# Wavelet Analysis for Image Processing

Tzu-Heng Henry Lee
Graduate Institute of Communication Engineering,
National Taiwan University, Taipei, Taiwan, ROC
E-mail: r96942133@ntu.edu.tw

## Abstract

Wavelet transforms have become increasingly important in image compression since wavelets allow both time and frequency analysis simultaneously. This paper investigates the fundamental concept behind the wavelet transform and provides an overview of some improved algorithms on the wavelet transform. The latter part of this paper emphasize on lifting scheme which is an improved technique based on the wavelet transform.

## 1. Introduction

The wavelet transform plays an extremely crucial role in image compression. For image compression applications, wavelet transform is a more suitable technique compared to the Fourier transform. Fourier transform is not practical for computing spectral information because it requires all previous and future information about the signal over the entire time domain and it cannot observe frequencies varying with time because the resulting function after Fourier transform is a function independent of time [1]. On the other hand, wavelet transforms are based on wavelets which are varying frequency in limited duration [2]. Due to the practicality of the wavelet transforms, this research paper is written to investigate the properties and the improvements that can be made to enhance the performance of the wavelet transforms. At the beginning of this paper, background information such as time-frequency analysis and multiresolution analysis are given. Techniques related to multiresolution theory are also briefly discussed. The mid-portion of the paper focuses on the wavelet transforms and their derivations for both one dimensional and two dimensional cases. Improved algorithms for the wavelet transforms including the fast wavelet transform, lifting scheme, and reversible integer wavelet transform are provided in the remainder of this paper. Lifting-based discrete wavelet transform has many advantages over the convolution-based transform and it can be combined with the concept of integer-to-integer transform in order to enhance the performance of lossless image compression.

# 2. Backgrounds on Time-Frequency Analysis and the Windowed Fourier Transform

A window function *w*(*t*) [1] is used to identify regions in time corresponding to the desired spectral characteristics at all frequencies. Multiplying a signal with a window function before taking FT can restrict the spectral info of the signal to the domain of influence of the window function. By translating a window function in time domain, we can cover all the information in entire time domain and analyze the spectral info in localized neighborhoods in time. The center t* and radius $\Delta_w$ of a window function w are defined as

$$t^* = \|w\|_2^{-2} \int_{-\infty}^{\infty} t \left|w(t)\right|^2 dt \qquad (1)$$

$$\Delta_w = \|w\|_2^{-1} \left[ \int_{-\infty}^{\infty} (t-t^*)^2 \left|w(t)\right|^2 dt \right]^{1/2} \qquad (2)$$

We assume that both $w$ and $\hat{w}$ ( $w$ is well-localized in time and $\hat{w}$ is well-localized in frequency) are window functions with rapid decay in time and frequency, respectively. The FT of any window function *w* with t*=0 is defined as

$$T^w(\tau, w)(f) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)\overline{w(t-\tau)} e^{-jwt} dt \qquad (3)$$

$2\Delta_w$ and $2\Delta_{\hat{w}}$ are the width (width of time window) and height (width of frequency window) of the window function respectively. The constant time-frequency area is $4\Delta_w\Delta_{\hat{w}}$. In order to achieve a localized time-frequency spectrum, narrow time and frequency windows are required. Therefore, a different windowed FT that can achieve varying time and frequency is needed. In order to realize different degrees of localization, the size of the time window must be changed. This can be done by reciprocally varying the size of the frequency window while keeping the area of the window constant. But there is a tradeoff between time and frequency localization [1].

