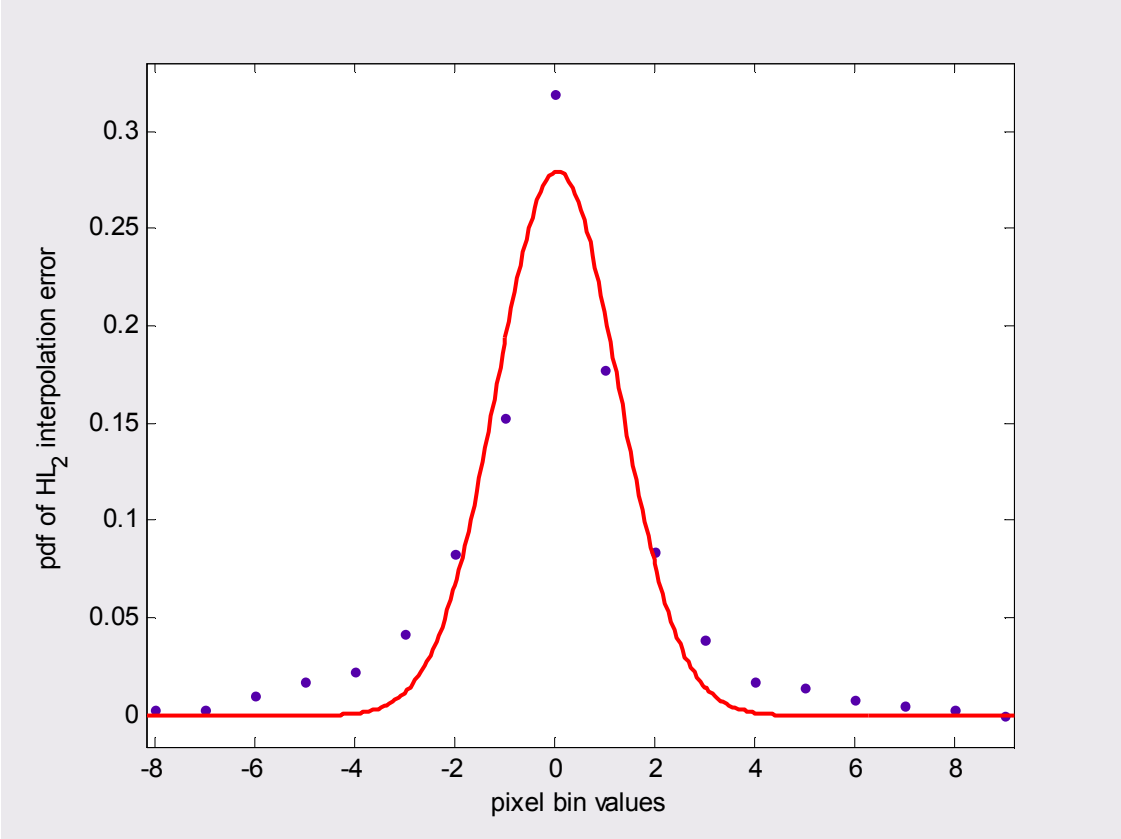

Figure 4.2. Gaussian curve fitted for HL_2 interpolation error. RMSE of fit = 0.0219, mean of distribution = 0.0578, standard deviation of distribution = 1.715.

ror. Similarly, interpolate LH_2 to obtain LH_1, and then compute error, which would then be termed as LH_2 interpolation error. Repeat the procedure for HL and HH components as well.

In the next stage, one tries to model the probability distribution of each of the error information thus obtained. It was found that a Gaussian curve fits the probability distribution of each of the error information obtained with a reasonable degree of accuracy, as shown in Fig.4.2 and Fig.4.3. Thus, a Gaussian curve was fitted to the probability distribution of each of the error information at each level.

