CHAPTER VI
WAVELET PHASE FILTER

6.1 Introduction

In the previous chapter, we saw that the modulus maxima of wavelet transforms can be used to represent the multiscale edges of an image. For a signal with low level noise, the denoising algorithm which discards modulus maxima below a threshold is effective for removing noise from the contaminated signal. But for a signal with a high level of noise, like PET noise, the thresholding technique is not so effective. In this chapter, we study the probability distribution of phase of 2D symmetric Haar wavelet transform of noise, and develop denoising schemes exploiting the phase distribution of noise. Numerical experiments applying the schemes to remove Poisson noise occurring in PET sinogram are also discussed; they indicate that the schemes improves the reconstruction of PET image, compared with algorithms based solely on thresholding.

The remainder of this chapter is organized as follows: First, we extend the symmetric wavelet transforms to the 2D case and discuss the features of PET raysum images and the limitations of wavelet modulus based denoising schemes for such applications. Second, we study the distribution of the wavelet phase of noise. Third, we discuss some implementation issues of the symmetric Haar wavelets. Then, we describe wavelet phase based denoising algorithms. Finally, we describe our numerical experiments and analyze the algorithms theoretically and experimentally.

6.2 Two-dimensional Haar Wavelet Transform

In this section, we first extend symmetric Haar wavelet transform to the two-dimensional case and define wavelet modulus and wavelet phase for the two-dimensional case. Then, we discuss two-dimensional wavelet modulus based multiscale edge detection and denoising schemes.

6.2.1 2D Symmetric Haar Transform. Recall that the 1D symmetric Haar matrix pair is defined in the previous chapter by

$$G = \frac{1}{2}\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & ... & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & ... & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & ... & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & ... & 0 & 0 \\ & & & & & ... & & \\ 0 & 0 & 0 & 0 & 0 & ... & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & ... & 1 & 0 \end{bmatrix}$$

and

$$H = \frac{1}{2}\begin{bmatrix} 0 & -1 & 0 & 0 & 0 & ... & 0 & 1 \\ 1 & 0 & -1 & 0 & 0 & ... & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & ... & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & ... & 0 & 0 \\ & & & & & ... & & \\ 0 & 0 & 0 & 0 & 0 & ... & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & ... & 1 & 0 \end{bmatrix}.$$

For signal $f = (f(0), f(1), f(2), ..., f(N-1))$, the symmetric Haar transform of f is defined by

$$\begin{cases} u = Gf \\ v = Hf \end{cases} \tag{6.2.1}$$

where u is referred to as the lowpass component and v is referred to as the highpass component. Also, we knew that the signal f can be reconstructed by

$$f = G^T u + H^T v. \tag{6.2.2}$$

To extend the 1D symmetric Haar transform the 2D case, we first express the preceding equations in the form of linear convolution. Let

$$f = (..., f(-2), f(-1), f(0), f(1), f(2), ...)$$

and

$$h = (..., h(-2), h(-1), h(0), h(1), h(2), ...)$$

denote two sequences, which can be used to represent signals and filters. The linear convolution of f and h is defined by

$$(h \otimes f)(i) = \sum_{k=-\infty}^{\infty} h(k) f(i-k).$$

Observe that the preceding formula only sums a number of finite terms if either h or f has finite length. It is usually assumed that a filter h has finite length and a signal f is the periodic extension of a finite signal. Suppose that h is a finite filter, then there is a positive integer M such that h(j)=0 for all $|j| > M$. We use the shorthand notation $h = (h(-M), ..., h(-2), h(-1), h(0), h(1), h(2), .., h(M))$ in this case (ignoring terms in the sequence h which are 0). Let $f = (f(0), f(1), f(2), ..., f(N-1))$ denote a finite signal and $\tilde{f}$ be its periodic extension, that is,

$$\tilde{f}(i) = f(i), \quad i = 0, 1, 2, ..., N-1,$$

and

$$\tilde{f}(i) = f(j), \quad i = kN + j \text{ for } k = \pm 1, \pm 2, ... \text{ and } j = 0, 1, 2, ..., N-1.$$

Let

$$\begin{cases} \tilde{u} = g \otimes \tilde{f} \\ \tilde{v} = h \otimes \tilde{f}, \end{cases} \tag{6.2.3}$$

where $g = \frac{1}{2}(1, 0, 1)$ and $h = \frac{1}{2}(-1, 0, 1)$, and

$$\tilde{f} = g^* \otimes \tilde{u} + h^* \otimes \tilde{v}. \tag{6.2.4}$$

where $g^* = \frac{1}{2}(1, 0, 1)$ and $h^* = \frac{1}{2}(1, 0, -1)$. Then , it can be verified that for u and v in Eq. (6.2.1),

$$u(i) = \tilde{u}(i), \quad i = 0, 1, ..., N-1$$

$$v(i) = \tilde{v}(i), \quad i = 0, 1, ..., N-1$$

and for f in Eq. (6.2.2),

$$f(i) = \tilde{f}(i), \quad i = 0, 1, ..., N-1.$$

Similarly, the linear convolution of 2D sequences $f(i,j), i,j = ..., -2, -1, 0, 1, 2, ...$, and and $h(i,j), i,j = ..., -2, -1, 0, 1, 2, ...$, is defined by

$$h \otimes f(i,j) = \sum_{k=-\infty}^{\infty} \sum_{\ell=-\infty}^{\infty} h(i-k, j-\ell) f(k, \ell).$$

Now we are ready to define two-dimensional symmetric Haar transform. As in the 1D case, we assume that a 2D filter is finite, that is, it has finite number of nonzero terms and that a 2D signal (image) is the periodic extension of a 2D finite signal. Let $h_x = (h_x(i,j)), i,j = -1, 0, 1$, be a 3×3 matrix defined by

$$\begin{bmatrix} 1/4 & 0 & -1/4 \\ 0 & 0 & 0 \\ 1/4 & 0 & -1/4 \end{bmatrix}$$

$h_y = (h_y(i,j)), i,j = -1, 0, 1$, be a 3×3 matrix defined by

$$\begin{bmatrix} 1/4 & 0 & 1/4 \\ 0 & 0 & 0 \\ -1/4 & 0 & -1/4 \end{bmatrix}.$$

Let g be a $N \times N$ matrix with nonnegative entries which are sum to 1, where N is a positive integer. Then, h_x is the shorthand notation of the 2D finite filter $\tilde{h}_x(i,j), i,j = -\infty, ... -1, 0, 1, ..., \infty$, where $\tilde{h}_x(i,j) = h_x(i,j)$ if $|i| \leq 1$ and $|j| \leq 1$; $\tilde{h}_x(i,j) = 0$ otherwise. It is the same for h_y and g. Let f denote a 2D sequence (image), which is periodically extended from a finite 2D sequence (image). The 2D symmetric Haar transform of f is defined by

$$(u_x, u_y, v) = (h_x \otimes f, h_y \otimes f, g \otimes f)$$

where $h_x \otimes f$ is called the highpass component in the x direction , $h_y \otimes f$ is called the highpass component in the y direction and and $g \otimes f$ is called the smooth component (We will explore $g \otimes f$ more in Section 6.4).

For the one-dimensional case, it is easy to reconstruct the original signal from the lowpass and highpass components of the symmetric Haar transforms. But for the 2D case, the reconstruction is not so obvious. To reconstruct the original image from its symmetric Haar transform, we need to find matrices h_x^*, h_y^* and g^* such that

$$f = h_x^* \otimes u_x + h_y^* \otimes u_y + g^* \otimes v.$$

This implies that the matrices h_x, h_y, g, h_x^*, h_y^* and g^* must satisfy

$$h_x^* \otimes h_x(i,j) + h_y^* \otimes h_y(i,j) + g^* \otimes g(i,j) = \delta(i,j)$$

where $\delta(i,j)$ is equal to 1 if i=j=0 and equal to 0 otherwise. Suppose that both h_x^* and h_y^* are 3×3 matrices. Then, both $h_x^* \otimes h_x$ and $h_y^* \otimes h_y$ are 5×5 matrices. This implies that $g^* \otimes g$ is also a 5×5 matrix. One could try different matrices to see if the preceding equation is satisfied but it would be very tedious to do so. It becomes easier if we use some results in signal processing theory. The Fourier transform of a 2D sequence $x(j_1, j_2)$, where $j_1, j_2 = ..., -2, -1, 0, 1, 2, ...,$ is defined by

$$\hat{x}(\omega_1, \omega_2) = \sum_{j_1=-\infty}^{\infty} \sum_{j_2=-\infty}^{\infty} x(j_1, j_2) e^{-i\omega_1 j_1} e^{-i\omega_2 j_2}. \tag{6.2.5}$$

It is easy to verify that the Fourier transform of the linear convolution of two sequences is equal to the product of their Fourier transforms. Thus, the problem becomes to find Fourier transforms $\hat{h}_x$, $\hat{h}_y$, $\hat{g}$, $\hat{h}_x^*$, $\hat{h}_y^*$ and $\hat{g}*$ such that

$$\hat{h}_x^* \hat{h}_x + \hat{h}_y^* \hat{h}_y + \hat{g}^* \hat{g} = 1. \tag{6.2.6}$$

Next, we give an example which satisfies Eq. (6.2.6). From the definition of W_x and W_y, we know that

$$\hat{h}_x = \frac{1}{4}(e^{-i\omega_1 - i\omega_2} + e^{-i\omega_1 + i\omega_2} - e^{i\omega_1 + i\omega_2} - e^{i\omega_1 - i\omega_2}) = -i\cos\omega_2 \sin\omega_1$$

and

$$\hat{h}_y = \frac{1}{4}(e^{-i\omega_1 - i\omega_2} - e^{-i\omega_1 + i\omega_2} - e^{i\omega_1 + i\omega_2} + e^{i\omega_1 - i\omega_2}) = -i\cos\omega_1 \sin\omega_2.$$

If we choose h_x^* as

$$\begin{bmatrix} -1/4 & 0 & 1/4 \\ 0 & 0 & 0 \\ -1/4 & 0 & 1/4 \end{bmatrix}$$

and h_y^* as

$$\begin{bmatrix} -1/4 & 0 & -1/4 \\ 0 & 0 & 0 \\ 1/4 & 0 & 1/4 \end{bmatrix},$$

then $\hat{h}_x^* = i\cos\omega_2 \sin\omega_1$ and $\hat{h}_y^* = i\cos\omega_1 \sin\omega_2$. The equation (6.2.6) becomes

$$\cos^2\omega_1 \sin^2\omega_2 + \cos^2\omega_1 \sin^2\omega_2 + \hat{g}^*\hat{g} = 1.$$

That is, $\hat{g}^*$ and $\hat{g}$ must satisfy:

$$\hat{g}^*\hat{g} = 1 - \cos^2\omega_1 \sin^2\omega_2 + \cos^2\omega_1 \sin^2\omega_2$$

$$= \cos^2\omega_1 \cos^2\omega_2 + \sin^2\omega_1 \sin^2\omega_2. \qquad (6.2.7)$$

One trivial solution of Eq. (6.2.7) is

$$\hat{g} = \cos^2\omega_1 \cos^2\omega_2 + \sin^2\omega_1 \sin^2\omega_2$$

and

$$\hat{g}^* = 1.$$

Write

$$\hat{g} = \cos^2\omega_1\cos^2\omega_2 + \sin^2\omega_1\sin^2\omega_2$$

$$= (\frac{e^{i\omega_1} + e^{-i\omega_1}}{2})^2(\frac{e^{i\omega_2} + e^{-i\omega_2}}{2})^2 + (\frac{e^{i\omega_1} - e^{-i\omega_1}}{2i})^2(\frac{e^{i\omega_2} + e^{-i\omega_2}}{2i})^2$$

$$= (\frac{2 + e^{2i\omega_1} + e^{-2i\omega_1}}{4})(\frac{2 + e^{2i\omega_2} + e^{-2i\omega_2}}{4})$$

$$+(\frac{-2 + e^{2i\omega_1} + e^{-2i\omega_1}}{-4})(\frac{-2 + e^{2i\omega_2} + e^{-2i\omega_2}}{-4})$$

$$= \frac{1}{2} + \frac{1}{8}(e^{-2i(\omega_1+\omega_2)} + e^{-2i(\omega_1-\omega_2)} + e^{2i(\omega_1-\omega_2)} + e^{2i(\omega_1+\omega_2)})$$