## 2.1 Heisenberg’s Uncertainty Principle

Heisenberg’s Uncertainty Principle imposes a lower bound on the area of the time frequency window of any window function:

$$4\Delta_w\Delta_{\hat{w}} = 4\cdot\frac{\|xw\|_2}{\|w\|_2}\cdot\frac{\|\omega\hat{w}\|_2}{\|\hat{w}\|_2}\geq 2 \tag{4}$$

The principle suggests that a function's frequency component and the position of the component cannot both be measured to an arbitrary degree of precision simultaneously:

$$\frac{\|xf\|_2}{\|f\|_2}\cdot\frac{\|w\hat{f}\|_2}{\|\hat{f}\|_2}\geq\frac{1}{2}\quad\text{for } f\in C^{\infty}(R) \tag{5}$$

The windowed FT $T^{w}(\tau,w)(f)$ is called Gabor transform if the window function is a Gaussian. Gabor transform has a tight and rigid time-frequency window and it does not vary over time or frequency. A sinusoid's frequency is able to localize transient spectral information with high frequency to a narrow interval and it allows wider time interval [1].

## 3. Multiresolution Analysis

Multiresolution analysis (MRA) [3] is a very well-known and unique mathematical theory that incorporates and unifies various image processing techniques such as subband coding, pyramidal image processing, and quadrature mirror filtering. These techniques will be discussed in the following section. The main purpose of this analysis is to obtain different approximations of a function *f*(*x*) at different levels of resolution [2], [4]. Both the mother wavelet $\psi(x)$ and the scaling function $\varphi(x)$ are important functions to be considered in multiresolution analysis [4].

A linear combination of expansion functions [2] is often a better representation to express a signal function *f*(*x*):

$$f(x)=\sum_k \alpha_k\varphi_k(x) \tag{6}$$

The $\alpha_k$ and the $\varphi_k(x)$ in (6) are called real-valued expansion coefficients and real-valued expansion functions respectively. A unique $\varphi_k(x)$ is called a basis function for a unique expansion in which only one set of $\alpha_k$ exists for a particular *f*(*x*). The function space of the expansion set $\{\varphi_k(x)\}$ can be expressed as

$$V=\underset{k}{span}\{\varphi_k(x)\} \tag{7}$$

The expansion coefficients can be found by computing the inner product of the function *f*(*x*) and the dual function of $\varphi_k(x)$:

$$\alpha_k = \langle \tilde{\varphi}_k(x), f(x) \rangle = \int \tilde{\varphi}_k^*(x) f(x) dx \tag{8}$$

The fact that $\varphi_k(x) = \tilde{\varphi}_k(x)$ is true if the expansion functions form an orthonormal basis. On an orthonormal basis, the inner product is expressed as

$$\langle \varphi_j(x), \varphi_k(x) \rangle = \delta_{jk} = \begin{cases} 0 & , j \neq k \\ 1 & , j = k \end{cases} \tag{9}$$

The equation (8) can be rewritten as

$$\alpha_k = \langle \varphi_k(x), f(x) \rangle \tag{10}$$

In another occasion which the expansion functions are not orthonormal but are an orthogonal basis for *V*, then the basis functions and their duals are called biorthogonal. The biorthogonality can be shown as

$$\langle \varphi_j(x), \tilde{\varphi}_k(x) \rangle = \delta_{jk} = \begin{cases} 0 & , j \neq k \\ 1 & , j = k \end{cases} \tag{11}$$

## 3.1 Scaling Function

A scaling function [2] is used to approximate an image function at different level of approximations. Each approximation is differed by a factor of two from the approximation at the nearest neighboring level. Scaling functions are actually expansion functions which are composed of integer translations and binary scaling and contained in the set $\{\varphi_{j,k}(x)\}$. The general scaling functions are

$$\varphi_{j,k}(x) = 2^{j/2} \varphi(2^j x - k) \tag{12}$$

for both *j* and $k \in \mathbb{Z}$, and $\varphi(x) \in L^2(\mathbb{R})$. The parameter *k* and *j* determine the position of $\varphi_{j,k}(x)$ along the x-axis and the width of $\varphi_{j,k}(x)$ along the x-axis respectively. For a specific j, the subspace of an expansion set is commonly expressed as

$$V_j = \overline{\underset{k}{Span}\{\varphi_{j,k}(x)\}} \tag{13}$$

The parameter *j* is proportional to the size of $V_j$ [2]. There are four fundamental requirements [2] of multiresolution analysis that scaling functions 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:

$$V_{-\infty} \subset \cdots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset V_\infty . \tag{14}$$