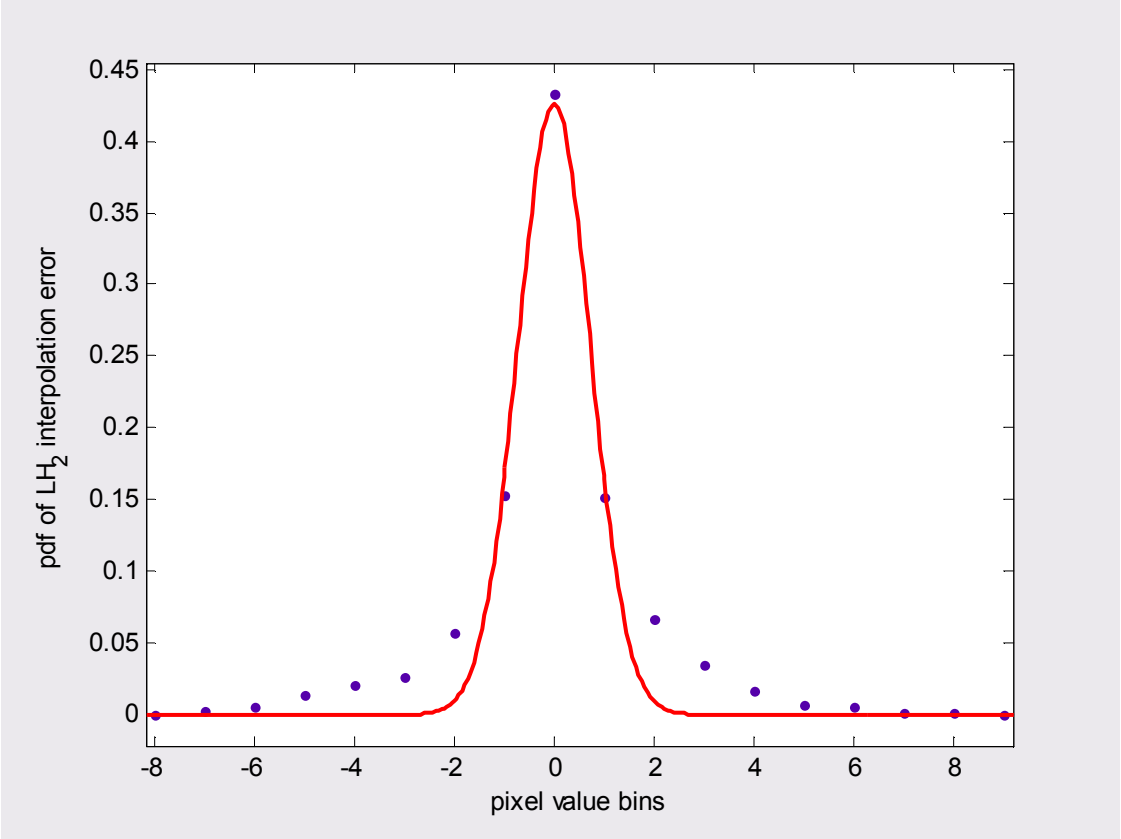

Figure 4.3. Gaussian curve fitted for LH_2 interpolation error. RMSE of fit = 0.0244, mean of distribution = -0.0017, standard deviation of distribution = 1.026.

4.2 Interpolation error prediction and denoising

One now has the interpolation error probability distribution for levels 3 and 2. Next one tries to predict the interpolation error distribution for level 1, i.e., the probability distribution of the error one would obtain, were one to predict LH, HL and HH components from their level 1 counterparts. It can be observed that the mean remains more or less the same across the levels, but variance of distribution differs. One can make use of a property of wavelet coefficients while trying to predict the variance of the error distribution. One considers the property of exponential decay of coefficients across the bands at various levels. It can be explained as [1]: the variances for the error m $(\sigma_m^K)^2$ in scale K are estimated according to