For

$$g = \begin{bmatrix} 1/8 & 0 & 0 & 0 & 1/8 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1/8 & 0 & 0 & 0 & 1/8 \end{bmatrix}, \tag{6.2.8}$$

since

$$\hat{g}(\omega_1, \omega_2) = \sum_{j_1=-2}^{2}\sum_{j_2=-2}^{2} g(j_1, j_2)e^{-i\omega_1 j_1}e^{-i\omega_2 j_2}$$

$$= \frac{1}{2} + \frac{1}{8}(e^{-2i(\omega_1+\omega_2)} + e^{-2i(\omega_1-\omega_2)} + e^{2i(\omega_1-\omega_2)} + e^{2i(\omega_1+\omega_2)}),$$

the g in (6.2.8) is the desired filter. Similarly, we obtain

$$g^* = \delta(i, j).$$

6.2.2 Wavelet Modulus and Phase. Similar to the one-dimensional case, we define two-dimensional symmetric Haar wavelet transform by applying the symmetric Haar transform to each smooth part. Formally, let

$$g^0 = 1$$

$$g^j = g^{j-1} \otimes g, j > 0.$$

The symmetric Haar wavelet transform of a two-dimensional signal f at scale j is defined by

$$(W_x^j f, W_y^j f, S^j f) = (h_x \otimes (g^{j-1} \otimes f), h_y \otimes (g^{j-1} \otimes f), g^j \otimes f).$$

For each pixel (m,n), the modulus of the two-dimensional symmetric Haar wavelet transform of f at scale j is defined by

$$M^j f(m,n) = \sqrt{(W_x^j f(m,n))^2 + (W_y^j f(m,n))^2}$$

and the phase of the two-dimensional symmetric Haar wavelet transform of f at scale j is

$$P^j f(m,n) = \mathrm{Arctan} \frac{W_y^j f(m,n)}{W_x^j f(m,n)} =$$

$$\begin{cases} \arctan \frac{W_y^j f(m,n)}{W_x^j f(m,n)} & W_x^j f(m,n) \geq 0, W_y^j f(m,n) \geq 0 \\ \arctan \frac{W_y^j f(m,n)}{W_x^j f(m,n)} + \pi & W_x^j f(m,n) < 0 \\ \arctan \frac{W_y^j f(m,n)}{W_x^j f(m,n)} + 2\pi & W_x^j f(m,n) \geq 0, W_y^j f(m,n) < 0. \end{cases}$$

For convenience, we call the $M^j f(m,n)$'s and $P^j f(m,n)$'s wavelet moduli and wavelet phases respectively in the discussion below. The modulus of the 2D wavelet transform is a straightforward extension of that in the 1D case. The wavelet modulus maxima based multiscale edge detection and signal restoration techniques discussed in the previous chapter can be extended to the 2D case in a straightforward manner. The next subsection will summarize these techniques for the 2D case. But the wavelet phase is a characteristic of 2D wavelet transforms which has no counterpart in the 1D case. We will discuss it throughout this chapter.

6.2.3 Wavelet Modulus Based Image Restoration. In this subsection, we extend the main results obtained in the previous chapter to the two-dimensional case.

In the previous chapter, edge locations in a 1D signal is detected by finding local modulus maxima of wavelet transforms. In practice, the local modulus maxima is determined by comparing the value of the wavelet transform of the signal with the values of its left and right neighboring locations. This technique can be extended to the 2D case in a straightforward manner. For a pixel (i,j) in the 2D case, however, the comparison of moduli of wavelet transforms must be taken for the pixels $(i+k, j+\ell)$ where k, ℓ =-1, 0, 1 except k=ℓ=0. The detailed multiscale edge detection algorithms are given in Section 6.5.1.

For the 1D case, we saw in the previous chapter that the original signal can be approximated by interpolating the values of wavelet transforms at edge locations. This is also true for the 2D case. However, the interpolation is taken for both x component and y component of the wavelet transforms in the x and y directions respectively. See Section 6.5.4 for the details of the interpolation algorithm of the 2D case.

For the 1D case, noisy edges can be discriminated by thresholding the moduli of wavelet transforms since it was seen in the previous chapter that the mean square of the wavelet transform of *i.i.d.* noise at scale j is bounded by $O(2^{-j}\sigma^2)$ where σ^2 is the variance of the noise and that the mean square is usually smaller than that of a true edge. Now, we look at the 2D case. As in the 1D case, we can define symmetric Haar wavelet transforms for random signals. Let $X = \{X(i,j), i,j = ..., -2, -1, 0, 1, 2, ...\}$ denote a 2D random signal. The symmetric Haar wavelet transform of X at scale k is defined by

$$(h_x \otimes (g^{j-1} \otimes X), h_y \otimes (g^{j-1} \otimes X), g^j \otimes X)$$

where $\otimes$ denotes linear convolution.

Next, we consider wavelet transform of noise. Let $\{\epsilon(i,j)\}$'s, i,j=...,-2,-1,0, 1, 2,..., denote *i.i.d.* random variables with mean 0 and variance σ^2. $\epsilon = \{\epsilon(i,j)\}$ is called an *i.i.d.* noise. For the 2D *i.i.d.* noise, we observe that the variance (mean square) of the moduli of of the symmetric Haar wavelet transform coefficients at scale k is $O(2^{-2k+1}\sigma^2)$. The argument for this claim is similar to that in Section 5.5 for the 1D case. We give it as follows:

Since $W_x^k\epsilon(i,j) = h_x \otimes g^{k-1} \otimes \epsilon(i,j)$ contains at most 4^k terms of $\epsilon(k,l)$'s but has a multiplier 4^{-k} for each term and since each $\epsilon(k,l)$ is independent to each other and has variance σ^2, the mean square of $W_x^k\epsilon(i,j)$

$$E|W_x^k\epsilon(i,j)|^2 = E|h_x \otimes g^{k-1} \otimes (\epsilon(i,j))|^2 \leq 4^{-k}\sigma^2.$$

Similarly, the mean square of $W_y^k\epsilon(i,j)$

$$E|W_y^k\epsilon(i,j)|^2 \leq 4^{-k}\sigma^2.$$

Therefore, the mean square of the wavelet modulus $M^k\epsilon(i,j)$

$$\begin{aligned} EM^k\epsilon(i,j)^2 &= E(|W_x^k\epsilon(i,j)|^2 + |W_y^k\epsilon(i,j)|^2) \\ &\leq 4^{-k}\sigma^2 + 4^{-k}\sigma^2 \\ &= 2^{-2k+1}\sigma^2. \end{aligned}$$

That is, the mean square of the modulus of the symmetric Haar wavelet transform of $\epsilon(i,j)$ is $O(2^{-2k+1}\sigma^2)$.

In summary, we conclude that the mean square of the modulus of the wavelet transform of two-dimensional *i.i.d.* noise decays four times as scale increases by one. This observation can be exploited to discriminate noisy edges from true edges when σ^2 is relatively small, compared with the square of the wavelet modulus of a true edge. The denoising algorithm based on this observation is given in Section 6.5.2.

6.2.4 Limitations of Modulus based Denoising Schemes. Mallat *et al* [MALL92a,b] successfully restored noisy images with sharp edges using the modulus based denoising scheme. The algorithm can also be used to remove Poisson-type noise in the Shepp-Logan phantom itself, where the edges are sharply defined. See Figure 3 and 6 and the discussions below. But in our numerical experiments, the same algorithm did not perform well for the denoising of PET raysum images. Compare Figure 6 and Figure 14. In this subsection, we compare the Shepp-Logan phantom image and its raysum image and give tentative explanation of why the modulus based algorithm does not perform well for the raysum image.

Figure 3, known as the Shepp-Logan phantom, is a simplified image which simulates radioisotope densities in the human brain. This phantom is often chosen as a standard to test the accuracy of a restoration algorithm and the results reported here all use it. The image is discretized by 128×128 pixels. Using Mallat's algorithm of multiscale edge representation, it is possible to reconstruct the original image from the multiscale edges with high precision for such an image, see Figure 4.

From the discussions in Chapter 5 and Section 6.2.1 , we know that the original image can be reconstructed from its wavelet transforms. Since human observation is sensitive to edges, it presumably preserves most features contained in the images by saving wavelet transforms at edge locations and by interpolating these wavelet transforms to obtain wavelet transforms at other locations. If edges in images are sharper, the wavelet moduli of the images at the edge pixels are larger so that the edges can be more likely to be distinguished by comparing and thresholding wavelet moduli. As discussed in Section 5.5, if such images with sharp edges are contaminated by noise with relatively small variance, true edges will be more likely to be saved and noisy edges will be more likely to be removed by thresholding wavelet moduli. This might partially answer the question of why the modulus based reconstruction and denoising schemes work better for the images with sharp edges.

Figure 7 is a typical raysum image of the Shepp-Logan phantom. The raysum data is discretized by 180 angles of view (displayed by row) and 128 segments for each angle of view (displayed by column). (For convenience of computer display, we artificially cut off the lower part of the raysum image in Figure 7.) We can see from Figure 7 that the edges of the image are usually not so sharp as in the original Shepp-Logan phantom, and the intensity of the image changes slowly and subtly, rather than abruptly as in the original Shepp-Logan phantom. The analysis for magnitude based edge detection methods in Chapter IV and the analysis for multiscale edge representation in Chapter V suggests that a restoration scheme based on moduli alone cannot be expected to work well for the denoising of the raysum image since the modulus based method is sensitive to the variance of noise, which is typically high in the raysum image. This motivates us to study the phase of the wavelet transform, which does not have a one-dimensional counterpart, and develop a denoising algorithm which incorporates phase information.

6.3 Distribution of Wavelet Phase of Noise

In this section, we state and prove the main theorem of this thesis. The theorem states that the distribution of the phase of the wavelet transform of Poisson-type noise, which usually occurs in PET process, is asymptotically uniformly distributed on $[0, 2\pi)$. The theorem will be used as the basis for our denoising algorithm. First, we review some probability theory which will be used later in this section.

Definition 6.3.1 (Four Types of Convergence): Let $\{X, X_n, n \geq 1\}$ be a family of random variables on a probability space (Ω, Σ, P).

(a) X_n is said to converge to X almost everywhere if $P(\lim_{n\to+\infty} X_n = X) = 1$; notation: $X_n \to X$ a.e.;

(b) X_n is said to converge to X in probability if for any $\epsilon > 0$, $\lim_{n\to+\infty} P(|X_n - X| > \epsilon) = 0$; notation: $X_n \to X$ in p.;

(c) X_n is said to converge to X in distribution if $F_{X_n}(x)$ converges to $F_X(x)$ at all points where $F_X(x)$ is continuous; notation: $X_n \to X$ in d.;

(d) X_n is said to converge to X in L_p-norm if $\lim_{n\to+\infty} E(|X_n - X|^p) = 0$; notation: $X_n \to X$ in L_p.

In this section, we are mainly concerned with the convergence in distribution. The following theorem is well known:

Theorem 6.3.1 (Central Limit Theorem): Let $X_n, n \geq 1$, be independent identically distributed (iid) random variables with mean μ and variance σ^2. Then

$$\frac{\sum_{i=1}^{n}(X_i - \mu)}{\sqrt{n}\sigma}$$

converges in distribution to a standard Gaussian random variable.

Proof: See [CHUN74] pp.169-170.

The next three propositions exploit some algebraic and trigonometric properties of Gaussian random variables. Our later discussion will use these propositions.

Proposition 6.3.1: Let X and Y be independent Gaussian random variables with mean zero and variance σ^2. Then $\Theta = \text{Arctan}(\frac{Y}{X})$ is uniformly distributed on $[0,2\pi)$.