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

$$V_{-\infty} = \{0\}. \tag{15}$$

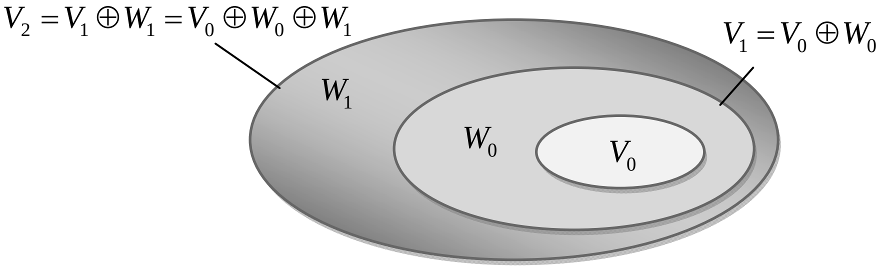


Fig. 1 The spatial relation of scaling and wavelet function spaces.

This is also called downward completeness property [4].

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.

$$V_\infty = \left\{ L^2 \left( \mathbf{R} \right) \right\} \tag{16}$$

This is also called upward completeness property [4]. With the above condition being satisfied, the weighted sum of the expansion functions of subspace $V_{j+1}$ can be used to express the expansion functions of subspace $V_j$ [2]:

$$\varphi_{j,k}(x) = \sum_n \alpha_n \varphi_{j+1,n}(x) \tag{17}$$

## 3.2 Wavelet Function

Wavelet function [2] is a type of scaling function that satisfied the four MRA requirements for scaling functions described in the previous subsection. The wavelet function is analogous to the scaling function expression in (12). Both integer translation and binary scaling are incorporated. The function $\psi(x)$ is defined as

$$\psi_{j,k}(x) = 2^{j/2} \psi(2^j x - k) \tag{18}$$

for all $k \in \mathbf{Z}$ that spans the space $W_j$ where

$$W_j = \underset{k}{span} \left\{ \psi_{j,k}(x) \right\} \tag{19}$$

The wavelet function spans the difference between any two adjacent scaling subspaces, $V_j$ and $V_{j+1}$ as depicted in

Fig. 1. Therefore, a general equation describing the relationship between the scaling and wavelet function spaces is derived:

$$V_{j+1} = V_j \oplus W_j \tag{20}$$

This expression can be further extended to express the space of all measurable, square-integrable functions:

$$L^2\left(\mathbf{R}\right) = V_0 \oplus W_0 \oplus W_1 \oplus ....... \qquad (21)$$

Equation (21) can be expressed without the scaling function space:

$$L^2\left(\mathbf{R}\right) = ..... \oplus W_{-2} \oplus W_{-1} \oplus W_0 \oplus W_1 \oplus W_2 \oplus ..... \qquad (22)$$

Fig. 1 also illustrates crucial information about the purpose of wavelet and scaling functions. The scaling function at the lowest resolution level $V_0$ provides an approximation to the actual function *f*(*x*) and wavelets from $W_0$ encodes the difference between this approximation and the actual function [2]. In fact, any wavelet function can be expressed as a weighted sum of shifted, double-resolution scaling functions [2]:

$$\psi(x) = \sum_n h_\psi(n)\sqrt{2}\phi(2x-n) \qquad (23)$$

where the $h_\psi(n)$ are called the wavelet function coefficients. The $h_\psi(n)$ can be related to $h_\phi(n)$ by

$$h_\psi(n) = (-1)^n h_\phi(1-n) \qquad (24)$$

# 4. MRA Related Techniques

## 4.1 Subband Coding

Subband coding [2] decomposes an image into a set of band-limited components, called subbands. The subbands can be reassembled to reconstruct the original image perfectly without error. Fig. 2 shows the components of two-band subband coding and decoding system. Reconstruction of the original signal is accomplished by upsampling, filtering, and summing the individual subbands as shown in the top block diagram of Fig. 2.