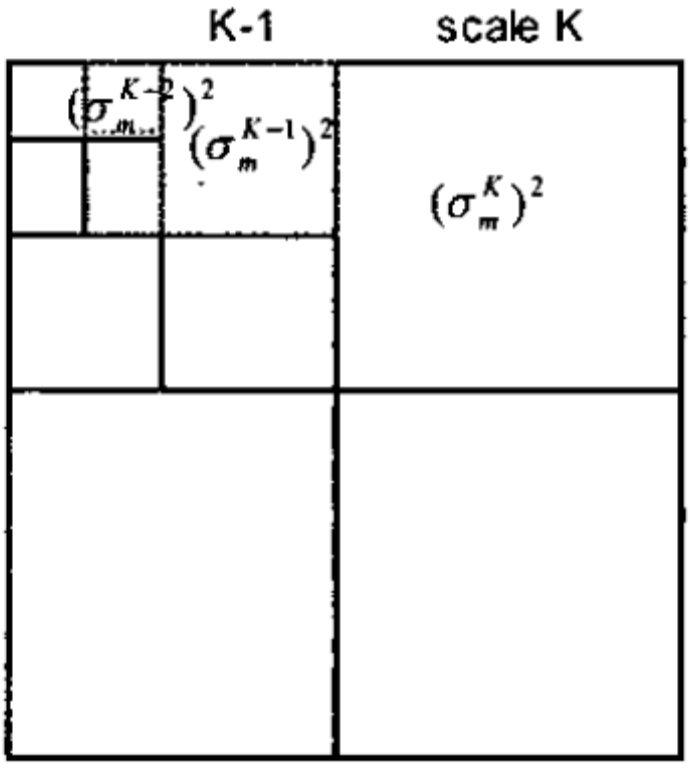

Figure 4.4. Exponential decay of variances across scales[1].

$$(\sigma_m^K)^2 = \frac{(\sigma_m^{K+1})^2}{(\sigma_m^{K+2})^2} \cdot (\sigma_m^{K+1})^2 \tag{4.1}$$

where m denotes if it is LH, HL or HH, while K denotes the level at which coefficients are located, LL being the highest level, and LL_3 being the lowest level.

The exponential decay of variances across scales are shown in Fig.4.4. The same model is essentially adopted for interpolation errors.

Thus, once the mean and variance are known, one has all of the parameters required to specify the interpolation error distribution of level 1. Next, one proceeds

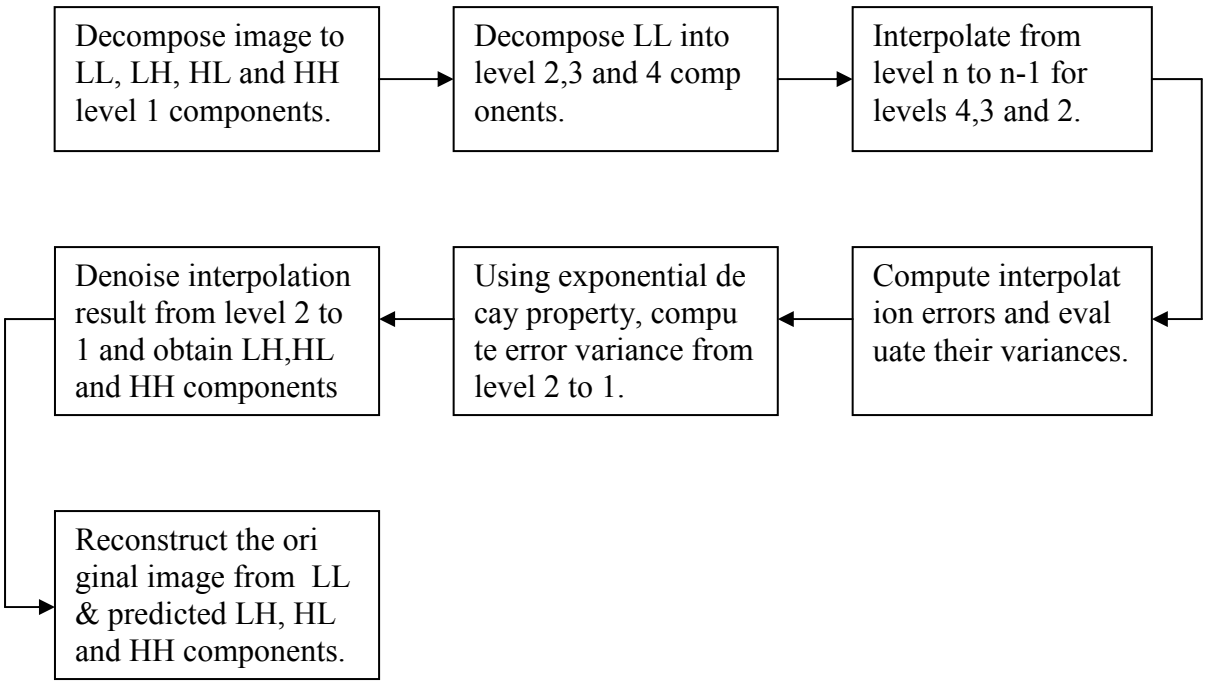

Figure 4.5. Block diagram representing steps in image interpolation.

