An Introduction to Wavelet Transform

Pao-Yen Lin
E-mail: r97942117@ntu.edu.tw
Graduate Institute of Communication Engineering
National Taiwan University, Taipei, Taiwan, ROC

Abstract

Wavelet transforms have become one of the most important and powerful tool of signal representation. Nowadays, it has been used in image processing, data compression, and signal processing. This paper will introduce the basic concept for Wavelet Transforms, the fast algorithm of Wavelet Transform, and some applications of Wavelet Transform. The difference between conventional Fourier Transform and modern time-frequency analysis will also be discussed.

1 Introduction

In conventional Fourier transform, we use sinusoids for basis functions. It can only provide the frequency information. Temporal information is lost in this transformation process. In some applications, we need to know the frequency and temporal information at the same time, such as a musical score, we want to know not only the notes (frequencies) we want to play but also when to play them. Unlike conventional Fourier transform, wavelet transforms are based on small waves, called wavelets. It can be shown that we can both have frequency and temporal information by this kind of transform using wavelets. Moreover, images are basically matrices. For this reason, image processing can be regarded as matrix processing. Due to the fact that human vision is much more sensitive to small variations in color or brightness, that is, human vision is more sensitive to low frequency signals. Therefore, high frequency components in images can be compressed without distortion. Wavelet transform is one of a best tool for us to determine where the low frequency area and high frequency area is. These kinds of applications will be discussed later.

2 Background

When we look at some images, generally we see many regions (objects) that are
formed by similar texture. If the objects are small in size, we normally examine them at high resolutions. Contrary we examine big objects at low resolutions. This is the fundamental motivation for multi-resolution processing.

### 2.1 Image pyramids

We start from a powerful, but conceptually structure for representing signal at more than one resolution called *image pyramid*. It was first proposed in early 1980s. As shown in Fig. 1, the base of the pyramid has the highest resolution however the apex contains a low-resolution approximation. The image size and resolution both decrease during moving up the pyramid. For a fully populated pyramid, we have $J + 1$ resolution levels from $2^J \times 2^J$ to $2^0 \times 2^0$, but generally we truncate to $P + 1$ levels from $2^J \times 2^J$ to $2^{J-P} \times 2^{J-P}$. Fig. 2 shows a simple system for constructing image pyramids. The *level J−1 approximation* is used to create *approximation pyramids*. In addition, the *level J prediction residual* is used to create *prediction residual pyramids*. A $P + 1$ level pyramid is built by executing through the operations $P$ times in the block diagram of Fig. 2. Moreover, we only need the level $J − P$ approximation and $P$ prediction residual then we can construct the original level $J$ image by adding the approximation and prediction residual iteratively.

![Fig. 1 a J-level image pyramid](image1.png)

![Fig. 2 Block diagram for creating image pyramids](image2.png)
2.2 Subband coding

In addition to image pyramid, *subband coding* is another technique for multiresolution analysis. In subband coding, an image is decomposed into a set of bandlimited components, called *subbands*. It can be proved that we can perfectly reconstruct the original image using the subbands. Because the bandwidth of the resulting subbands $y_0(n)$ and $y_1(n)$ are smaller than the original $x(n)$, it can be downsampled without loss of information. Fig. 3 shows the components of a two-band subband coding and decoding system. Reconstruction of the original image is accomplished by upsampling, filtering, and summing the individual subbands. For error-free reconstruction of the input, we impose the following conditions:

$$H_0(-z)G_0(z) + H_1(-z)G_1(z) = 0$$
$$H_0(z)G_0(z) + H_1(z)G_1(z) = 2$$

For finite impulse response (FIR) filters ignoring the delay, we can also find a relation between the analysis and synthesis bandpass filter:

$$g_0(n) = (-1)^n h_1(n)$$
$$g_1(n) = (-1)^{n+1} h_0(n)$$

From Eq. 2, we can see that if the analysis FIR filters’ coefficients are known, we can find their corresponding synthesis FIR filters uniquely.

![Diagram](image)

**Fig. 3** Two-band filter bank for one-dimensional subband coding and decoding system and the corresponding spectrum of the two bandpass filters
3 Time-Frequency atoms

When we listen to music, we clearly hear the time variation of the sound frequencies. The properties of sounds are revealed by transforms that decompose signals over elementary functions that are well concentrated in time and frequency. Windowed Fourier transforms and wavelet transforms are two important classes of local time-frequency decompositions. A linear time-frequency transform correlates the signal with a family of waveforms that are well concentrated in time and in frequency. These waveforms are called time-frequency atoms. Let us consider a family of time-frequency atoms \( \{ \varphi_\gamma \}_{\gamma \in \mathbb{N}} \). We suppose that \( \varphi_\gamma \in L^2(\mathbb{R}) \) and that \( \| \varphi_\gamma \| = 1 \).