Proof: Let $r = \sqrt{x^2 + y^2}$ and $R = \sqrt{X^2 + Y^2}$. Then

$$\begin{cases} x = r\cos\theta, & \theta \in [0, 2\pi) \\ y = r\sin\theta, & \theta \in [0, 2\pi) \end{cases}$$

establishes a one-to-one mapping between (r, θ) plane to (x,y) plane with the origin removed in each case. Let $f_{R\Theta}(r,\theta)$ and $f_{XY}(x,y)$ be the joint density functions of (R,Θ) and (X,Y) respectively. According to the Jacobian transformation theorem,

$$f_{R\Theta}(r,\theta) = f_{XY}(x,y)J(\frac{x,y}{r,\theta})$$

where

$$J(\frac{x,y}{r,\theta}) = |det \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \theta} \end{bmatrix} | = |r| = r.$$

Therefore,

$$f_{R\Theta}(r,\theta) = \frac{r}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$
$$= \frac{r}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}}$$

and

$$f_{\Theta}(\theta) = \int_0^{+\infty} f_{R\Theta}(r,\theta)dr = \frac{1}{2\pi}.$$

This means that θ is uniformly distributed on $[0,2\pi)$.

QED

Proposition 6.3.2: Let X be Y are two independent identically distributed random variables. Then X+Y and X-Y are independent if and only if both X and Y are Gaussian distributed.

Proof: The proof of "if" part is straightforward, note that both X+Y and X-Y are Gaussian distributed if X and Y are Gaussian and that the Gaussian random variables X and Y are independent if and only if E(XY)=E(X)E(Y). For the proof of "if only" part, see [CHUN74] pp.171-172.

QED

Proposition 6.3.3: Let X_i and Y_i, i=1,2, be Gaussian distributed random variables. Suppose X_i and Y_j are independent for i,j=1, 2. Then $X_1 + X_2$ and $Y_1 + Y_2$ are independent.

Proof : The proof is straightforward, note that for the case of Gaussian distribution, X and Y are independent if and only if E(XY)=E(X)E(Y).

QED

Next, we apply above basic properties of Gaussian distribution to analyze the phase of the symmetric Haar wavelet transform defined in Section 6.2.2 for independent Gaussian noise.

Let **L**={(i,j): i,j are integers} be a two-dimensional lattice. A discrete image is modeled as a collection of random variables indexed in the lattice **L**, denoted by

X(i,j): (i,j) $\in$ **L**. Sometimes, (i,j) is referred to as a pixel or node. From the statistical point of view, the problem of image restoration is to estimate the mean $\mu(i,j)$ of X(i,j) from the observation x(i,j) under some criterion. One of the commonly used criteria is to minimize the mean square error, *i.e.*, to minimize $\sum_i \sum_j \epsilon(i,j)^2$, where $\epsilon(i,j) = X(i,j) - \mu(i,j)$ is referred to as noise. $\epsilon(i,j)$ is called Gaussian noise if X(i,j) is Gaussian distributed, and Poisson (or Poisson-type) noise if X(i,j) is Poisson distributed. The following theorem states the distribution of the symmetric Haar wavelet transform of Gaussian noise.

Theorem 6.3.2: Suppose that $\{\epsilon$(i,j): (i,j)$\in$ **L**$\}$ are independent Gaussian noises. Let $W_x\epsilon(i,j)$ and $W_y\epsilon(i,j)$ denote the components of the 2D symmetric Haar wavelet transform of $\epsilon(i,j)$. Then $\theta(i,j) = \text{Arctan}\frac{W_y\epsilon(i,j)}{W_x\epsilon(i,j)}$ is uniformly distributed on $[0, 2\pi)$ for each (i,j).

Proof: We prove the theorem for the case of scale s=1. The argument for other scales is similar. It is easy to verify that

$$W_x\epsilon(i,j) = \frac{1}{4}(\epsilon(i-1,j-1) - \epsilon(i+1,j-1) + \epsilon(i-1,j+1) - \epsilon(i+1,j+1))$$

and

$$W_y\epsilon(i,j) = \frac{1}{4}(\epsilon(i-1,j-1) - \epsilon(i-1,j+1) + \epsilon(i+1,j-1) - \epsilon(i+1,j+1)).$$

Let

$$U(i,j) = \frac{1}{4}(\epsilon(i-1,j-1) - \epsilon(i+1,j+1))$$

and

$$V(i,j) = \frac{1}{4}(\epsilon(i+1,j-1) - \epsilon(i-1,j+1)).$$

Then, it follows that $W_x\epsilon(i,j) = U(i,j) - V(i,j)$ and $W_y\epsilon(i,j) = U(i,j) + V(i,j)$. Since $\{\epsilon(i,j)\}$'s are independent Gaussian for $i \neq j$, by Proposition 6.3.3, U(i,j) and V(i,j) are independent Gaussian random variables. Using Proposition 6.3.2, it follows that $W_x\epsilon(i,j)$ and $W_y\epsilon(i,j)$ are independent. Since $\{\epsilon(i,j)\}$'s are Gaussian noises with zero mean, it is easy to verify that both $W_x\epsilon(i,j)$ and $W_y\epsilon(i,j)$ are

Gaussian distributed with mean 0 and the same variance. By Proposition 6.3.1, Arctan$\frac{W_y\epsilon(i,j)}{W_x\epsilon(i,j)}$ is uniformly distributed on $[0,2\pi)$.

QED

In the PET image reconstruction application, the noise is not Gaussian but of Poisson type. Using the model in Chapter II, the numbers of photons at detector pairs constitute independent Poisson random variables. Let Δt denote a fixed length of time and let $N_n(i,j)$ denote the number of photons detected at pair (i,j) during time $[n\Delta t, (n+1)\Delta t), n = 0, 1, 2, ...$ Then (ignoring the finite halflife of the radioisotopes), $N_n(i,j)$'s are independent Poisson distributed random variables with the same mean. Let $\lambda(i,j)$ denote the mean. Let $T_n = n\Delta t$ and let $N_{T_n}(i,j)$ denote the number of photons detected at detector pair (i,j) in the time interval $[0, T_n)$. Then, $N_{T_n}(i,j)$ is a Poisson distributed random variable with mean $n\lambda(i,j)$ since it is a sum of n independent Poisson distributed random variables. Letting $T_n \to \infty$, *i.e.*, $n \to \infty$, according to the classic central limit theorem, $\frac{N_{T_n}(i,j)-n\lambda(i,j)}{\sqrt{n\lambda(i,j)}}$ converges in distribution to a standard Gaussian random variable. Let $\epsilon_n(i,j) = N_{T_n}(i,j) - n\lambda(i,j)$, then $\epsilon_n(i,j)$ is a random variable with mean zero and variance $n\lambda(i,j)$ and $\frac{\epsilon_n(i,j)-n\lambda(i,j)}{\sqrt{n\lambda(i,j)}}$ converges in distribution to a standard Gaussian random variable as $n \to \infty$. $\epsilon_n(i,j)$ is referred to as Poisson noise.

Definition 6.3.1: The Poisson noise $\epsilon_n(i,j)$ is said to be regular if for disjoint (i_1, j_1) and (i_2, j_2), $\frac{\epsilon_n(i_1,j_1)}{\sqrt{n\lambda(i_1,j_1)}}$ and $\frac{\epsilon_n(i_2,j_2)}{\sqrt{n\lambda(i_2,j_2)}}$ converge in distribution to two independent standard Gaussian random variables as $n \to \infty$.

For notational convenience, we denote $X_n = W_x\epsilon_n(i,j)$ and $Y_n = W_y\epsilon_n(i,j)$. For such 2D wavelet transforms of regular Poisson noise, we have the following main theorem:

Theorem 6.3.3: Let $n\Delta t$ be the length of observation. Let (X_n, Y_n) be the 2D symmetric Haar wavelet transform of the regular Poisson noise observed during $[0, n\Delta t)$. Then Arctan$\frac{Y_n}{X_n}$ converges in distribution to a random variable which is uniformly distributed on $[0, 2\pi)$ as $n \to \infty$.

To prove this theorem, we need to prove the following lemmas first.

Lemma 6.3.1: X_n converges to X in distribution if and only if $\phi_n(t)$ converges to ϕ(t) in every finite interval, where ϕ_n, ϕ denote the characteristic functions of X_n, X respectively.

Proof: See [CHUN74] pp. 160-162.

Lemma 6.3.2: If X_n converges to X in distribution, then for a strictly increasing continuous function h(x), h(X_n) converges to h(X) in distribution.

Proof: X_n converges to X in distribution means for any point x of continuity of F, $F_{X_n}(x) \to F_X(x)$. We need to show that for any point x of continuity of $F_{h(X)}$, $F_{h(X_n)}(x) \to F_{h(X)}(x)$. Since h(x) is strictly increasing and continuous, the inverse function $h^{-1}(x)$ exists and is also continuous. Since $F_X(h^{-1}(x)) = P(X \leq h^{-1}(x)) = P(h(X) \leq x) = F_{h(X)}(x)$, $h^{-1}(x)$ is a point of continuity of F_X if and only if x is a point of continuity of $F_{h(X)}$. Therefore, if x is a point of continuity of F_{X_n},

$$F_{h(X_n)}(x) = P(h(X_n) \leq x) = P(X_n \leq h^{-1}(x)) = F_{X_n}(h^{-1}(x))$$

converges to

$$F_X(h^{-1}(x)) = F_{h(X)}(x).$$

That is, $h(X_n)$ converges to h(X) in distribution. *QED*∎

Before going to next lemma, we state a proposition about convergence in distribution. The proof for this proposition can be found in page 31 of [BILL68]. The proposition will be used in the proof of Lemma 6.3.3 and Lemma 6.3.5 below.

Proposition 6.3.4: Suppose that h is a real, bounded, measurable function. Assume that the measure of the set of the discontinuity points of h is zero. Then, if (X_n, Y_n) converges to (X, Y) in distribution, necessarily

$$\int h dF_n \to \int h dF,$$

where F_n and F are the distribution function of (X_n, Y_n) and (X,Y) respectively.

Lemma 6.3.3:

(i) Suppose X_n and Y_n, $n \geq 1$, are independent, and X and Y are independent. If $X_n \to X$ in d., $Y_n \to Y$ in d., then $X_n + Y_n \to X + Y$ in d..

(ii) If, in addition the conditions in (i), we assume that X and Y are continuous random variables, then $\frac{Y_n}{X_n} \to \frac{Y}{X}$ in d..

Proof:

(i) By the assumption of independence, we have

$$\phi_{X_n+Y_n}(t) = \phi_{X_n}(t)\phi_{Y_n}(t)$$

and

$$\phi_{X+Y}(t) = \phi_X(t)\phi_Y(t).$$

From Lemma 6.3.1, it follows that $X_n + Y_n$ converges to X+Y in d..

(ii) Note

$$F_{\frac{Y_n}{X_n}}(x) = P(\frac{Y_n}{X_n} \leq x)$$

$$= P(Y_n \leq xX_n,\ X_n > 0) + P(Y_n \geq xX_n,\ X_n < 0) + P(X_n = 0, Y_n < 0).$$

Since $X_n \to X$ in d., $P(X_n = 0, Y_n < 0) = E_{I_S}(X_n, Y_n) = \int I_S dF_{(X_n,Y_n)}$, where $S = \{(u,v) : u = 0, v < 0\}$ and I_S is the indicator function of S. Since I_S is a function in R^2 satisfying the condition of Proposition 6.3.4, $E_{I_S}(X_n, Y_n)$ converges to $E_{I_S}(X, Y)$. That is, $P(X_n = 0, Y_n < 0)$ converges to $P(X = 0, Y < 0)$. Next, we show that $P(Y_n \leq xX_n, X_n > 0)$ converges to $P(Y \leq xX, X > 0)$. Assuming that $X_n \to X$ in d., $Y_n \to Y$ in d., X_n is independent of Y_n and X is independent Y, it follows that the joint distribution function of X_n and Y_n converges pointwise to the joint distribution function of X and Y at each point (x,y) at which the latter is continuous. Note $P(Y_n \leq xX_n, X_n > 0) = E_{I_A}(X_n, Y_n) = \int I_A dF_{X_n,Y_n}$, where $A = \{(u,v) : v - xu \leq 0, u > 0\}$ and I_A is the indicator function of A. Since I_A is a function in R^2 satisfying the conditions of Proposition 6.3.4, $E_{I_A}(X_n, Y_n)$ converges to $E_{I_A}(X, Y)$. That is, $P(Y_n \leq xX_n, X_n > 0)$ converges to $P(Y \leq xX, X > 0)$. The proof that $P(Y_n \geq xX_n, X_n < 0)$ converges to $P(Y \geq xX, X < 0)$ is identical. Thus, (ii) is true.