to interpolate level one components to obtain LH',HL' and HH'. The interpolated components, as well as the probability distribution of the error are thus obtained. This error is then treated as noise which was added to the original LH, HL and HH components.

This leads to the next step in our interpolation algorithm, which is, image denoising in the wavelet domain. One makes use of an already existing image denoising algorithm [5]. The algorithm receives an input matrix and noise distribution characteristics, and produces a denoised image. One feeds the interpolated components, as well as predicted interpolation error distribution characteristics, and obtain denoised LH, HL and HH components, thus completing the estimation of LH, HL and HH components. Once these components are present, one can recontruct the original image making use of the LL component already present.

Table 4.1. Interpolation Results.(PSNR in dB)

Image	Bicubic Method	HMT based[1]	Proposed Method
Lena	29.47	33.97	34.02
Girl	32.70	36.10	36.18
Boat	27.28	30.26	30.30
Peppers	39.98	41.74	42.26

4.3 Results

Lena as well as Girl images in original and interpolated form is shown in this section. A comparison with exisitng methods in terms of PSNR(dB) is shown in table 4.1.

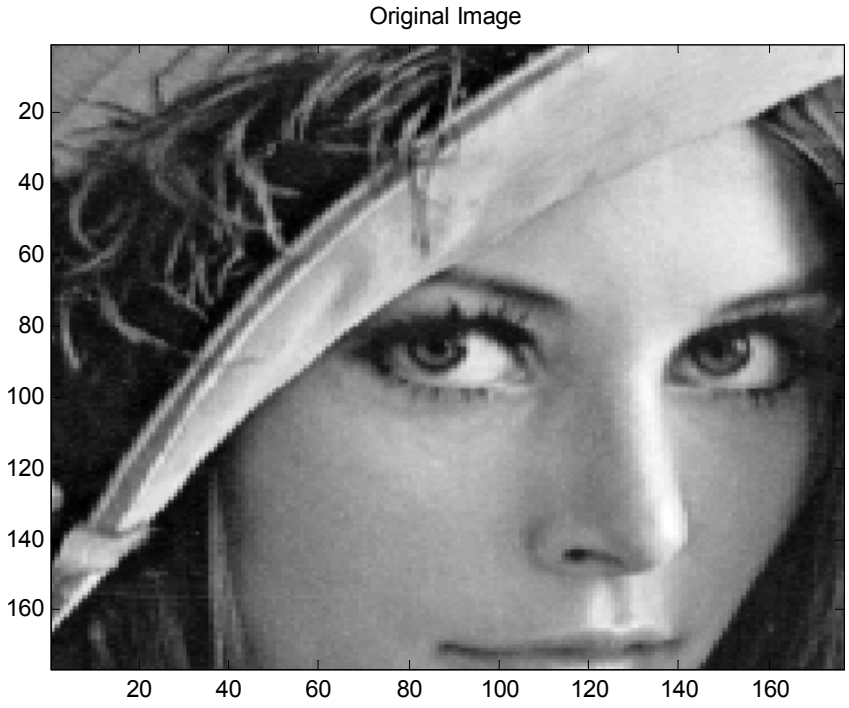

Figure 4.6. Original Lena Image.