The corresponding linear time-frequency transform for a function \( f \in L^2(\mathbb{R}) \) is defined by

\[
T_f(\gamma) = \int_{-\infty}^{+\infty} f(t) \varphi_\gamma^*(t) \, dt = \langle f, \varphi_\gamma \rangle
\]  

(3)

If \( \varphi_\gamma(t) \) is nearly zero when \( t \) is outside a neighborhood of an abscissa \( u \), then \( \langle f, \varphi_\gamma \rangle \) depends only on the values of \( f \) in this neighborhood.

3.1 Heisenberg Boxes

The slice of information provided by \( \langle f, \varphi_\gamma \rangle \) is represented in a time-frequency plane \((t, \omega)\) by a region whose location and width depends on the time-frequency spread of \( \varphi_\gamma \). In last section, we have seen that \( \| \varphi_\gamma \| = 1 \). Thus we can write

\[
\| \varphi_\gamma \|^2 = \int_{-\infty}^{+\infty} | \varphi_\gamma(t) |^2 \, dt = 1
\]  

(4)

we interpret \( | \varphi_\gamma(t) |^2 \) as a probability distribution centered at

\[
\mu_\gamma = \int_{-\infty}^{+\infty} t \, | \varphi_\gamma(t) |^2 \, dt
\]  

(5)

The spread around \( \mu_\gamma \) is measured by the variance

\[
\sigma_\gamma^2(\gamma) = \int_{-\infty}^{+\infty} (t - \mu_\gamma)^2 \, | \varphi_\gamma(t) |^2 \, dt
\]  

(6)

By Plancherel formula \( \int_{-\infty}^{+\infty} | \hat{\varphi}_\gamma(\omega) |^2 \, d\omega = 2\pi \| \varphi_\gamma \|^2 \), we can show that the center frequency of \( \varphi_\gamma \) is therefore defined by
Fig. 4  Heisenberg box representing an atom $\varphi_\gamma$.

$$\omega_\gamma = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \omega |\hat{\varphi}_\gamma(\omega)|^2 d\omega$$  \hspace{1cm} (7)

and its spread around $\omega_\gamma$ is

$$\sigma_\omega^2(\gamma) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (\omega - \omega_\gamma)^2 |\hat{\varphi}_\gamma(\omega)|^2 d\omega$$  \hspace{1cm} (8)

The time-frequency resolution of $\varphi_\gamma$ is represented in the time-frequency plane $(t, \omega)$ by a Heisenberg box centered at $(\mu_\gamma, \omega_\gamma)$, whose width along time is $\sigma_t(\gamma)$ and whose width along frequency is $\sigma_\omega(\gamma)$. The Heisenberg uncertainty proves that the area of the rectangle is at least $1/2$ :

$$\sigma_t \sigma_\omega \geq \frac{1}{2}$$  \hspace{1cm} (9)

This is illustrated by Fig. 4.

4  Windowed Fourier Transform

In 1946, Gabor introduced windowed Fourier atoms to measure the “frequency variations” of sounds. A real and symmetric window $g(t) = g(-t)$ is translated by $\mu$ and modulated by the frequency $\zeta$ :

$$g_{\mu,\zeta}(t) = e^{i\zeta t} g(t - \mu)$$  \hspace{1cm} (10)

The window function $g(t)$ is normalized. So that $\|g(t)\| = 1 = \|g_{\mu,\zeta}(t)\|$ for any $(\mu, \zeta) \in \mathbb{R}^2$. The resulting windowed Fourier transform of $f \in L^2(\mathbb{R})$ is written by:

$$S\{f(\mu, \zeta)\} = \langle f, g_{\mu,\zeta} \rangle = \int_{-\infty}^{+\infty} f(t) g(t - \mu) e^{-i\zeta t} dt$$  \hspace{1cm} (11)
Windowed Fourier transform is also called short time Fourier transform, because of the multiplication of the window function localized the Fourier integral in a neighborhood of $t = \mu$.

4.1 Heisenberg boxes of windowed Fourier atoms

We can also find the Heisenberg box by interpreting $|g_{\mu,\zeta}(t)|^2$ as a probability distribution. Calculate its variance we can find the center point and spread in the time-frequency plane. The time spread around $\mu$ is:

$$\sigma_t^2 = \int_{-\infty}^{+\infty} (t - \mu)^2 |g_{\mu,\zeta}(t)|^2 dt = \int_{-\infty}^{+\infty} t^2 |g(t)|^2 dt$$

(12)

By Parseval theorem, the frequency spread around $\zeta$ is:

$$\sigma_\omega^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (\omega - \zeta)^2 |\hat{g}_{\mu,\zeta}(\omega)|^2 d\omega = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \omega^2 |\hat{g}(\omega)|^2 d\omega$$

(13)

for which the Fourier transform of $g_{\mu,\zeta}(t)$ is:

$$\hat{g}_{\mu,\zeta}(\omega) = \hat{g}(\omega - \zeta) \exp[-i\mu(\omega - \zeta)]$$

(14)

We can easily find that both time and frequency spread is independent of $\mu$ and $\zeta$. It is shown in Fig. 5, which means that a windowed Fourier transform has the same resolution across the time-frequency plane.
5 Wavelet Transform

5.1 Continuous Wavelet Transform (CWT)
In order to analyze signals of very different sizes, it is necessary to use time-frequency atoms with different time supports. The wavelet transform decomposes signals over dilated and translated functions called wavelets, which transform a continuous function into a highly redundant function.