QED

Lemma 6.3.4: Let $n\Delta t$ be the length of observation. Let (X_n, Y_n) be the 2D symmetric Haar wavelet transform of the regular Poisson noise observed during $[0, n\Delta t)$. Then, there exist two independent random variables U_n and V_n such that

(a) $\frac{U_n}{\sqrt{n}}$ and $\frac{V_n}{\sqrt{n}}$ converge in distribution to independent Gaussian random variables U and V as $n \to \infty$.

(b) $\frac{Y_n}{X_n} = \frac{U_n+V_n}{U_n-V_n}$.

Proof: Similar to the arguments in the proof of Theorem 6.3.2, we have

$$W_x\epsilon_n(i,j) = \frac{1}{4}(\epsilon_n(i-1,j-1) - \epsilon_n(i+1,j-1) + \epsilon_n(i-1,j+1) - \epsilon_n(i+1,j+1))$$

and

$$W_y\epsilon_n(i,j) = \frac{1}{4}(\epsilon_n(i-1,j-1) - \epsilon_n(i-1,j+1) + \epsilon_n(i+1,j-1) - \epsilon_n(i+1,j+1)).$$

Let

$$U_n(i,j) = \frac{1}{4}(\epsilon_n(i-1,j-1) - \epsilon_n(i+1,j+1))$$

and

$$V_n(i,j) = \frac{1}{4}(\epsilon_n(i+1,j-1) - \epsilon_n(i-1,j+1)).$$

Then $W_x\epsilon_n(i,j) = U_n(i,j) - V_n(i,j)$ and $W_y\epsilon_n(i,j) = U_n(i,j) + V_n(i,j)$ follows. Since the noises at different pixels are independent, U_n and V_n are independent.

Next, we show that both $\frac{U_n}{\sqrt{n}}$ and $\frac{V_n}{\sqrt{n}}$ converge to some Gaussian random variables U and V in distribution. First, we consider the case of $\frac{U_n}{\sqrt{n}}$. As mentioned above, $\epsilon_n(i-1,j-1)$ is assumed to be the sum of n independent random variables with mean 0 and variance $\lambda(i-1,j-1)$. By the classical central limit theorem, it follows that

$$\frac{\epsilon_n(i-1,j-1)}{\sqrt{n\lambda(i-1,j-1)}}$$

converges to a standard Gaussian random variable U_1. By the same argument,

$$\frac{\epsilon_n(i+1,j+1)}{\sqrt{n\lambda(i+1,j+1)}}$$

converges to a standard Gaussian random variable U_2. Since the noise is assumed to be regular, U_1 and U_2 are independent. Since $\epsilon_n(i-1,j-1)$ and $\epsilon_n(i+1,j+1))$ are independent, by (i) in Lemma 6.3.3, it follows that

$$\frac{U_n(i,j)}{\sqrt{n}} = \frac{1}{4}(\sqrt{\lambda(i-1,j-1)}\frac{\epsilon_n(i-1,j-1)}{\sqrt{n\lambda(i-1,j-1)}}$$

$$-\sqrt{\lambda(i+1,j+1)}\frac{\epsilon_n(i+1,j+1)}{\sqrt{n\lambda(i+1,j+1)}})$$

converges in distribution to a Gaussian random variable U with mean zero and variance $\frac{1}{16}(\lambda(i-1,j-1)+\lambda(i+1,j+1))$. By the same argument, $\frac{V_n}{\sqrt{n}}$ converges in distribution to a Gaussian random variable V.

The remaining is to show that U and V are independent. Since U_i, V_i, i=1,2, are independent Gaussian random variables, for constants c_i, d_i, i=1,2, $X_i = c_iU_i$ and $Y_i = d_iV_i$ are also independent Gaussian random variables. Since $U = c_1U_1 + c_2U_2 = X_1 + X_2$ and $V = d_1V_1 + d_2V_2 = Y_1 + Y_2$ by Proposition 6.3.3, it follows that U and V are independent. The claim is true.

QED

Lemma 6.3.5: Suppose $Y_n = U_n + V_n$, $X_n = U_n - V_n$ where U_n and V_n are independent. Let U and V be two independent continuous random variables. If $U_n \to U$ in d. and $V_n \to V$ in d., then $\frac{Y_n}{X_n} \to \frac{U+V}{U-V}$ in d..

Proof: The proof is similar to that of Lemma 6.3.3 (ii). Note

$$F_{\frac{Y_n}{X_n}}(x) = P(\frac{Y_n}{X_n} \leq x)$$

$$= P(Y_n \le xX_n,\ X_n > 0) + P(Y_n \ge xX_n,\ X_n < 0) + P(X_n = 0, Y_n < 0).$$

$$= P(U_n + V_n \le x(U_n - V_n),\ U_n - V_n > 0)$$

$$+P(U_n + V_n \ge x(U_n - V_n),\ U_n - V_n < 0) \tag{6.3.1}$$

$$+P(U_n - V_n = 0, U_n + V_n < 0).$$

We show

$$P(U_n + V_n \le x(U_n - V_n),\ U_n - V_n > 0)$$

converges to

$$P(U + V \le x(U - V),\ U - V > 0).$$

Assuming $U_n \to U$ in d., $V_n \to V$ in d., since U_n is independent of V_n and U is independent V, it follows that the joint distribution function of U_n and V_n converges to the joint distribution function of U and V. Note

$$P(U_n + V_n \le x(U_n - V_n),\ U_n - V_n > 0) = E_{I_A}(U_n, V_n)$$

where $A = \{(u, v) : u + v \le x(u - v), u - v > 0\}$ and I_A denotes the indicator function of A. Since I_A is a function in R^2 satisfying the conditions of Proposition 6.3.4, it follows that $E_{I_A}(U_n, V_n)$ converges to $E_{I_A}(U, V)$ That is, $P(U_n + V_n \le x(U_n - V_n),\ U_n - V_n > 0)$ converges to $P(U + V \le x(U - V),\ U - V > 0)$. The argument for the convergence of other two terms in the RHS of (6.3.1) is identical. The claim is true.

QED

Proof of Theorem 6.3.3: From Lemma 6.3.4,

$$\frac{Y_n}{X_n} = \frac{U_n + V_n}{U_n - V_n}$$

where U_n and V_n are two independent random variables and both $\frac{U_n}{\sqrt{n}}$ and $\frac{V_n}{\sqrt{n}}$ converge to two independent Gaussian random variables, say, F and G, in distribution. Applying Lemma 6.3.5 for $\frac{U_n}{\sqrt{n}}$ and $\frac{V_n}{\sqrt{n}}$, it follows that

$$\frac{Y_n}{X_n} = \frac{\frac{U_n}{\sqrt{n}} + \frac{V_n}{\sqrt{n}}}{\frac{U_n}{\sqrt{n}} - \frac{V_n}{\sqrt{n}}}$$

converges to $\frac{F+G}{F-G}$. Since both F-G and F+G are Gaussian, it implies from Proposition 6.3.2 that F-G and F+G are independent. Note F-G and F+G have the same variance. Using Proposition 6.3.1 for Y=F+G and X=F-G, it follows that Arctan$\frac{F+G}{F-G}$ is uniformly distributed on $[0, 2\pi)$. The theorem is proved.

QED

In practice, the observation time T_n cannot be infinitely long. Therefore, the distribution of the wavelet phase of the Poisson noise cannot be an exact uniformly distributed random variable on $[0, 2\pi)$. Nevertheless, if the observation time is sufficiently long and if it is assumed that there is no decay of radioactivities, then the number of the detected photons is sufficiently large and the Poisson distribution can be estimated by Gaussian distribution with high accuracy. Thus, the distribution of the wavelet phase is approximately a uniformly distributed random variable on $[0, 2\pi)$.

6.4 Implementation Issue of the 2D Haar Wavelet

In this section, we discuss some implementation issues on the wavelet based filtering scheme.

Let f be an additive noise modeled image given by

$$f(i,j) = u(i,j) + N(i,j) \tag{6.4.1}$$

where u(i,j) is a real-valued number and N(i,j) denotes noise. Recall in Section 6.2 that a symmetric Haar transform of f is defined by

$$(W_x f, W_y f, Sf) = (h_x \otimes f, h_y \otimes f, g \otimes f)$$

where

$$h_x = \begin{bmatrix} 1/4 & 0 & -1/4 \\ 0 & 0 & 0 \\ 1/4 & 0 & -1/4 \end{bmatrix},$$

$$h_y = \begin{bmatrix} 1/4 & 0 & 1/4 \\ 0 & 0 & 0 \\ -1/4 & 0 & -1/4 \end{bmatrix}$$

and

$$g = \begin{bmatrix} 1/8 & 0 & 0 & 0 & 1/8 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1/8 & 0 & 0 & 0 & 1/8 \end{bmatrix}.$$

We also knew that f can be reconstructed by its symmetric Haar transform via

$$f = h_x^* \otimes W_x f + h_y^* \otimes W_y f + g^* \otimes g$$

where

$$h_x^* = \begin{bmatrix} -1/4 & 0 & 1/4 \\ 0 & 0 & 0 \\ -1/4 & 0 & 1/4 \end{bmatrix},$$

$$h_y^* = \begin{bmatrix} -1/4 & 0 & -1/4 \\ 0 & 0 & 0 \\ 1/4 & 0 & 1/4 \end{bmatrix}$$

and

$$g^* = \delta(i, j).$$

Suppose $(\tilde{W}_x f, \tilde{W}_y f)$ is the estimates of $(W_x f, W_y f)$ after employing a wavelet denoising algorithm and the reconstructed image is

$$\tilde{f} = h_x^* \otimes \tilde{W}_x f + h_y^* \otimes \tilde{W}_y f + g^* \otimes Sf.$$

Since

$$u = h_x^* \otimes W_x u + h_y^* \otimes W_y u + g^* \otimes Su$$

where

$$(W_x u, W_y u, Su) = (h_x \otimes u, h_y \otimes u, g \otimes u),$$

$$\tilde{f} - u = h_x^* \otimes (\tilde{W}_x f - W_x u) + h_y^* \otimes (\tilde{W}_y f - W_y u) + g^* \otimes (Sf - Su)$$

$$= h_x^* \otimes (\tilde{W}_x f - W_x u) + h_y^* \otimes (\tilde{W}_y f - W_y u) + g^* \otimes g \otimes N$$

$$= h_x^* \otimes (\tilde{W}_x f - W_x f + W_x N) + h_y^* \otimes (\tilde{W}_y f - W_y f + W_y N) + g^* \otimes SN$$

where

$$(W_x N, W_y N, SN) = (h_x \otimes N, h_y \otimes N, g \otimes N).$$

Therefore, the estimation error

$$||\tilde{f} - u|| \leq ||h_x^* \otimes (\tilde{W}_x f - W_x f + W_x N)|| + ||h_y^* \otimes (\tilde{W}_y f - W_y f + W_y N)|| + ||g^* \otimes SN|| \tag{6.4.2}$$

where $||.||$ denotes a norm. The first two error terms in (6.4.2) depend how much noise has been removed and how accuracy the interpolation is after removing noise but it is difficult to give an analytic estimation for the them. Since the wavelet phase of the Poisson noise is a random variable asymptotically uniformly distributed on $[0, 2\pi)$ no matter how large the variance of the noise is, the wavelet phase based denoising scheme is not sensitive to the variance of noise. Therefore, it is expected to detect more noisy edges than the modulus based scheme for the PET raysum images, which are characterized by noise with high variance; and the noise can be removed even more effectively if both modulus and phase information is used. Thus, the errors in the first two terms of (6.4.2) is expected to be reduced after employing wavelet modulus and phase based denoising schemes. This was also suggested in our numerical experiments but the quantitative error estimates for the first two terms of (6.4.2) are yet to be obtained and remain as a future research topic. However, the error in the third term still remains no matter how good the denoising algorithm is. It is desirable to analyze and if possible to reduce the error in the third term of (6.4.2).