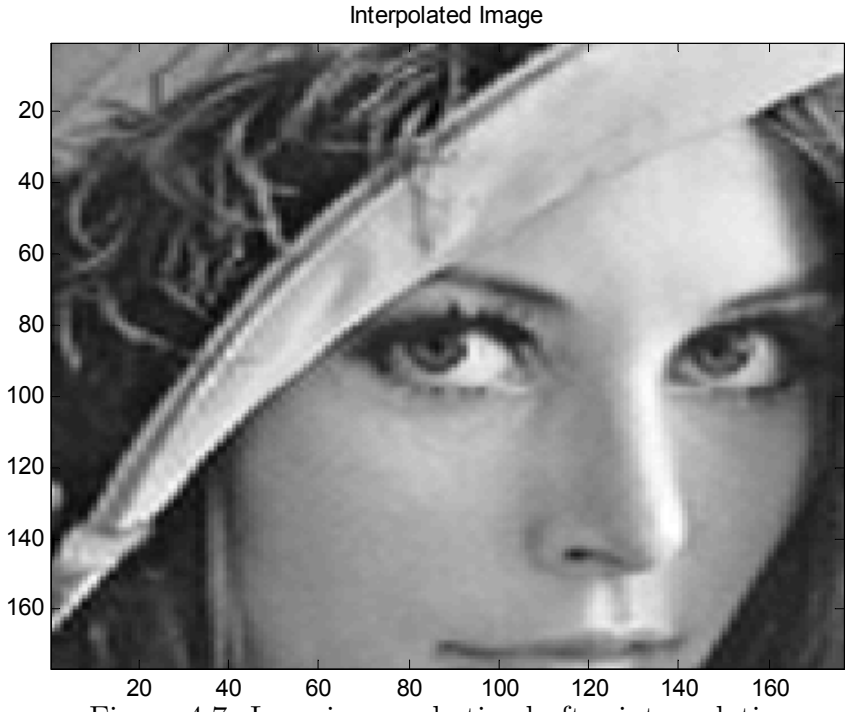

Figure 4.7. Lena image obatined after interpolation.

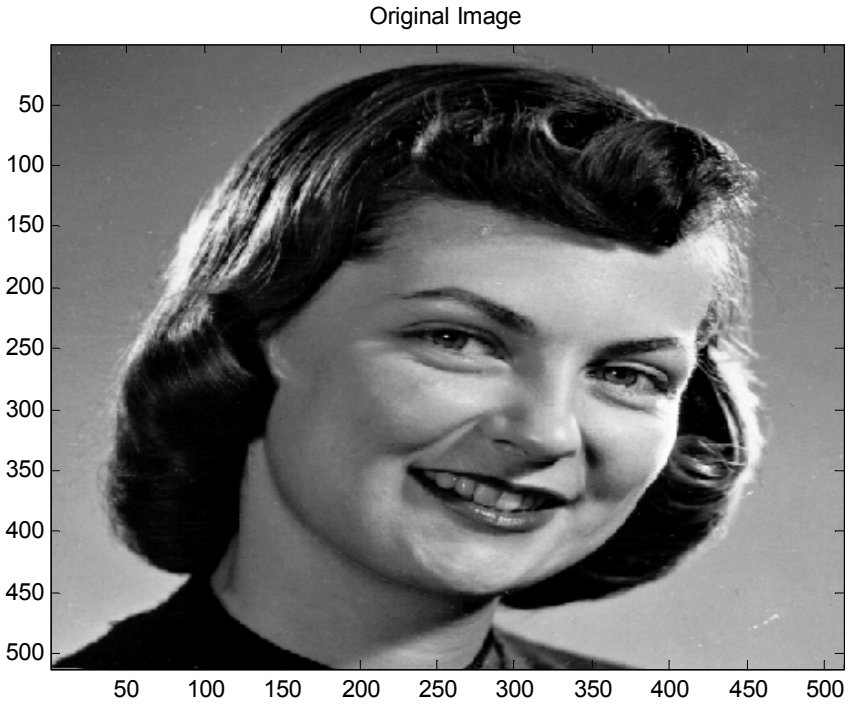

Figure 4.8. Original Girl Image.