5.1.1 Wavelet function
A wavelet is a function with zero average:

$$\int_{-\infty}^{+\infty} \varphi(t) dt = 0$$ (15)

It is normalized $\|\varphi(t)\| = 1$ and centered at $t = 0$. A family of time-frequency atoms is obtained by scaling $\varphi(t)$ by $s$ and translating it by $\mu$:

$$\varphi_{\mu,s}(t) = \frac{1}{\sqrt{s}} \varphi\left(\frac{t - \mu}{s}\right)$$ (16)

We can now write the wavelet transform of $f \in L^2(\mathbb{R})$ at time $\mu$ and scale $s$:

$$W\{f(\mu,s)\} = \langle f, \varphi_{\mu,s} \rangle = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}} \varphi\left(\frac{t - \mu}{s}\right) dt$$ (17)

Now we define a variable $C_\varphi$, where

$$C_\varphi = \int_{0}^{+\infty} \frac{\left|\hat{\varphi}(\omega)\right|^2}{\omega} d\omega$$ (18)

It can be proved that if it satisfied $C_\varphi < +\infty$, called wavelet admissibility condition, then any $f \in L^2(\mathbb{R})$ has its inverse wavelet transform

$$f(t) = \frac{1}{C_\varphi} \int_{0}^{+\infty} \int_{-\infty}^{+\infty} W\{f(\mu,s)\} \frac{1}{\sqrt{s}} \varphi\left(\frac{t - \mu}{s}\right) du \frac{ds}{s^2}$$ (19)

Note that to guarantee the integral of $C_\varphi$ is finite we must ensure that $\hat{\varphi}(0) = 0$, which explains why we imposed that wavelets must have a zero average.

5.1.2 Scaling function
When $W\{f(\mu,s)\}$ is known only for $s < s_0$, to recover $f$ we need a complement of information corresponding to $W\{f(\mu,s)\}$ for $s > s_0$. It can be obtained by a scaling function $\Phi$, which is an aggregation of wavelets at scales larger than 1. We
define that
\[ |\hat{\phi}(\omega)|^2 = \int_{-\infty}^{+\infty} |\hat{\phi}(s\omega)|^2 \frac{ds}{s} = \int_{\omega}^{+\infty} |\hat{\phi}(\zeta)|^2 \frac{d\zeta}{\zeta} \]  
(20)

We can verify that \( \|\Phi\| = 1 \) and we derive from the admissibility condition that

\[ \lim_{\omega \to 0} |\hat{\phi}(\omega)|^2 = C_\phi \]  
(21)

Thus the scaling function can be interpreted as the impulse response of a low-pass filter. And the low-frequency approximation of \( f \) at scale \( s \) can be written as

\[ L\{f(\mu, s)\} = \langle f, \phi_{\mu, s} \rangle = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}} \phi^*(t - \mu/s) \, dt \]  
(22)

Therefore, the inverse wavelet transform can be rewritten as

\[ f(t) = \frac{1}{C_\phi} \int_{0}^{C_\phi} W\{f(., s)\} * \varphi_{\mu, s}(t) \frac{ds}{s^2} + \frac{1}{C_\phi s_0} L\{f(., s_0)\} * \phi_{\mu, s}(t) \]  
(23)

5.1.3 Heisenberg boxes of wavelet atoms

To analyze the time evolution of frequency tones, it is necessary to use an analytic wavelet to separate the phase and amplitude information of signals. A function is said to be analytic if its Fourier transform is zero for negative frequency.

\[ \hat{f}_a(\omega) = 0 \quad \text{for} \quad \omega < 0 \]  
(24)

We can decompose ones Fourier transform as sum of analytic function

\[ \hat{f}(\omega) = \frac{\hat{f}_a(\omega) + \hat{f}_a^*(-\omega)}{2} \]  
(25)

This relation can be inverted

\[ \hat{f}_a(\omega) = \begin{cases} 2\hat{f}(\omega) & \text{if} \quad \omega \geq 0 \\ 0 & \text{if} \quad \omega < 0 \end{cases} \]  
(26)

The analytic part \( f_a(t) \) of a signal \( f(t) \) is the inverse Fourier transform of \( \hat{f}_a(\omega) \). It is also called the pre-envelope of signal \( f(t) \).