Here we use Z-transform to express the subband coding theory because Z-transform can easily handle changes in sampling rate. According to the *Z*-transform and its sampling theorem, we can express the output as

$$\begin{aligned}\hat{X}(z) &= \tfrac{1}{2}\left[H_0(z)G_0(z) + H_1(z)G_1(z)\right]X(z) \\ &\quad + \tfrac{1}{2}\left[H_0(-z)G_0(z) + H_1(-z)G_1(z)\right]X(-z)\end{aligned} \qquad (25)$$

where the second component, which contains the –z dependence, represents the

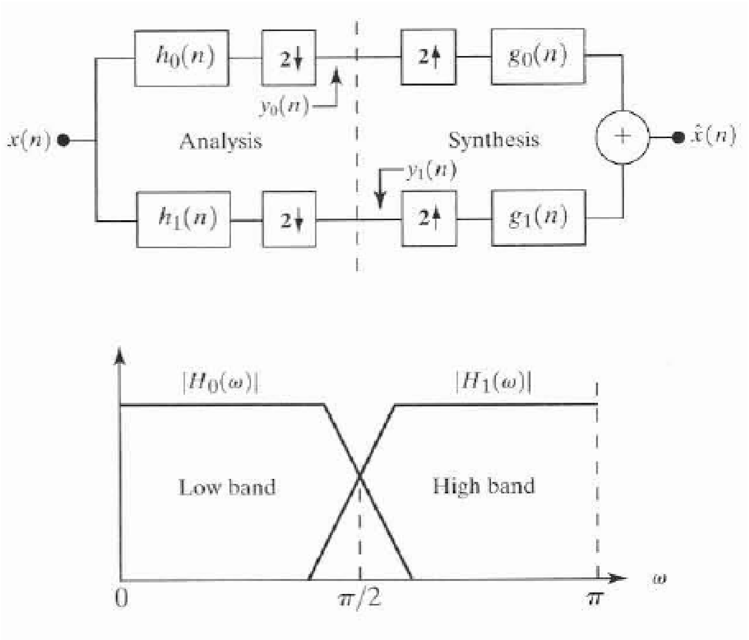


Fig. 2 A two-band filter bank for coding and decoding system and the corresponding spectrum.

aliasing introduced by the downsampling and upsampling processes.

The following conditions are required for perfect reconstruction of the input (i.e. $\hat{X}(z) = X(z)$):

$$\begin{aligned} H_0(-z)G_0(z) + H_1(-z)G_1(z) = 0 \\ H_0(z)G_0(z) + H_1(z)G_1(z) = 2 \end{aligned} \tag{26}$$

## 4.2 Pyramidal Image Processing

In early 1980s, a concept based on a pyramid structure was proposed. Image pyramid theory [2], [4] was actually developed earlier than the multiresolution analysis was formed. A pyramidal structure containing a collection of decreasing resolution images are depicted in Fig. 3. In that figure, the level *J* image has the highest resolution while the 0 level represents the lowest resolution approximation. For a fully populated pyramid, there are *J* + 1 levels from $2^J \times 2^J$ to $2^0 \times 2^0$ and the size of the image decreases as the resolution goes down. Fig. 4 shows the fundamental system that can be iterated to generate both prediction residual pyramid and approximation pyramid. In the figure, a level *J* input is first fed into an approximation filter followed by a

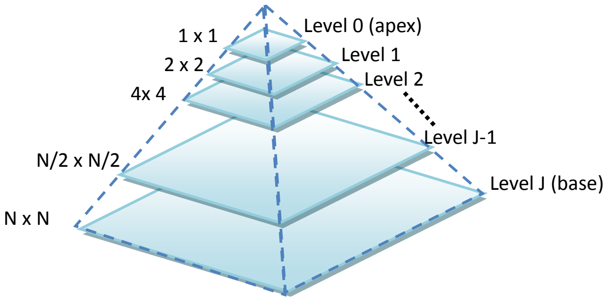


Fig. 3 A pyramidal structure of an image.