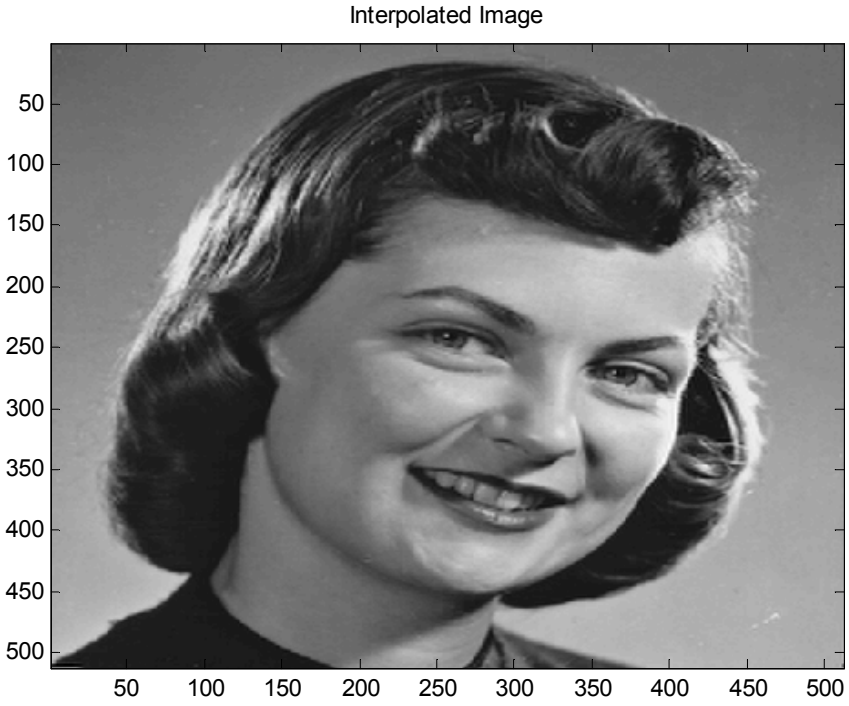

Figure 4.9. Girl image obatined after interpolation.

CHAPTER 5

CONCLUSIONS

This chapter concludes, and also give directions for possible future work. A method for image mirroring and rotation in wavelet domain in the case of JPEG2000 filters has been presented. One could investigate extending the case for a general perfect reconstruction filter bank, including the asymmetric cases too. One could explore rotaion by 45° and 135° as well. In the case of image interpolation, currently the image error is modeled as Gaussian. Since the wavelet coefficients themselves are modelled as a mixture of Gaussian functions, one could try to model the noise in the same manner, and denoise the image for each Gaussian function seperately.

One could extend the mirroring, rotation and interpolation methods to color images as well. Mirroring and rotation is straighforward, since one simply has to repeat what was done for grayscale images for each of the color components (Fig.5.1). For rotation by a certain angle, for example, one would have to rotate each of the color components by the same angle. A simple block diagram representation for image interpolation for color images is shown in Fig.5.2.

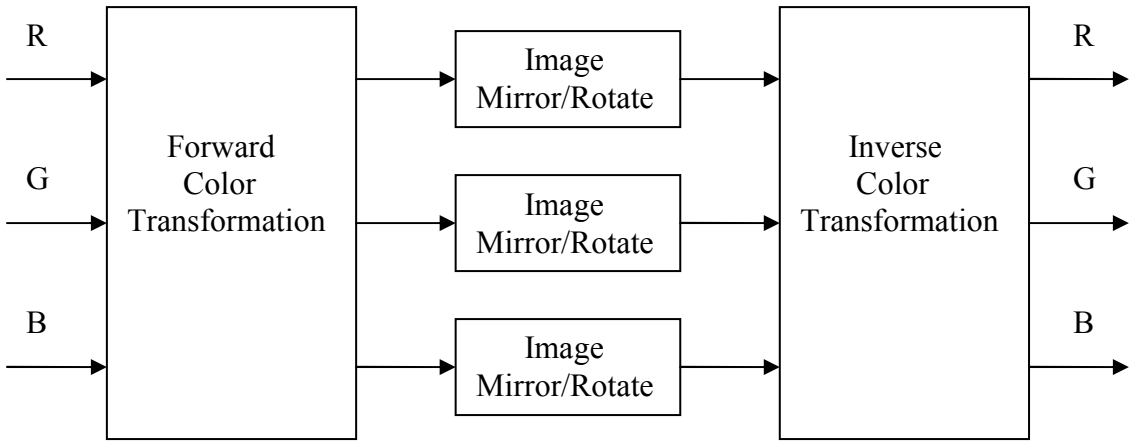

Figure 5.1. Image mirroring for a color image whose red(R), green(G) and blue(B) components are given.