The time-frequency resolution of an analytic wavelet transform depends on the time-frequency spread of the wavelet atoms \( \varphi_{\mu, s} \). We suppose that \( \varphi \) is centered at 0, which implies that \( \varphi_{\mu, s} \) is centered at \( t = \mu \). By changing variable \( v = \frac{t-\mu}{s} \), we verify that
\[
\int_{-\infty}^{+\infty} (t - \mu)^2 |\varphi_{\mu,s}(t)|^2 \, dt = s^2 \sigma_t^2
\]  
(27)

where \( \sigma_t^2 = \int_{-\infty}^{+\infty} t^2 |\varphi(t)|^2 \, dt \). Since \( \hat{\varphi}(\omega) \) is zero at negative frequencies, the center frequency \( \eta \) of \( \hat{\varphi}(\omega) \) is

\[
\eta = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \omega^2 |\hat{\varphi}(\omega)|^2 \, d\omega
\]  
(28)

and we find that the Fourier transform of \( \varphi_{\mu,s} \) is a dilation of \( \hat{\varphi}(\omega) \) by \( \frac{1}{s} \):

\[
\hat{\varphi}_{\mu,s}(\omega) = \sqrt{s} \hat{\varphi}(s\omega) \exp(-i\omega\mu)
\]  
(29)

Its center frequency is therefore \( \frac{\eta}{s} \). And the frequency spread around \( \frac{\eta}{s} \) is

\[
\frac{1}{2\pi} \int_{0}^{+\infty} \left( \omega - \frac{\eta}{s} \right)^2 |\hat{\varphi}_{\mu,s}(\omega)|^2 \, d\omega = \frac{\sigma_\omega^2}{s^2}
\]  
(30)

with

\[
\sigma_\omega^2 = \frac{1}{2\pi} \int_{0}^{+\infty} (\omega - \eta)^2 |\hat{\varphi}(\omega)|^2 \, d\omega
\]  
(31)

The energy spread of a wavelet time-frequency atom \( \varphi_{\mu,s} \) thus corresponds to a Heisenberg box centered at \( (\mu, \frac{\eta}{s}) \), of size \( s\sigma_t \), along time and \( \frac{\sigma_\omega}{s} \) along frequency. This is shown by Fig. 6. We can see that the area of the box remains \( \sigma_t \sigma_\omega \), but the

![Fig. 6 Heisenberg boxes of two wavelets. Smaller scales decrease the time spread but increase the frequency support and vice versa.](image)
resolution in time and frequency depends on \( s \). It is quite different to the Heisenberg box of windowed Fourier transform’s one.

### 5.1.3 Some examples of continuous wavelet

Nowadays, many kinds of wavelets are mentioned. Each with different characteristic and is used in different applications. In this section, we introduce some examples of continuous wavelet.

**Mexican hat wavelet**

The *Mexican hat wavelet* is defined as the second derivative of the Gaussian function:

\[
g(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}}
\]

which is

\[
\phi(t) = \frac{1}{\sqrt{2\pi}\sigma^3} \left[ e^{-\frac{t^2}{2\sigma^2}} \left( \frac{t^2}{\sigma^2} - 1 \right) \right]
\]

It is shown in Fig. 7.

**Morlet wavelet**

The most commonly used CWT wavelet is the *Morlet wavelet*, it is defined as following in time and frequency domains:

\[
\phi(t) = \pi^{-\frac{1}{4}} e^{imt} e^{-\frac{t^2}{2}}
\]

\[
\tilde{\phi}(\omega) = \pi^{-\frac{1}{4}} U(\omega) e^{-\frac{(\omega-m)^2}{2}}
\]

where \( U \) is step function, and \( m \) is an adjustable parameter of wavenumber that allows for an accurate signal reconstruction. This time and frequency domain plot is shown in Fig. 8. The white curve is the real component and the cyan curve is the complex component.

**Shannon wavelet**

The signal analysis by ideal pass-band filters define a decomposition known as Shannon wavelets. The Shannon wavelet and its Fourier Transform are:
The Mexican hat wavelet

Morlet wavelet with \( m \) equals to 3

The Shannon wavelet in time and frequency domains

\[
\varphi(t) = \text{sinc}(t/2)\cos(3\pi t/2) \quad (36)
\]

\[
\hat{\varphi}(f) = \begin{cases} 
1 & 0.5 \leq |f| \leq 1 \\
0 & \text{otherwise}
\end{cases} \quad (37)
\]

The Shannon wavelets in time and frequency domain are shown in Fig. 9.
A simple example
We give one example of time-frequency analysis using continuous wavelet transform.
In this example we choose Morlet wavelet for the mother wavelet. For an input signal
\[ f(t) \text{, where} \]
\[ f(t) = \begin{cases} 
\sin(2\pi100t) & 0 \leq t < 0.5 \\
\sin(2\pi200t) & 0.5 \leq t < 1 \\
\sin(2\pi400t) & 1 \leq t < 1.5 
\end{cases} \] (38)
The resulting wavelet transform is shown in Fig. 10. We can see that the resolution in
high frequency is less than low frequency part.