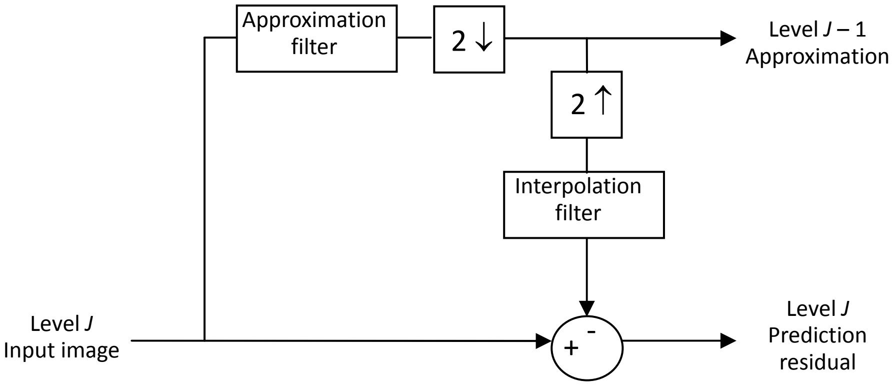


Fig. 4 System block diagram for building approximation and prediction residual pyramids.

downsampler. The result of the approximation filter is downsampled by a factor of 2. As a result, a level *J* – 1 approximation that is reduced in resolution is produced and it is also used to generate a prediction image using an upsampler and an interpolation filter. Finally, a prediction residual image is obtained by computing the difference of the level *J* input image and the prediction image [2].

## 5. The Integral Wavelet Transform

The general integral wavelet transform or continuous wavelet transform [1], [2], [4] is defined by

$$(W_\psi f)(a,b) = \int_{-\infty}^{\infty} f(x)\psi_{a,b}(x)dx \tag{27}$$

where the mother wavelet is

$$\psi_{a,b}(t) = |a|^{-1/2}\,\psi\left(\tfrac{t-b}{a}\right) \tag{28}$$

with $a, b \in \mathbb{R}$ and $\psi \in L^2(\mathbb{R})$. The continuous real variables $a$ and $b$ are concerned with the function's dilation (scaling) and translation respectively [4]. In (28), $\psi$ is called the wavelet function and $\psi_{a,b}$ are called wavelets. Suppose both the wavelet function $\psi$ and the FT $\hat{\psi}$ are window functions with centers *t** and *w**, and radii $\Delta_\psi$ and $\Delta_{\hat{\psi}}$, the wavelet transform can be written as

$$(W_\psi f)(a,b) = |a|^{-1/2} \int_{-\infty}^{\infty} f(t)\psi_{a,b}(t)dt \tag{29}$$

The $2a\Delta_\psi$ and $2a^{-1}\Delta_{\hat{\psi}}$ are the width (width of time window which is inversely proportional to the center frequency $a^{-1}\omega^*$) and height (width of frequency window which is directly proportional to the center frequency) of the time-frequency window respectively. The constant time-frequency area is $4\Delta_\psi\Delta_{\hat{\psi}}$ [1]. The admissibility criterion [1], [2], [4] must be satisfied in order to allow the wavelet transform to have invertible property:

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\xi)|^2}{|\xi|} d\xi < \infty \tag{30}$$

where $\xi$ is space and scale variable and $\Psi(\xi)$ is the Fourier transform of $\psi(x)$. Once the admissibility criterion is satisfied, the inverse transform can be constructed by

$$f(x) = \frac{1}{C_\psi} \int_{a=-\infty}^{+\infty} \int_{b=-\infty}^{+\infty} \frac{1}{|a|^2} W(a,b)\psi_{a,b}(x)dadb \tag{31}$$