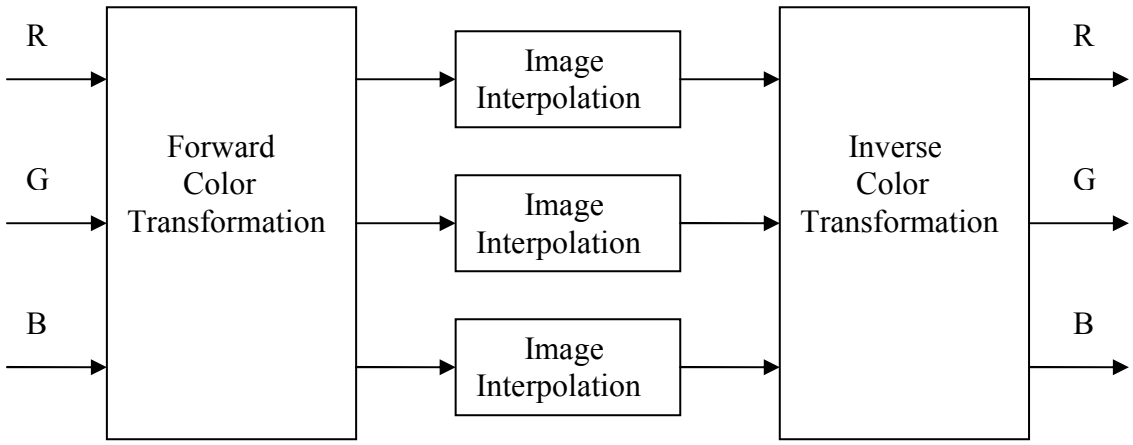

Figure 5.2. Image interpolation for a color image whose red(R), green(G) and blue(B) components are given.

APPENDIX A

IMAGE DENOISING ALGORITHM

The image denoising algorithm used here is based on the work in [5]. In [5], a method for the removal of noise from digital images based on a statistical model of the coefficients of an overcomplete multiscale oriented basis is described. The coefficients are modeled based on a Gaussian scale mixture (GSM), the product of a Gaussian random vector, and an independent hidden random scalar multiplier. A random vector $\mathbf{x}$ is a GSM, if and only if it can be expressed as the product of a zero-mean Gaussian vector $\mathbf{u}$ and an independent positive random variable $\sqrt{z}$

$$\mathbf{x} = \sqrt{z}\mathbf{u} \tag{A.1}$$

Here, equivalent denotes equivalent in distribution. The variable z is the multiplier. The vactor x is thus an infinite mixture of Gaussian vectors, whose density is determined as

$$p_x(\mathbf{x}) = \int p(\mathbf{x}/z)p_z(z)dz \tag{A.2}$$

The pdf of $\mathbf{x}$ when conditioned on z is Gaussian. The probability density $p_z(z)$ need to be specified in order to complete the model. A noninformative prior known as Jeffrey's prior is used for the same and in the current context, one obtains

$$p_z(z) \propto \frac{1}{z} \tag{A.3}$$

The procedure for image denoising can be outlined as follows:

1)Decompose the image into pyramid sub bands at different scales and orientations.

2)Denoise each sub band.

3)Invert the pyramid transform and obtain the denoised image.

The assumption made is that the image is corrupted by independent additive white Gaussian noise of known variance. Thus, a vector $\mathbf{y}$ corresponding to a neigh-

bourhood of N observed coefficients of the pyramid representation can be expressed as:

$$\mathbf{y} = \mathbf{x} + \mathbf{w} = \sqrt{z}\mathbf{u} + \mathbf{w} \tag{A.4}$$

Both $\mathbf{u}$ and $\mathbf{w}$ are zero-mean Gaussian vectors, with associated covariance matrices $\mathbf{C_u}$ and $\mathbf{C_w}$. Further details about obtaining them are given in [5].