5.2 Discrete Wavelet Transform
If we uniformly sample a function \( \hat{f}(t) \) at intervals \( N^{-1} \) over \([0,1]\). By changing
of variable, the wavelet transform of \( \hat{f}(t) \) is
\[ W\{\hat{f}(\mu, s)\} = N^{-1/2}W\{f(N\mu, Ns)\} \] (39)
The discrete wavelet transform is computed at scales \( s = a^j \). By choosing \( a = \frac{1}{2} \), we
can define a discrete wavelet set \( \{\varphi_{j,k}(x)\} \) where
\[ \varphi_{j,k}(x) = 2^{j/2} \varphi(2^j x - k) \] (40)
For all \( j, k \in \mathbb{Z} \) and \( \varphi(x) \in L^2(\mathbb{R}) \). And we can also write the scaling function as
\[ \phi_{j,k}(x) = 2^{j/2} \phi(2^j x - k) \] (41)
Here, \( k \) determines the position of \( \varphi_{j,k}(x) \) and \( \phi_{j,k}(x) \) along the x-axis; and \( 2^{j/2} \)controls their height or amplitude. Fig. 11 and Fig. 12 give some examples of scaling
and wavelet functions. By choosing the scaling function \( \Phi(x) \) wisely, \( \{\Phi_{j,k}(x)\} \) can
be made to span \( L^2(\mathbb{R}) \).

![Fig. 10](image_url)

the resulting wavelet transform of Eq. (38) using Morlet wavelet
Generally, we will denote the subspace spanned over \( k \) for any \( j \) as

\[
V_j = \text{Span} \{ \phi_{j,k}(x) \}
\]  

(42)

We can verify that the size of \( V_j \) can be increased by increasing \( j \), allowing functions with finer detail. There are four fundamental requirements of multiresolution analysis that scaling function and wavelet function must follow:

1. The scaling function is orthogonal to its integer translates.
2. The subspaces spanned by the scaling function at low resolutions are contained within those spanned at higher resolutions:

\[
V_{-\infty} \subset \cdots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset V_{+\infty}
\]  

(43)

3. The only function that is common to all \( V_j \) is \( f(x) = 0 \). That is

\[
V_{-\infty} = \{0\}
\]  

(44)
4. Any function can be represented with arbitrary precision. As the level of the expansion function approaches infinity, the expansion function space $V$ contains all the subspaces.

$$V_{+\infty} = \{L^2(R)\}$$ (45)

Under these conditions, the expansion functions of subspace $V_j$ can be expressed as a weighted sum of the expansion functions of subspace $V_{j+1}$

$$\phi_{j,k}(x) = \sum_n \alpha_n \phi_{j+1,k}(x)$$ (46)

Substituting for $\Phi_{j+1,k}(x)$ and changing variable $\alpha_n$ to $h_\phi(n)$, this becomes

$$\phi(x) = \sum_n h_\phi(n) \sqrt{2} \phi(2x - n)$$ (47)

where the $h_\phi(n)$ are called the scaling function coefficients and $h_\phi$ is referred to as a scaling vector. On the other hand, for all $k \in Z$ we denote the subspace spanned by discrete wavelet set as:

$$W_j = \text{span}\{\varphi_{j,k}(x)\}$$ (48)

And we define the discrete wavelet set $\varphi_{j,k}(x)$ spans the difference between any two adjacent scaling subspaces, $V_j$ and $V_{j+1}$. It is shown in Fig. 13 and related by

$$V_{j+1} = V_j \oplus W_j$$ (49)

where $\oplus$ denotes the union of spaces. We can thus express the space of $L^2(R)$ as:

$$L^2(R) = V_0 \oplus W_0 \oplus W_1 \oplus \cdots$$ (50)

This equation can even be extended as following:

$$L^2(R) = \cdots \oplus W_{-2} \oplus W_{-1} \oplus W_0 \oplus W_1 \oplus W_2 \oplus \cdots$$ (51)

which eliminates the scaling function and represents a function in terms of wavelets.
alone. If a function \( f(x) \) is an element of \( V_1 \), an expansion contains an approximation of \( f(x) \) using \( V_0 \) and the wavelets from \( W_0 \) encode the difference between this approximation and the actual function. Generally, we start from an arbitrary scale \( j_0 \). Then the equation can be written as:

\[
L^2(R) = V_{j_0} \oplus W_{j_0} \oplus W_{j_0+1} \oplus \cdots
\]  

(52)

Moreover, any wavelet function can be expressed as a weighted sum of shifted, double-resolution scaling functions. That is, we can write

\[
\varphi(x) = \sum_n h_\varphi(n) \sqrt{2} \phi(2x - n)
\]  