If we use $\tilde{g}$ instead of g as the smoothing operator and replace g^* by $\tilde{g}^*$, then reconstructed image is

$$\tilde{f} = h_x^* \otimes \tilde{W}_x f + h_y^* \otimes \tilde{W}_y f + \tilde{g}^* \otimes \tilde{g} f$$

and the difference between $\tilde{f}$ and u is

$$\tilde{f} - u = h_x^* \otimes (\tilde{W}_x f - W_x u) + h_y^* \otimes (\tilde{W}_y f - W_y u) + (\tilde{g}^* \otimes \tilde{g} - g^* \otimes g)u + \tilde{g}^* \otimes \tilde{g}N.$$

The total error $||\tilde{f} - u||$ will be reduced if we can find such a $\tilde{g}$ and $\tilde{g}^*$ so that

$$||(\tilde{g}^* \otimes \tilde{g} - g^* \otimes g)u + \tilde{g}^* \otimes \tilde{g}N|| < ||g^* \otimes gN||.$$

In our numerical experiments, it was found that for

$$\tilde{g} = \begin{bmatrix} 1/8 & 0 & 1/8 \\ 0 & 1/2 & 0 \\ 1/8 & 0 & 1/8 \end{bmatrix}$$

and

$$\tilde{g}^* = \delta(i, j),$$

the features in the original images have been recovered better than with using g; that is, the errors measured by subjective observation are reduced. Again, we have no analytic error estimate; this is also a topic for future research.

6.5 Filtering Algorithms

This section describes PET image filtering schemes using wavelet phase based filters. Before discussing specific schemes, we summarize the results obtained in the previous sections as follows:

1. An pixel is considered to be an edge pixel if its wavelet transform achieves local maximum in gradient direction;

2. An edge pixel is considered to be generated by noise if the moduli of its wavelet transforms at different scales are relatively small or decay relatively fast as scales increase.

3. Edge pixels generated by Poisson noise have wavelet transforms whose phases are random variables asymptotically uniformly distributed on $[0, 2\pi)$.

6.5.1 Edge Detection using Wavelet Transforms. Let f(x,y) denote an image and $(W_s^1 f(x,y), W_s^2 f(x,y))$ denote its wavelet transform at scale s. Using the same notation as before, the modulus of $(W_s^1 f(x,y), W_s^2 f(x,y))$ is given by

$$Mf(x,y) = \sqrt{(W_s^1 f(x,y))^2 + (W_s^2 f(x,y))^2},$$

and the phase of $(W_s^1 f(x,y), W_s^2 f(x,y))$ is given by

$$Pf(x,y) = \mathrm{Arctan}\frac{W_s^2 f(x,y)}{W_s^1 f(x,y)} =$$

$$\begin{cases} \arctan \frac{W_s^2 f(x,y)}{W_s^1 f(x,y)} & W_s^1 f(x,y) \geq 0, W_s^2 f(x,y) \geq 0 \\ \arctan \frac{W_s^2 f(x,y)}{W_s^1 f(x,y)} + \pi & W_s^1 f(x,y) < 0 \\ \arctan \frac{W_s^2 f(x,y)}{W_s^1 f(x,y)} + 2\pi & W_s^1 f(x,y) \geq 0, W_s^2 f(x,y) < 0 \end{cases} \tag{6.5.1}$$

From Section 5.3, the wavelet phase is in fact the gradient direction of f(x,y) for some suitably chosen wavelet. The point (x,y) is considered to be an edge point if Mf(x,y) achieves local maximum in the gradient direction. Let $f(i,j)$ be the image discretized from f(x,y). We quantize the gradient direction of f(x,y) by 8 possible values 0, $\pi/4$, $\pi/2$, $3\pi/4$, π, $5\pi/4$, $3\pi/2$ and $7\pi/4$, that is, the quantized gradient direction $\theta(i,j)$ of f(i,j) is one of these directions which has value closest to Pf(i,j). The Mf(i,j) is considered to achieve local maximum in the quantized gradient direction $\theta(i,j)$ if it is larger than the modulus of the wavelet transform of the image on one side and not less the modulus on the other side of (i,j) along the quantized gradient direction $\theta(i,j)$. For example, if Pf(i,j)= $\pi/6$ then $\pi/4$ is considered to be the quantized gradient direction, Mf(i,j) is considered to achieve local maximum along the gradient direction if

$$Mf(i,j) > Mf(i-1,j-1) \text{ and } Mf(i,j) \geq Mf(i+1,j+1)$$

or

$$Mf(i,j) \geq Mf(i-1,j-1) \text{ and } Mf(i,j) > Mf(i-1,j-1).$$

Let K be the total number of scales, I be the total number of pixels in each row and J be the total number of pixels in each column. The quasi-code for the multiscale edge detection scheme is given as follows:

Multiscale Edge Detection Algorithm

1. For k=1 to K
2. For i=1 to I
3. For j=1 to J
4. Find wavelet phase Pf(i,j) according to (6.5.1)
5. $\theta(i,j) = \min\{|\theta - Pf(i,j)| : \theta = 0, \pi/4, \pi/2, 3\pi/4, \pi, 5\pi/4, 3\pi/2, 7\pi/4\}$
6. $Mf(i,j) = \sqrt{(W_k^1 f(i,j))^2 + (W_k^2 f(i,j))^2}$
7. Determine if (i,j) is edge by comparing Mf(i,j) in the direction θ
8. Next j
9. Next i
10. Next k

6.5.2 Modulus Maximum based Denoising. An edge pixel is considered to be generated by noise if the modulus of its wavelet transform is relatively small. Several techniques have been used in our study to determine if the modulus is small. First, a threshold on the values of moduli of wavelet transforms of detected edge pixels has been applied to determine if it is small. If the value is less than the threshold value, the pixel is considered to be an edge pixel generated by noise; otherwise, it is considered to be a true edge. In our numerical experiments, we use different threshold values for different scales. Another way to remove noisy edges is to embed thresholding into edge detection process. For the example given in the previous subsection, (i,j) is considered to be an edge pixel if

$$Mf(i,j) > \alpha Mf(i-1,j-1) \text{ and } Mf(i,j) \geq Mf(i+1,j+1)$$

or

$$Mf(i,j) \geq Mf(i-1,j-1) \text{ and } Mf(i,j) > \alpha Mf(i-1,j-1),$$

where α is a number slightly greater than 1. The larger the number α is, the more edges will be removed. From our numerical experiments, a value around 1.1 is selected.

In practice, the modulus maxima based denoising can be embedded into edge detection algorithm. The quasi-code of this denoising scheme is given as following:

Modulus Thresholding Algorithm

1. For k=1 to K
2. For i=1 to I
3. For j=1 to J
4. Find wavelet phase Pf(i,j) according to (6.5.1)
5. $\theta(i,j) = \min\{|\theta - Pf(i,j)| : \theta = 0, \pi/4, \pi/2, 3\pi/4, \pi, 5\pi/4, 3\pi/2, 7\pi/4\}$
6. $Mf(i,j) = \sqrt{(W_k^1 f(i,j))^2 + (W_k^2 f(i,j))^2}$
7. (i,j) is a true edge if

(a) $M(i,j) > M_k$, where M_k is the threshold for scale k.

(b) Mf(i,j) is larger than that on one side of (i,j) along the direction $\theta(i,j)$ and larger than α_k multiplying that of that on the other side along $\theta(i,j)$, where α_k is chosen greater than 1.0.

8. Next j
9. Next i
10. Next k

As observed by Mallat and Hwang [MALL92], another characteristic of noisy edges is that the modulus of the wavelet transform of noise decays faster from scale to scale than does the modulus of the wavelet transform of a "true edge". From this observation, noisy edges can in principle be detected by comparing the moduli of their wavelet transforms at different scales. Due to the fact that the pixel locations of edges may shift from scale to scale, we have to consider the neighboring pixels of the location in the next scale. We describe the algorithm as following:

For scale k and edge pixel (i,j), let $M_k f(i,j)$ denote the modulus of the wavelet transform of f(i,j) at (i,j) and let W_k be a positive integer denoting search window size for scale k. In our numerical experiments, W_k is chosen a value between 1 and 4. The edge pixel (i,j) is considered to be generated by noise if, for scale (k+1), there is no edge pixel (i+m, j+n), $-W_{k+1} \leq m, n \leq W_{k+1}$, such that $M_{k+1}f(i+m, j+n) > \alpha M_k f(i,j)$ where α is a relaxation parameter which controls the number of deleted edges. The larger α is, the more edges will be deleted. In our numerical experiments, α was chosen between 0.1 and 0.25.

The quasi-code for the modulus maxima denoising scheme is given as following:

Modulus Decay Based Denoising Algorithm

1. For k=1 to K-1
2. For i=1 to I
3. For j=1 to J
4. (i,j) is noisy edge if $M_k f(i,j) < \alpha M_{k+1} f(i+m, j+n)$ for all edge pixel (i+m,j+n) where $m, n = -W_{k+1}, -W_{k+1}+1, ..., 0, ...W_{k+1}$
5. Next j
6. Next i
7. Next k

6.5.3 Phase based Denoising. As shown by our discussions in the previous section, the phase of wavelet transform of regular Poisson noise is asymptotically uniformly distributed on $[0, 2\pi)$. On the other hand, the "true edges" of an image separate two neighboring regions and the neighboring edge pixel usually have similar gradient directions. Therefore, all edges constitute a map of lines or curves even though some of these lines or curves may be very short in length. Since the phase of wavelet transform of an image is the gradient direction of the image, it is reasonable to say that, for a "true edge" pixel, there exist at least one neighboring edge pixels which have similar wavelet phases. Otherwise, the edge pixel is most likely caused by noise. Based on this observation, we developed the following denoising scheme, which will be referred to as the "edge tracking" method.

1. Find tangent direction of the gradient direction for each edge pixel. The tangent direction of gradient direction is the direction orthogonal to gradient direction. Determine a search region along the tangent direction. Roughly, the search region is taken to be two triangles which are located on both sides of the edge pixel with it as a vertex and which are symmetric with respect to the tangent line. The search region is said to have order n if the searching is extended to the *nth* pixel from the edge pixel along the tangent line.

2. Delete edges for which there are no other edges with similar phase in the search region. Keep edges for which there is a curve of reasonable length and along which all edge pixels have similar phase values.

The quasi-code for the above edge tracking algorithm is given as following:

Edge Tracking Algorithm

1. For k=1 to K

2. For i=1 to I

3. For j=1 to J

4. Find wavelet phase Pf(i,j) according to (6.5.1)

5. $\theta(i,j) = \min\{|\theta - Pf(i,j)| : \theta = 0, \pi/4, \pi/2, 3\pi/4, \pi, 5\pi/4, 3\pi/2, 7\pi/4\}$

6. Determine a search region in the direction orthogonal to $\theta(i,j)$.

7. Find wavelet phase Pf(m,n) according to (6.5.1) for all (m,n) in the region.

8. Calculate the number of edges in the region satisfying $|Pf(m,n) - Pf(i,j)| < \theta_0$, where θ_0 is a threshold.

9. (i,j) is a "true edge" if the number obtained in Step 8 is greater than some thresholding value.