For each neighbourhood, one estimates x_c, the reference coefficient at the center of the neighbourhood, from $\mathbf{y}$, the set of observed coefficients. The Bayes least squares (BLS) estimate is the conditional mean

$$E(x_c|\mathbf{y}) = \int_0^\infty p(z|\mathbf{y})E(x_c|\mathbf{y}, z)dz \tag{A.5}$$

Here, the coefficient neighbourhood vector $\mathbf{x}$ is Gaussian when conditioned on z. The expected value in (A.5) would thus become a local linear estimate, which is given by:

$$E(x|\mathbf{y}, z) = z\mathbf{C_u}(z\mathbf{C_u} + \mathbf{C_w})^{-1}\mathbf{y} \tag{A.6}$$

The remaining component in (A.5) is computed as

$$p(z|\mathbf{y}) = \frac{p(\mathbf{y}/z)p_z(z)}{\int_0^\infty p(\mathbf{y}|\alpha)p_\alpha(\alpha)d\alpha} \tag{A.7}$$

.

Further details and results of denoising can be found in [5].

REFERENCES

[1] D.H.Woo, I.K.Eom and Y.S.Kim, Image interpolation based on inter-scale dependency in wavelet domain, *IEEE ICIP,*vol. 3, pp. 1687 - 1690, Oct.2004.

[2] A.Skodras, C.Christopoulos, and T.Ebrahimi, The JPEG2000 still image compression standard, *IEEE Signal Processing Magazine,*vol. 18, pp. 36-58, Sept 2001.

[3] B. Shen and I. K. Sethi, Inner-block operations on compressed images, *Proc. of the Third ACM International Conference on Multimedia,* pp. 489-498, San Francisco, CA, Nov.1995.

[4] D.N.Kim and K. R. Rao, Sequence mirroring properties of orthogonal transforms having even and odd symmetric vectors, *ECTI Transactions on Computer and Information Technology,*vol. 3, Nov.2007.

[5] J.Portilla *et al*, Image denoising using scale mixtures of Gaussians in the wavelet domain, *IEEE Trans.Image Processing,*vol. 12, pp.1338-1351,Nov.2003.

[6] C.S.Burrus, R.A.Gopinath and H.Guo, *Wavelets and wavelet transforms,* Prentice-Hall, 1998.

[7] G.Strang and T.Nguyen, *Wavelets and filterbanks,* Wellesley-Cambridge Press, 1997.

[8] M.Vetterli and C.Herley, Wavelets and filterbanks: theory and design, *IEEE Trans.Signal Processing,*vol. 40, pp.2207-2232, Sept.1992.

[9] S. K. Mitra, *Digital signal processing,* Tata McGrawHill, 2006.

[10] J.K.Romberg, H.Choi and R.G.Baraniuk, Bayesian tree-structured image modeling using wavelet-domain hidden Markov models, *IEEE Trans.Image Processing,*vol. 10, pp.1056-1068,July,2001.

[11] M.S.Crouse, R.D.Nowak and R.C.Baraniuk, Wavelet-based statistical signal processing using hidden Markov models, *IEEE Trans.Signal Processing,*vol.46, pp.886 - 902, Apr.1998

[12] R.L. de Queiroz, Processing JPEG-compressed images and documents, *IEEE Trans.Image Processing,*vol. 7, pp. 1661-1672, Dec 1998.

[13] S.Mallat, *A wavelet tour of signal processing,* Academic Press, 1999.

[14] D.N.Kim and K. R. Rao, Two-dimensional discrete sine transform scheme for image mirroring and rotation, *Journal of Electronic Imaging,*vol. 17, pp.013011-1 to 013011-4, Jan - Mar 2008.

[15] J.M.Shapiro, Embedded image coding using zerotrees of wavelet coefficients, *IEEE Trans.Signal Processing,*vol.41, pp.3445-3462, Dec.1993.

BIOGRAPHICAL STATEMENT

Theju Isabelle Jacob received her Bachelor of Technology degree in Electronics and Communication from the University of Calicut, Kerala, India in 2005 and her Master of Science degree in Electrical Engineering from the University of Texas at Arlington in 2008. She worked as associate software engineer in Birla Soft Ltd, Bangalore, India, from 2005 to 2006. From September 2007 to May 2008, she was with FastVDO Inc, Columbia, Maryland as graduate intern. Her research interests are in the areas of wavelets and filterbanks, image and video coding and video streaming.