(53)

where the \( h_\varphi(n) \) are called the wavelet function coefficients and \( h_\varphi \) is the wavelet vector. If the function being expanded is a sequence of numbers, like samples of a continuous function \( f(x) \), the resulting coefficients are called the discrete wavelet transform (DWT) of \( f(x) \). By applying the principle of series expansion, the DWT coefficients of \( f(x) \) are defined as

\[
W_\varphi(j_0, k) = \frac{1}{\sqrt{M}} \sum_x f(x) \phi_{j_0, k}(x)
\]  

(54)

\[
W_\varphi(j, k) = \frac{1}{\sqrt{M}} \sum_x f(x) \varphi_{j, k}(x)
\]  

(55)

for \( j \geq j_0 \) and the parameter \( M \) is a power of 2 which range from 0 to \( J - 1 \). The function \( f(x) \) can now expressed as

\[
f(x) = \frac{1}{\sqrt{M}} \sum_k W_\varphi(j_0, k) \phi_{j_0, k}(x) + \frac{1}{\sqrt{M}} \sum_{j=j_0}^{\infty} \sum_k W_\varphi(j, k) \varphi_{j, k}(x)
\]  

(56)

where \( \frac{1}{\sqrt{M}} \) acts as a normalizing factor.

### 5.3 Fast Wavelet Transform

The fast wavelet transform (FWT) is an efficient implementation of the discrete wavelet transform. By finding the relationship between the coefficients of the DWT at adjacent scales, we can reduce the compute complexity. The FWT is similar to the two-band subband coding we’ve seen in 2.2. consider the multiresolution refinement equation

\[
\phi(x) = \sum_n h_\phi(n) \sqrt{2} \phi(2x - n)
\]  

(57)

By a scaling of \( x \) by \( 2^j \), translation of \( x \) by \( k \) units, and letting \( m = 2k + n \), we
would get

\[ \phi(2^j x - k) = \sum_n h_{\phi}(n) \sqrt{2} \phi(2(2^j x - k) - n) \]
\[ = \sum_m h_{\phi}(m - 2k) \sqrt{2} \phi(2^{j+1} x - m) \] (58)

and similarly

\[ \varphi(2^j x - k) = \sum_m h_{\varphi}(m - 2k) \sqrt{2} \varphi(2^{j+1} x - m) \] (59)

Now consider the DWT coefficient functions \( W_{\varphi}(j, k) \). By changing variable we can get

\[ W_{\varphi}(j, k) = \frac{1}{\sqrt{M}} \sum_x f(x) 2^{j/2} \phi\left(2^j x - k\right) \] (60)

which, upon replacing \( \varphi(2^j x - k) \), it becomes

\[ W_{\varphi}(j, k) = \frac{1}{\sqrt{M}} \sum_x f(x) 2^{j/2} \left[ \sum_m h_{\varphi}(m - 2k) \sqrt{2} \varphi(2^{j+1} x - m) \right] \] (61)

rearranging the terms then gives

\[ W_{\varphi}(j, k) = \sum_m h_{\varphi}(m - 2k) \left[ \frac{1}{\sqrt{M}} \sum_x f(x) 2^{(j+1)/2} \sqrt{2} \varphi(2^{j+1} x - m) \right] \] (62)

where the bracketed quantity is identical to \( W_{\varphi}(j_0, k) \) with \( j_0 = j + 1 \). We can thus write

\[ W_{\varphi}(j, k) = \sum_m h_{\varphi}(m - 2k) W_{\varphi}(j + 1, k) \] (63)

similarly, the approximation coefficients is written by

\[ W_{\varphi}(j, k) = \sum_m h_{\varphi}(m - 2k) W_{\varphi}(j + 1, k) \] (64)

Fig. 14 the FWT analysis filter bank
We can find that both $W_\phi(j,k)$ and $W_\psi(j,k)$ can be obtained by convolving $W_\phi(j+1,k)$, the scale $j+1$ approximation coefficients, with time-reversed scaling and wavelet vectors $h_\phi(-n)$ and $h_\psi(-n)$, and downsampling the results by 2. It is a surprising and useful relationship. Fig. 14 illustrates the construction of FWT using filter bank.

Reconstruction of $f(x)$ can also be formulated by using the scaling and wavelet vectors employed in the forward transform. It is called the inverse fast wavelet transform (IFWT). To generate the level $j+1$ approximation coefficients, it is a little bit different to FWT. We first upsample the approximation and detail coefficients of level $j$ and then summing the result which are passed through the scaling and wavelet vector $h_\phi(n)$ and $h_\psi(n)$. This system is then depicted in Fig. 15. Note that the FWT analysis filter are $h_0(n) = h_\phi(-n)$ and $h_1(n) = h_\psi(-n)$, the required inverse FWT synthesis filters are $g_0(n) = h_\phi(n)$ and $g_1(n) = h_\psi(n)$.