10. Next j

11. Next i

12. Next k

The θ_0 in Step 8 is the tolerance level of wavelet phase variation. The smaller value of θ_0 is chosen, the less tolerance of phase variation is allowed. The edge number in Step 8 determine how many edges will be removed. The larger number it is, the more edges will be removed. In our numerical experiments, θ_0 is chosen a value around $\pi/6$ and the edge number is chosen around 3 for a search region with order 4.

Another denoising scheme of exploiting wavelet phase is to consider local wavelet phase variation. For pixel (i,j) and positive integer W_k, let

$$\overline{W}_s^1 f(i,j) = \sum_{k=-W_k}^{W_k} \sum_{l=-W_k}^{W_k} W_s^1 f(i+k, j+l)$$

and

$$\overline{W}_s^2 f(i,j) = \sum_{k=-W_k}^{W_k} \sum_{l=-W_k}^{W_k} W_s^2 f(i+k, j+l)$$

where W_k is referred to as window size. The mean wavelet phase at (i,j) is defined by

$$\overline{P} f(i,j) = \mathrm{Arctan} \frac{\overline{W}_s^2 f(i,j)}{\overline{W}_s^1 f(i,j)}.$$

From the analysis in Section 4.3 and Section 6.3, the difference between the phase of $(W_s^1 f(x,y), W_s^2 f(x,y))$

$$P f(x,y) = \mathrm{Arctan} \frac{W_s^2 f(x,y)}{W_s^1 f(x,y)}$$

and $\overline{P} f(i,j)$ is expected to be small if the pixel (i,j) is "true edge" but large if the pixel (i,j) is an edge generated by noise. Therefore, one might distinguish "true edges" from noisy edges by thresholding $Pf(x,y) - \overline{P} f(i,j)$.

We can also discriminate noisy edges from true edges by considering phase changes from scale to scale. The algorithm is similar to that exploiting modulus maximum changes among scales. The quasi-code is given as follows:

Scale Variation Algorithm

1. For k=1 to K-1
2. For i=1 to I
3. For j=1 to J
4. Find wavelet phase Pf(i,j) according to (6.5.1)

5. Select window size W_k.

6. Compute wavelet phase $P_{k+1}f(i+m, j+n)$ according to (6.5.1) for all edge pixel (i+m, j+n) where $m, n = -W_{k+1}, -W_{k+1}+1, ..., 0, ...W_{k+1}$

7. (i,j) is noisy edge if $|P_{k+1}f(i+m, j+n) - P_k f(i,j)| > \theta_0$ for all edge pixel (i+m, j+n) where $m, n = -W_{k+1}, -W_{k+1}+1, ..., 0, ...W_{k+1}$ and θ_0 is a threshold.

8. Next j

9. Next i

10. Next k

In our numerical experiments, the W_k in Step 5 was chosen to be from 1 to 4 and θ_0 in Step 7 was chosen to be about $\pi/6$.

6.5.4 Wavelet Interpolation. After removing wavelet coefficients of noisy edges, what remains are wavelet coefficients of edge pixels. But the wavelet coefficients at all pixels are required to reconstruct the original image from its wavelet coefficients and smooth part. To obtain the wavelet coefficients at non-edge pixels, an interpolation scheme is suggested in [MALL92]. We also use this interpolation scheme in our algorithms. This subsection describe how to obtain wavelet coefficients for non-edge pixels located between two consecutive edge pixels.

Let us discuss one-dimensional case first. Let x_0 and x_1 be two consecutive edge pixels. For scale k, the interpolation function suggested in [MALL92] is

$$\epsilon_k(x) = \alpha e^{2^{-k}x} + \beta e^{-2^{-k}x} \tag{6.5.2}$$

where α and β satisfy the equation

$$\begin{cases} \alpha e^{2^{-k}x_0} + \beta e^{-2^{-k}x_0} = W_k f(x_0) \\ \alpha e^{2^{-k}x_1} + \beta e^{-2^{-k}x_1} = W_k f(x_1) \end{cases}$$

where $W_k f(x_0)$ and $W_k f(x_1)$ are the wavelet coefficients of signal f at location x_0 and x_1 for scale k. The wavelet coefficient at location x is approximated by $\epsilon_k(x)$.

For the two dimensional case, we essentially apply one dimensional interpolation for each row and each column. Let (x_0, y) and (x_1, y) be two consecutive edge pixels in the same row and let $(W_k^x f(x_0, y), W_k^y f(x_0, y))$ and $(W_k^x f(x_1, y),$ $W_k^y f(x_1, y))$ be wavelet coefficients of the image f at (x_0, y) and (x_1, y). We apply (6.5.2) to interpolate x component $W_k^x f(x, y)$. For x located between x_0 and x_1 in the row y, $(W_k^x f(x, y)$ can be interpolated by

$$\epsilon_k(x) = \alpha_x e^{2^{-k} x} + \beta_x e^{-2^{-k} x} \tag{6.5.3}$$

where α_x and β_x satisfy

$$\begin{cases} \alpha_x e^{2^{-k} x_0} + \beta_x e^{-2^{-k} x_0} = W_k^x f(x_0, y) \\ \alpha_x e^{2^{-k} x_1} + \beta_x e^{-2^{-k} x_1} = W_k^x f(x_1, y) \end{cases} \tag{6.5.4}$$

The interpolation for y components is similar. Let (x, y_0) and (x, y_1) be two consecutive edge pixels in the same column. For any y between y_0 and y_1, $(W_k^y f(x, y)$ is interpolated by

$$\epsilon_k(y) = \alpha_y e^{2^{-k} y} + \beta_y e^{-2^{-k} y} \tag{6.5.5}$$

where α_y and β_y satisfy

$$\begin{cases} \alpha_y e^{2^{-k} y_0} + \beta_y e^{-2^{-k} y_0} = W_k^y f(x, y_0) \\ \alpha_y e^{2^{-k} y_1} + \beta_y e^{-2^{-k} y_1} = W_k^y f(x, y_1) \end{cases} \tag{6.5.6}$$

Let K be the total number of scales, I be the total number of pixels in each row and J be the total number of pixels in each column. Let NX[k] store the number of edge pixels in x direction and NY[k] store the number of edge pixels in y direction. The quasi-code for two dimensional wavelet interpolation is given as follows:

1. For k =1 to K
2. For j =1 to J
3. For n =1 to NX[k]-1

4. Interpolate x component of wavelet coefficients for pixels between i_n and i_{n+1} in each row using (6.5.3) and (6.5.4).

5. Next n

6. Next j

7. For i =1 to I

8. For m =1 to NY[k]-1

9. Interpolate y component of wavelet coefficients for pixels between m and (m+1) in each column using (6.5.5) and (6.5.6).

10. Next m

11. Next i

12. Next k

6.6 Numerical Experiments

This section describes the numerical experiments of implementing wavelet phase algorithm and also compares the results obtained by the wavelet phase algorithm with those obtained by other methods.

6.6.1 Experimental Environments. The numerical experiments in this research were conducted on a SUN workstation with Xwindow system. The programs are written by C and FORTRAN languages. Some IMSL (International Mathematics and Statistics Library) routines were called to generate Poisson random numbers.

6.6.2 Procedures in PET Image Processing. Several procedures are used in our PET image reconstruction and restoration process: a procedure to build a model (Shepp-Logan phantom), a procedure to generate projection data, a procedure to simulate random counts, some procedures to reduce noise and a procedure to reconstruct the image. These procedures are illustrated in Figure 1.

Shepp-Logan Phantom. In PET image processing, we first need to build a model which simulates the radioactivity occurring in the human brain.

The Shepp-Logan phantom [Figure 3] is a typical model which indicates the concentration of some marked chemical inside the brain by different gray levels. The phantom consists of 10 ellipses; each of them has a different gray level. This phantom has been used as a standard to test the accuracy of reconstruction and restoration algorithms.

Parallel Projection. After building model, we need some procedures to simulate PET emission process. Parallel projection is a conventional procedure for Fourier reconstruction methods. The parallel projection procedure generates a set of data, which represents the number of detected photons in detector pairs, by adding up pixel values for each pixel along parallel lines in each direction. The data set with indices of directions and segments in each direction constitutes a two-dimensional array, which is referred to as raysum image [Figure 7].

Simulation of Poisson Noise. According to our analysis in Chapter 3, the pixel values in the raysum image are independent Poisson random variables. To simulate these random variables, we use a Poisson random number generator to generate independent random numbers which have means equal to the values obtained in the projection procedure. Our wavelet filters are applied to these random variables. We know from Chapter 3 that the greater the total number of photons observed, the less noisy the raysum image. In our numerical experiments, we assume that 2 million photons are observed in total. The raysum image of 2 million counts is seen as Figure 8.

Denoising. Our denoising procedures are applied to the random raysum image. These procedures implement the algorithms stated in the previous section. Figure 2 illustrates these procedures.

Reconstruction. After denoising procedure, we need a procedure to reconstruct images from the restored raysum data. Backprojection is a conventional method, and direct Fourier reconstruction is an alternative approach. We have implemented both methods for the restored raysum images. Since the convolution back projection method, while more costly, produced consistently better reconstructed images, we give only one image which used direct Fourier reconstruction. See Figure 27 (compared with Figure 9).

6.6.3 Experimental Results. Figures 3 to 37 give the numerical results obtained in this study. The size of the original image (Shepp-Logan phantom) is 128x128 and the size of the raysum image is 180x128 (We only display the first 128 rows of the raysum image for convenience, see Figure 7). The pixel values of all images are normalized to 8 bits (ranged from 0 to 255). A total of 2 million photons is used for all noisy cases.

Figure 9 is the image reconstructed from noiseless raysum data by using the Convolution Back Projection (CBP) method with ramp filter. From the figure, we can see that most features of the Shepp-Logan phantom are preserved. Figure 10 is the image reconstructed from noisy raysum by using CBP method with ramp filter. From the figure, we can see a circle hidden in the middle upper side but cannot see any small features.

Figure 11 is the image reconstructed from noiseless raysum data by using CBP method with the parabola filter. We can see that the image is blurred by the lowpass filter. Figure 12 is the image reconstructed from noisy raysum data by using CBP method with a lowpass filter called parabola filter. Some small features present in the original Shepp-Logan phantom can be seen clearly but the image is very blurred. The parabola filter is a special lowpass filter for PET image restoration and the computer source code used here is from Washington University. According to [YANG91], the filter has been among the most effective lowpass filters for CBP method. The filter is called "parabola" since it is a parabola function in the frequency domain. In fact, the filter is the product of triangle function

$$u(x) = \frac{x}{N}\delta_{[-N,N]}(x)$$

and ramp function

$$u(x) = \frac{N-x}{N}\delta_{[-N,N]}(x)$$

where N is the cutoff frequency.

Figure 13 is the image reconstructed from noiseless raysum by using CBP with wavelet modulus filter. Figure 14 is the image reconstructed from noisy raysum by using CBP with wavelet modulus filter. Some small features can be seen clearly but the image still has high variation. For both Figure 13 and Figure 14, wavelet transform is performed up to 3 scales. Modulus thresholding values are 0.2 for scale 1, 0.1 for scale 2 and 0.05 for scale 3. The decay rate from one to next is thresholded by 0.1.

Figure 15 to 18 are images reconstructed by using CBP method combined with edge tracking scheme, a type of wavelet phase filter discussed in Section 6.5.3. Figure 15 and 17 are reconstructed from noiseless raysum data while Figure 16 and 18 are reconstructed noisy raysum data. For Figure 15 and 16, a wavelet modulus filter as that used in Figure 13 and 14 is added to the phase filter. Figure 17 and 18 are obtained without modulus filter. The thresholding length of edge curves in the algorithm described in Section 6.5.3 is 3. Here we find that two small features can be seen very clearly for noisy cases and that phase and modulus combined filter perform better than separate modulus or phase filter.