### 5.4 Wavelet Transforms in two dimensions

The two-dimensional wavelet transforms are slightly different to one-dimensional ones. One can easily extend it by simply multiply the one-dimensional scaling and wavelet functions. Wavelet transform in two dimensions is used in image processing. For two-dimensional wavelet transform, we need one two-dimensional scaling function, $\phi(x,y)$, and three two-dimensional wavelet functions, $\varphi^H(x,y)$, $\varphi^V(x,y)$, and $\varphi^D(x,y)$. Each is the product of a one-dimensional scaling function $\phi$ and corresponding wavelet $\varphi$. These are shown as follows:

\[
\phi(x,y) = \phi(x)\phi(y) \tag{65}
\]

\[
\varphi^H(x,y) = \varphi(x)\phi(y) \tag{66}
\]

\[
\varphi^V(x,y) = \phi(x)\varphi(y) \tag{67}
\]
For image processing, these functions measure the variation of intensity for the image along different directions: $\varphi^H$ measures variations along columns, $\varphi^V$ measures variations along rows, and $\varphi^D$ measures variations along diagonals. The scaling function $\phi$ gives the approximation as same as the one-dimensional one. When the scaling function and wavelet functions are given, extension of the one-dimensional DWT to two-dimensions is straightforward. We first define the basis functions:

$$\phi_{j,m,n}(x,y) = 2^{j/2} \phi(2^j x - m, 2^j y - n)$$  \hspace{1cm} (69)

$$\varphi_{i,j,m,n}(x,y) = 2^{j/2} \varphi^i(2^j x - m, 2^j y - n), \quad i = \{H, V, D\}$$  \hspace{1cm} (70)

where the index $i$ defines the direction of the wavelet functions. The discrete wavelet transform of function $f(x,y)$ of size $M \times N$ is

$$W_\phi(j_0,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \phi_{j_0,m,n}(x,y)$$  \hspace{1cm} (71)

$$W_{\varphi^i}(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \varphi_{j,m,n}^i(x,y) \quad i = \{H, V, D\}$$  \hspace{1cm} (72)

Similarly, the variable $j_0$ is an arbitrary starting scale and $W_\phi(j_0,m,n)$ define the approximation of $f(x,y)$. Moreover, $f(x,y)$ can be obtained by two-dimensional inverse discrete wavelet transform defined by

$$f(x,y) = \frac{1}{\sqrt{MN}} \sum_{m} \sum_{n} W_\phi(j_0,m,n) \phi_{j_0,m,n}(x,y)$$

$$+ \frac{1}{\sqrt{MN}} \sum_{i=H,V,D} \sum_{j=j_0}^{\infty} \sum_{m} \sum_{n} W_{\varphi^i}(j,m,n) \varphi_{j,m,n}^i(x,y)$$  \hspace{1cm} (73)

### 5.5 Two-dimensional Fast Wavelet Transform

The two-dimensional wavelet transform also has its fast algorithm which is similar to the one-dimensional one. As mentioned in section 5.3, DWT can be implemented by using digital filters and downsamplers. With separable two-dimensional scaling and wavelet functions, we simply take the one-dimensional FWT of the rows of $f(x,y)$, followed by the one-dimensional FWT of the resulting columns. Fig. 16 shows the process in block diagram form. Like the one-dimensional FWT, the two-dimensional FWT is constructed of approximation and detail part. As we can see in Fig. 17, the

$$\varphi^D(x,y) = \varphi(x)\varphi(y)$$  \hspace{1cm} (68)
original image (matrix) can be decomposed into four subimages, which are \( W_\phi, \ W_\phi^H \), \( W_\psi \) and \( W_\psi^D \). We can again divide the scale \( j+1 \) approximation coefficients into four parts in smaller size. In other words, the \( j+1 \) approximation coefficients are constructed by the scale \( j \) approximation and detail coefficients. By the same idea in section 5.3, the two-dimensional IDWT reverse the processes described above. The reconstruction algorithm is similar to the one-dimensional one. At each iteration, four
scale $j$ approximation and detail coefficients are upsampled and convolved with two one-dimensional filters, one is for the subimages’ rows and the other is for its columns. Adding the results then we can obtain the $j+1$ approximation coefficients. By repeating the process, we can ultimately reconstruct the original image (matrix). Fig. 18 shows the synthesis filter bank of this operation.

6 Applications of Wavelet Transform

As mentioned, most of the applications of Wavelet Transform is about image processing such as image compression, edge detection, noise removal, etc. In last chapter, we see that images can be decomposed into four parts by two-dimensional Wavelet Transform. In fact, the decomposition can continue until the size of the sub-image is as small as you want. By setting some parts of its sub-images, we can reduce the quantity of information, in other words we can compress the image by setting the useless data. Edge detection and noise removal are based on the same idea. When we want to detect the edges of the image, we can simply set the diagonal sub-images to zero, then we can obtain the output image with edges clearly. Fig. 19 shows an example of edge detection. The black areas in the figure mean the coefficients we set to zero. The noise removal is the same, by abandoning the information not needed we can control the image to the way we want.

7 Comparison between Fourier transform, Windowed Fourier transform and Wavelet Transform

7.1 Resolution of FT, WFT and DWT

We start this chapter with a simple example we have discussed in section 5.1.3. Again, we see the Eq. (38). For the same signal $f(t)$ we use Fourier Transform, Windowed Fourier Transform and Discrete Wavelet Transform to analyze it. The window function $g(n)$ here we use is Hanning window. It is defined as:

$$g(n) = 0.5 \left(1 - \cos \left(\frac{2\pi n}{N-1}\right)\right)$$

(74)

where N is the maximum length of the window. And the mother wavelet of Discrete Wavelet Transform is Morlet wavelet. After these three kinds of transforms, we can obtain their intensity in time-frequency plane. The results are shown in Fig. 20, Fig.
It can be found that in Wavelet Transform and Windowed Fourier Transform we can obtain both frequency and time information, whereas in Fourier Transform we only have the frequency information in a specific time. This is the biggest difference between conventional Fourier Transform and modern time-frequency analysis technique. From Fig. 20, we can find that the size of Heisenberg box of Wavelet Transform is a function of frequency. When a

Fig. 19 example of edge detection using Discrete Wavelet Transform

Fig. 20 the result of Eq. (38) using Discrete Wavelet Transform

Fig. 21 the result of Eq. (38) using Windowed Fourier Transform
Fig. 22  the result of Eq. (38) using Fourier Transform

Fig. 23  Time-frequency tilings for Windowed Fourier Transform with different window size

Fig. 24  Time-frequency tilings for Wavelet Transform

Fig. 25  Time-frequency tilings for Fourier Transform

...
of Wavelet Transform are adjustable. By choosing proper mother wavelet (basis function), we can control the size of the time-frequency tilings in order to analyze the signals with the resolution we want. In addition to Wavelet Transform, the size of Windowed Fourier Transform is fixed as we saw in section 4.1. A wide window gives better frequency resolution but poor time resolution, and vice versa. This is depicted in Fig. 24. The Fourier Transform’s time-frequency tilings is depicted in Fig. 25. The time resolution of Fourier Transform is extremely small, that is, for Fourier Transform we only have the frequency information in a specific time.

7.2 Complexity of FT, WFT and DWT
We discuss the complexity of Fast Fourier Transform (FFT) first. It is an efficient algorithm for implementation of Discrete Fourier Transform. For a radix 2 Fast Fourier Transform (FFT), its complexity is \( O(N \log_2 N) \) which including \( (N/2) \log_2 N \) multiplication and \( N \log_2 N \) addition. Whereas, for a conventional DFT, \( N^2 \) operations are needed. Recall the Windowed Fourier Transform in Chapter 4, the WFT is a transform with a window \( g(t) \) being its basis. Similarly, we need a discretized window function \( g[n] \) for Discrete Windowed Fourier Transform (DWFT). Then the DWFT for an \( N \) periodic signal is

\[
S\{f[m,l]\} = \langle f, g_{m,l} \rangle = \sum_{n=0}^{N-1} f[n] g[n-m] \exp\left( -\frac{i2\pi nl}{N} \right) \tag{75}
\]

For each \( 0 \leq m < N \), \( Sf[m,l] \) is calculated for \( 0 \leq l < N \) with a DFT of \( f[n]g[n-m] \). The DFT calculation can be implemented by FFT. Therefore we need \( N \) FFT with size \( N \), and thus require a total of \( O(N^2 \log_2 N) \) operations.

Now, consider the one-dimensional DWT. Recall that the one-dimensional DWT can be implemented by filter banks we saw in Fig. 14. If the input signal length is \( N \) and the length of the filter is \( M \). Totally, we need one time FFT for the input signal, two times FFT for the two filters and two times FFT for the output signals. That is, we need a total of \( O(N \log_2 N) \) operations.
8 Conclusion

So far we’ve discussed the foundation of time-frequency analysis and multiresolution analysis. We introduced two of the most well-known transform of time-frequency analysis – Windowed Fourier transform and Wavelet transform. In the Windowed Fourier transform section, it shows the basic concept of time-frequency analysis. And we saw the advantage and disadvantage of Windowed Fourier transform and Wavelet transform. By deriving the Heisenberg boxes, we easily found the difference between them. Continuous Wavelet transform can also be discretized. It is basically based on continuous wavelet transform. Then we studied the implementation and the efficient algorithm of discrete Wavelet transform. Nowadays, discrete wavelet transform has become the most useful tool for signal processing and it still has many potentialities. For this reason, we should continue on developing more powerful tool or efficient algorithm in this area.

9 Reference