Figure 19 to 22 are images reconstructed by using CBP method combined with local phase variance thresholding scheme, another type of wavelet phase filter discussed in Section 6.5.3. Figure 19 and 21 are reconstructed from noiseless raysum data while Figure 20 and 22 are reconstructed noisy raysum data. For Figure 19 and 20, a wavelet modulus filter as that in Figure 13 and 14 is added to the phase filter. Figure 21 and 22 are obtained without modulus filter. The thresholding value for local phase variation is $\frac{\pi}{4}$. From the figures, we see that this method does not significantly improve the quality of the reconstructed images over those obtained by wavelet modulus filter.

Figure 23 to 26 are images reconstructed by using CBP method combined with phase scale variation thresholding scheme, the third type of wavelet phase filter discussed in Section 6.5.3. Figure 23 and 25 are reconstructed from noiseless raysum data while Figure 24 and 26 are reconstructed noisy raysum data. For Figure 23 and 24, a wavelet modulus filter as that in Figure 13 and 14 is added to the phase filter. Figure 25 and 26 are obtained without modulus filter. The phase variation from scale to next is thresholded to be $\frac{\pi}{6}$. from the images, two small features can also be seen clearly as the case of edge tracking scheme.

Figure 27 is the image reconstructed from noiseless raysum data by using Direct Fourier Reconstruction (DFR) method. The method is much faster than CBP method (see the analysis in Chapter 2). But we can see ringing artifacts in the image. Figure 28 is the image reconstructed from noisy raysum by using DFR method. From the figure, we hardly see any small features inside the region of interest.

Figure 29 is the image reconstructed from noisy raysum by using DFR method with lowpass filter. Figure 30 is the image reconstructed from noisy raysum by using DFR method with square rooted lowpass filter [see Section 3.5]. These two filters are considered to be the most effective ones among all conventional filtering schemes in DFR reconstruction [YANG91]. From Figure 29 and 30, the big circle located in the middle upper side can be seen clearly and two small features in the middle of the region of interest can be seen in some degree.

Figure 31 is the image reconstructed from noisy raysum by using DFR with wavelet modulus filter. For the noisy case, some small features can be seen clearly but the image is still very noisy. For Figure 31, wavelet transform is performed up to 3 scales. Modulus thresholding values are 0.2 for scale 1, 0.1 for scale 2 and 0.05 for scale 3. The decay rate from one to next is thresholded by 0.1.

Figure 32 is the image reconstructed from noisy raysum by using DFR method combined with edge tracking scheme, the same phase filter as in Figure 16. The thresholding length of edge curves in the algorithm described in Section 6.5.3 is 3. For the image, two small features can be seen more clearly than those in the images obtained by using wavelet modulus filter.

Finally, it may be helpful if we take a look at the restoration of the raysum image using different filtering techniques. Since it is difficult to distinguish significant features in the raysum image, we use the standard Lena image [Figure 33] for simulation. The Lena image has 256×256 pixels and 8 bits gray scale per pixel. Figure 34 is a noisy Lena image where noise is added in the same manner as that in the raysum image (see Figure 8) except photon counts are 100,000 in the Lena image while the photon counts are 2 million in the raysum image. Figure 35 is the restored Lena image using a 5x5 lowpass filter which moves a window over the noisy Lena image and takes average for the pixels inside the window. Figure 36 is the restored Lena image using the wavelet modulus thresholding technique where thresholding values are 0.4, 0.25, 0.15 for scale 1 to 3. Figure 37 is the restored Lena image using the wavelet modulus thresholding with scale phase variation thresholding where modulus thresholding values are 0.2, 0.15, 0.15 for scale 1 to 3 and phase thresholding value is $\pi/3$.

The following facts have been observed in our numerical experiments: For the lowpass filter, the noise in the image has been suppressed rather than removed (see Figure 35). For PET image reconstruction problem, it is an open issue whether the suppressed noise will be amplified during back projection so that the lowpass filter is not effective for restoration in PET image reconstruction. A wavelet modulus based technique cannot remove noise with a high level of variance, no matter how large a threshold is chosen (see Figure 36). The high level noise can be removed by a wavelet phase filter (see Figure 37). This is a heuristic argument for why a wavelet phase filter performs better than a wavelet modulus filter or a lowpass filter.

6.7 Evaluation and Analysis

6.7.1 Quality Evaluations. Quality Evaluation is an important issue in image restoration. In order to measure the efficiency of a restoration algorithm, it is required to evaluate the quality of the restored images. Usually, one may consider factors such as the smoothness and sharpness of the restored image as well as the reduction of artifacts in the images. There are two approaches to evaluate these factors, subjective methods and objective methods. A subjective method measures these factors by direct observation of human eyes. The subjective evaluation is effective and reliable in many applications. However, it might also be affected by viewers' taste, experience and backgrounds. An objective method measures the quality of images by quantitative criteria such as L_1 norm, L_2 norm, SNR ratio and contrast. It can also be fulfilled by standardized procedures or equipment. An objective method is repeatable and quantitatively comparable. However, Designing an effective objective methodology which captures subjective quality in some degree is not an easy task and is still an open issue for many applications.

Let us do some subjective evaluation for the restored PET images. By comparing Figure 16 with Figure 14 of CBP method and Figure 32 with Figure 31 of DFR method, we see that features can be seen more clear in the images restored by wavelet phase filter than in the images restored by a wavelet modulus filter. By comparing Figure 16 with Figure 12 of CBP method and Figure 32 with Figure 30 of DFR method, it also can be seen that the wavelet phase filter performs better than a conventional lowpass filter.

Next, let us do some objective evaluations for the restored PET images. The L_2 norm error (mean square error) and L_1 norm error (absolute error) are two conventional measurements for measuring the accuracy of restoration algorithms. Suppose the size of images is M $\times$ N. The normalized L_2 norm error is calculated by

$$\epsilon_2(\%) = \frac{||f_R - f_T||_2}{||f_T||_2} \times 100\%$$

$$= \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}[f_T(i,j) - f_R(i,j)]^2}{\sum_{i=1}^{M}\sum_{j=1}^{N} f_T^2(i,j)}} \times 100\%$$

and the normalized L_1 norm error is calculated by

$$\epsilon_1(\%) = \frac{||f_R - f_T||_1}{||f_T||_1} \times 100\%$$

$$= \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}|f_T(i,j)-f_R(i,j)|}{\sum_{i=1}^{M}\sum_{j=1}^{N}|f_T(i,j)|}\times 100\%$$

where f_T and f_R denote the true image and the reconstructed image respectively. Sometimes, only the Region of Interest (ROI) is concerned and the summations in the preceding formulas are applied to the pixels inside the ROI. For example, for the Shepp-Logan phantom, the ROI is the pixels inside the biggest white ellipse. The errors measured by L_1 norm and L_2 norm are listed in Table 6.1 (see page 6.44) for CBP method and in Table 6.2 (see page 6.44) for DFR method. From the tables, we see that quantitative measurements does not coincide with the eye observation. Observe that L_1 and L_2 errors are large even for the noiseless raysum cases though the visual quality of the reconstructed image [Figure 9] is good. After restricting error measurement to the region of interest (ROI), the errors have been reduced significantly. This means that much error is caused in the region outside ROI during reconstruction. But also observe that Figure 10, Figure 12 and Figure 16 are very different in subjective quality, but they have almost the same L_1 and L_2 error in the region of interest. This is not very surprising if one considers that both L_1 and L_2 are measurements for global error but human observations pay more attentions to some local factors such as contracts around edges and visibility of small features. A feasible quantitative measurement is desirable.

6.7.2 Computational Costs. The motivation of this thesis is to improve the inexpensive Fourier reconstruction methods by using wavelet filters. We have seen that the quality of reconstructed images has been significantly improved after applying wavelet filters. One remaining issue is to see how much computational cost is added to achieve the improvement. We have seen that the computational cost for the DFR method is bounded by $O(N^2 \log_2 N)$ and that that for the CBP method is bounded by $O(N^3)$ for an image with size N $\times$ N. Are the reconstruction methods still bounded by these cost levels after adding wavelet filter algorithms? We will answer this question in the next paragraphs. To answer this question, we will evaluate computational costs of all involved procedures including wavelet transform, edge detection, wavelet modulus based denoising, wavelet phase based denoising, wavelet interpolation and inverse wavelet transform. Before analyzing each procedure, we want to point out that we did not use the technique of subsampling the smoothed parts of wavelet transforms, which will reduce the storage requirements and will also theoretically reduce the computational costs. The reason we did not do so is due to that the Haar basis is simple and does not introduce too much computational burden and that the subsampling technique makes it difficult to compare pixel values in denoising process.

First, we evaluate the costs in the processes of wavelet transform and inverse wavelet transform. Suppose that the image size is N $\times$ N and that total number of scales is K. Then, it is easy to see that the costs of wavelet transform and inverse wavelet transform are bounded by $O(KN^2)$ [LIN94]. Since K is usually taken a number less than $\log_2 N$, the costs will be bounded by $O(N^2 \log_2 N)$. In our numerical experiments, K is taken to be 3. Therefore, the costs of these two procedures will not exceed the bounds of the reconstruction algorithms.

Second, we evaluate the computational cost of the procedure of edge detection where a modulus thresholding based denoising scheme is embedded. Let us look at the modulus thresholding algorithm in Section 6.5.2. For each scale, it takes one function evaluation (Arctan in Step 4) and at most 4 comparisons (Step 5) to determine gradient direction for each pixel. It also takes 2 multiplications (Step 6) to calculate the square of the modulus and at least 1 comparison (Step 7(a)) and at most 6 comparisons (Step 7(b)) to determine if it is a "true edge". Here we see that the total cost for this algorithm is still bounded by $O(N^2 \log_2 N)$. Using the same technique, we can see that the cost of the denoising scheme by exploiting modulus decay from scale to scale is also bounded by $O(N^2 \log_2 N)$.

Third, we evaluate the computational cost of the denoising scheme by exploiting wavelet phase filters. Let us look at the edge tracking algorithm as an example. For each edge pixel (i,j), it takes one function evaluation (Arctan in Step 4) and at most 4 comparisons (Step 5) to determine gradient direction for each pixel. Suppose the region in Step 6 contains L pixels. It will take L function evaluations (Step 7) and L comparisons (Step 8) to find all pixels which have the similar phases as that of the pixel (i,j). It also takes one comparison (Step 9) to determine if the pixel (i,j) is an edge generated by noise. Here we see that the cost is bounded by O(KLM), where M is the total number of edge pixels. Since the number of edge pixels is usually much less than the number of total pixels and since $K < \log_2 N$, the cost of the edge tracking technique is still bounded by $O(N^2 \log_2 N)$.

Finally, let us look at the process of recovering all wavelet transforms from the wavelet transforms of denoised edges. From Subsection 6.4.4, we see that the wavelet transforms of non-edge pixels are recovered by interpolating the wavelet transforms of two neighboring edge pixels. Since only one function evaluation is required for each non-edge pixel, the total cost of the interpolation process is also bounded by $O(N^2 \log_2 N)$.

From the above analysis, we conclude that the cost of the wavelet filtering schemes is bounded by $O(N^2 \log_2 N)$. Hence, it will not exceed the cost levels of the reconstruction algorithms by adding wavelet filters.

Table 6.1. The Error (%) of Reconstructed Images by CBP

	L_2 Norm	L_1 Norm	L_2 (ROI)	L_1 (ROI)	L_2 (Edge)	L_1 (Edge)
Noiseless	39.01	42.31	28.21	16.41	49.17	43.67
Ramp	59.51	75.67	32.76	24.96	50.59	49.05
Parabola	39.48	44.78	31.53	24.16	51.73	52.05
Modulus	56.32	71.33	38.10	29.67	53.15	51.23
Edge Tracking	52.75	66.05	34.08	24.78	51.78	49.78

Table 6.2. The Error (%) of Reconstructed Images by DFR

	L_2 Norm	L_1 Norm	L_2 (ROI)	L_1 (ROI)	L_2 (Edge)	L_1 (Edge)
Noiseless	41.42	49.58	22.66	15.43	46.73	39.65
Non-filter	73.67	93.00	31.10	24.48	51.81	49.77
Lowpass	46.36	59.21	26.67	21.76	47.47	46.30
Square Root	46.81	58.39	27.41	19.69	45.71	43.93
Modulus	66.50	84.49	33.28	25.88	49.72	47.81
Edge Tracking	60.47	77.71	30.40	23.85	47.34	46.51

CHAPTER VII
CONCLUSIONS

This chapter summarizes what is achieved in this thesis and points out what remains to be studied further.

7.1 Summary of the Thesis

The most significant contribution of this thesis, from the view of the writer, is the study of wavelet phase, a characteristic of 2D wavelet transform. In this thesis, we proved that the wavelet phase of Gaussian noise under the symmetric Haar transform is a random variable uniformly distributed on $[0, 2\pi)$ and that the wavelet phase of regular Poisson noise under the symmetric Haar basis converges in distribution to a random variable uniformly distributed on $[0, 2\pi)$. Applying the property of the wavelet phase of Poisson noise to PET image restoration, we improved the quality of the reconstructed PET images, in comparison with a traditional lowpass filtering scheme and also in comparison with Mallat and Hwang's wavelet modulus thresholding scheme. Since the computational costs of wavelet phase based algorithms are still bounded by the costs of the original reconstruction algorithms, the wavelet phase based approach does not introduce too much additional computational burden. Therefore, the wavelet phase based approach is effective and feasible.

Another contribution of this thesis is that it provided a systematic analysis of the PET image reconstruction problems from the mathematical point of view. In Chapter II, we studied the Poisson model of PET. In Chapter III, we studied the characteristics of the Poisson-type noise and reviewed several conventional filtering schemes. It was found that these schemes cannot efficiently filter out noise and preserve features of the image at the same time. In Chapter IV, we investigated several edge detection methods and found that the gradient angular method can avoid noise sensitivity occurring in the gradient modulus method and the Laplacian method and that multiscale edge detection is a useful tool for image analysis. The multiscale edge detection method and the gradient angular method were incorporated in the wavelet phase filter discussed later in Chapter VI. In Chapter V, we found that wavelet modulus based edge detection scheme is a natural extension of multiscale edge detection and that wavelet based filtering scheme is an approach which can filter out noise while preserving features present in the image. In Chapter VI, we analyzed the drawbacks of the wavelet modulus based denoising scheme, studied the properties of the wavelet phases of noise and proposed PET image denoising schemes based on the properties of the wavelet phases of noise. In summary, we gave a self-contained and rigorous analysis for PET image processing problems throughout the thesis.

7.2 Suggestions for Further Studies

In order to analyze the essence of the problem, we used a simplified model and intentionally neglected some physical details. Also, we considered only the Haar transform, one of the simplest wavelet transforms. This section discusses several further study issues.

1. Blurring Effects. As mentioned in Section 2.6, two emitted photons in real PET emission process do not travel on the exactly opposite directions so that the detected photons in detectors may not be independent; there also exists a distance between the location where a positron is emitted and the location where the positron annihilates with an electron to generate photons so that the reconstructed image does not reflect the radioactivities in the exact locations where the emission processes occur. Also, the number of detectors cannot be infinitely large so that there exists an approximation error in using projection model to estimate emission process. These facts generate a phenomenon called blurring effect. To simulate blurring effect, one may apply a Gaussian filter for each angle of view in raysum images. Solving blurring problems is an important issue in practical PET imaging, but we did not address it. To apply wavelet based denoising schemes to practical PET denoising problems, it would be desirable to take the blurring effects into account. This is an area for future research.

2. Other Wavelet Bases. We use the symmetric Haar basis in our study not only because the wavelet transform under such basis is easy to be implemented but also because the wavelet phase of the Poisson noise under such basis has a nice convergence property, Theorem 6.3.2 and Theorem 6.3.3. For other wavelet bases such as a Daubechies basis or a spline basis, what statistical properties the wavelet phase possesses and how well a wavelet phase based denoising scheme performs still remain open issues. They are also interesting possibilities for future research.

3. Quantitative Error Analysis. As mentioned in Section 6.7.1, the quality of the reconstructed images was improved by employing wavelet phase filter. The criterion used was, however, subjective. That is, the features in the original image appeared to be recovered better using wavelet phase filter than using wavelet modulus filter or a lowpass filter, according to human observation. But quantitative measurements such as L_1 norm and L_2 norm did not appear to agree with the subjective evaluations. It is desirable to have a quantitative measurement which can better reflect human perception of images.

4. Other Application Problems. In this thesis, we addressed denoising of PET images. But the properties of wavelet phase we explored are not limited to PET images. Section 6.3 also discussed the distribution of wavelet phase of Gaussian noise. The wavelet phase based denoising scheme is expected to be applicable for such cases. Generally, wavelet phase provides a mean to represent the correlation between neighboring edge pixels in the same scale and the correlation of edge pixels from scale to scale. It could be used to extract features in computer vision and detect edges in other medical imaging applications.

REFERENCES

[BILL68] P. Billingsley, *Convergence of Probability Measures*, John Wiley & Sons, New York, 1968.

[CHUN74] K. L. Chung, *A Course In Probability Theory*, 2nd Edition, Academic Press, Orlando, Florida, 1974.

[DERI84] H. Derin, H. Elliot, R. Cristi and D. Geman, "Bayes smoothing algorithms for segmentation of binary images modeled by Markov random field", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, No. 6, pp. 707-720, Nov. 1984.

[BESA86] J. Besag, "On the statistical analysis of dirty pictures", *J. Roy. Statist. Sco. B*, vol. 48, No. 3, pp.259-302, 1986.

[BOUM92] C. Bouman and K. Sauer, "A generalized Gaussian image model for edge-preserving MAP estimation", Technical Report, Purdue University, 1992.

[DAUB88] I. Daubechies, "Orthonormal bases of compactly supported wavelets", *Comm. Pure and Applied Math.*, vol 41, pp. 909-996, 1988.

[DAUB90] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis", *IEEE Trans. Inform. Theory*, vol 36, pp. 961-1005, Sept. 1990.

[DYM72] H. Dym and H. P. McKean, *Fourier Series and Integrals*, Academic Press, New York and London, 1972.

[GEMA84] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, No. 6, pp.721-741, Nov. 1984.

[GREG93] P. H. Gregson, "Using angular dispersion of gradient direction for detecting edge ribbons", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 15, No. 7, pp. 682-696, July 1993.

[HAIG67] F. A. Haight, *Handbook of the Poisson Distribution*, John Wiley & Sons, New York, 1967.

[HELS67] C. W. Helstrom, "Image restoration by the method of least squares", *J. Opt. Soc. Amer.*, vol. 57, pp. 297-303, Mar. 1967.

[HEYM90] D. P. Heyman and M. J. Sobel, *Stochastical Models*, North-Holland, Netherlands and New York, 1990.

[KARR86] A. F. Karr, *Point Processes and Their Statistical Inference*, Dekker, New York, 1986.

[KUAN85] D. T. Kuan, A. A. Sawchuk, T. C. Strand and P. Chavel, "Adaptive noise smoothing filter for images with signal-dependent noise", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-7, No. 2, pp. 165-177, Mar. 1985.

[LANG84] K. Langer and R. Carson, "EM Reconstruction Algorithms for Emission and Transmission Tomography", *J. Comp. Assisted Tomography*, vol. 8, pp. 302-316, 1984.

[LEWI86] R. M. Lewitt and G. Muehllehner, "Accelerated iterative reconstruction for positron emission tomography based on the EM algorithm for maximum likelihood estimation", *IEEE Trans. Med. Imaging*, vol. MI-5, No. 1, pp. 16-22, Mar. 1986

[LIN94] B. Lin, B. Nguyen and E. T. Olsen, "Chapter I, Orthogonal Wavelets and Signal Processing", *Signal Processing Methods for Audio, Images and Telecommunications*, Academic Press, New York, 1994.

[LIND90] T. Lindeberg, "Scale-space for discrete signals", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 12, No. 3, pp. 234-254, Mar. 1990.

[LIOW91] J. S. Liow and S. C. Strother, "Practical tradeoffs between noise, quantization, and number of iterations for maximum likelihood-based reconstructions", *IEEE Trans. Med. Imaging*, vol. 10, No. 4, pp. 563-571, Dec. 1991

[MALL91] S. Mallat and S. Zhong, "Complete signal representation with multiscale edges", Technical Report, New York University, 1991.

[MALL92a] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges", *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 14, No. 7, pp. 710-732, July 1992.

[MALL92b] S. Mallat and W. L. Hwang, "Singularity detection and processing with wavelets", *IEEE trans. Inform. Theory*, vol. 38, No. 2, pp. 617-645, Mar. 1992.

[MARR80] D. Marr and E. Hildreth, "Theory of edge detection", *Proc. Roy. Sco. Lon.*, vol. 207, pp. 187-217, 1980.

[MEYE90] Y. Meyer, *Ondelettes et Operateurs I, Ondelettes*, Hermann, Paris, 1990.

[MEYE93] Y. Meyer, (translated by R. Ryan) *Wavelets Algorithms and Applications*, SIAM, Philadelphia, 1993.

[PEEB76] P. Z. Peebles, Jr., *Communication System Principles*, pp. 69-70, Addison Wesley, New York, 1976.

[RIOU91] O. Rioul and M. Vetterli, "Wavelets and signal processing", *IEEE Signal Proc. Magazine*, vol.8, pp.14-38, Oct. 1991.

[SAGE71] A. P. Sage and J. L. Melsa, *Estimation Theory with Applications to Communications and Control*, McGraw-Hill, New York, 1971.

[SAUE93] K. Sauer and C. Bouman, "A local update strategy for iterative reconstruction from projections", *IEEE Trans. Signal Proc.*, in review

[SHEP82] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction in positron emission tomography", *IEEE Trans. Med. Imaging*, vol. MI-1, No. 2, pp. 113-122, Oct. 1982

[SLEP83] D. Slepian, "Some comments on Fourier analysis, uncertainty and modeling", *SIAM Review*, vol 25, No. 3, pp. 379-393, July 1983.

[SNYD87] D. L. Snyder, M. I. Miller, L. J. Thomas and D. G. Politte, "Noise and edge artifacts in maximum-likelihood reconstructions for emission tomography", *IEEE Trans. Med. Imaging*, vol. MI-6, No. 3, pp. 228-238, Sept. 1987

[STAR79] H. Stark, "Sampling theorems in polar coordinates", *J. Opt. Soc. Am.*, vol. 69, pp.1519-1524, Nov. 1979.

[STAR81 a] H. Stark, J. Wood, I. Paul and R. Hinorsni, "Direct Fourier reconstruction in computer tomography", *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-29, pp. 237-245, April 1981.

[STAR81 b] H. Stark, J. Wood, I. Paul and R. Hinorsni, "An investigation of computerized tomography by direct Fourier inversion and optimum interpolation", *IEEE Trans. Biomed. Eng.*, vol. BME-28, pp. 496-505, July 1981.

[STAR92] H. Stark and E. T. Olsen, "Projection based image restoration", *J. Opt. Soc. Amer.-A.*, in review.

[TREE68] H. V. Trees, *Detection, Estimation, and Modulation Theory*, John Wiley & Sons, New York, 1968.

[VARD85] Y. Vardi, L. A. Shepp and L. Kaufman, "A statistical model for positron emission tomography", *J. Amer. Statist. Assoc.*, vol. 80, pp. 8-20, Mar. 1985.

[YANG91] D. Yang, "Noise smoothing in direct Fourier positron emission tomography", M.S. Thesis, Illinois Institute of Technology, Dec. 1991.

[YOUL82] D. C. Youla and H. Webb, "Image restoration by the method of convex projection: Part I-Theory", *IEEE trans. Med. Imag.*, Vol. MI-1, No. 2, pp.81-94, Oct. 1982.

[ZAKH92] A. Zakhor, "Iterative procedures for reduction of blocking effects in transform image coding", *IEEE Trans. Circuits Syst.*, vol.2, No. 1, pp. 91-95, Mar. 1992.