# 6. The Discrete Wavelet Transform

It is necessary to express the continuous dilation and translation parameters $a$ and $b$ in terms of discrete values. A popular way to discretize $a$ and $b$ as suggested by Acharya and Ray [4] is expressing the parameters as

$$a = a_0^j, \qquad b = kb_0a_0^j \tag{32}$$

where the parameter *j* affect the scaling of the wavelet transform and *k* is related to the translation of the wavelet function. Since the translation distance varies with respect to the scale of the wavelet function, the parameter b in continuous domain must take the scaling factor into the account. Thus, $a_0^j$ is multiplied to $kb_0$ complete discretization of $b$. By substituting $a$ and $b$ into equation (28), the wavelet function can be represented as

$$\psi_{a,b}(x) = a_0^{-j/2}\psi(a_0^{-j}x - kb_0) \qquad (33)$$

According to the dyadic sampling method, the value of $a_0$ and $b_0$ are selected as 2 and 1 respectively. $a = 2^j$ and $b = k2^j$. Thus, the wavelet function [4] on an orthonormal basis is defined as

$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}x - k) \qquad (34)$$

The discrete wavelet coefficients [2] can be obtained by expanding the function *f*(*x*) as a sequence of numbers. Note that the wavelet function is analogous to the function shown in (18) with the negative signs of the dilation parameter *j*. By applying the principle of series expansion, the discrete wavelet transform coefficients are defined as

$$W_\varphi(j_0,k) = \frac{1}{\sqrt{M}}\sum_{x=0}^{M-1} f(x)\varphi_{j_0,k}(x) \qquad (35)$$

$$W_\psi(j,k) = \frac{1}{\sqrt{M}}\sum_{x=0}^{M-1} f(x)\psi_{j,k}(x) \qquad (36)$$

for $j \geq j_0$ and the $W_\phi(j_0,k)$ and $W_\psi(j,k)$ are the approximation coefficient and detail coefficient respectively. The parameter *M* is a power of 2 which ranges from 0 to *J - 1*. The DWT coefficients enable us to reconstruct the signal function *f*(*x*) as

$$f(x) = \frac{1}{\sqrt{M}}\sum_k W_\varphi(j_0,k)\varphi_{j_0,k}(x) + \frac{1}{\sqrt{M}}\sum_{j=j_0}^{\infty}\sum_k W_\psi(j,k)\psi_{j,k}(x) \qquad (37)$$

where $\frac{1}{\sqrt{M}}$ acts as a normalizing factor [2]. The reason that the discrete wavelet transform is a better transform than Fourier transform is because DWT have a better ability in localizing both time and frequency. This makes the image compression easier to manipulate [4].

## 7. The Fast Wavelet Transform

An algorithm called the fast wavelet transform (FWT) [2] has been developed by Mallat in order to achieve fast and efficient implementation of the discrete wavelet transform. The FWT is similar to the two-band subband coding scheme which is also based on the relationship between the coefficients of the DWT at adjacent scales.

We first consider the multiresolution equation

$$\varphi(x)=\sum_n h_\varphi(n)\sqrt{2}\varphi(2x-n) \tag{38}$$

By a scaling of *x* by $2^j$, translation of *x* by *k* units, and making *m* = 2*k* + *n*, we would get

$$\begin{aligned}\varphi(2^j x-k)&=\sum_n h_\varphi(n)\sqrt{2}\varphi\left(2(2^j x-k)-n\right)\\&=\sum_m h_\varphi(m-2k)\sqrt{2}\varphi(2^{j+1}x-m)\end{aligned} \tag{39}$$

and analogously,

$$\psi(2^j x-k)=\sum_m h_\psi(m-2k)\sqrt{2}\varphi(2^{j+1}x-m) \tag{40}$$

A property that involves the convolution of a scaling function and a wavelet coefficient can be derived by the following steps:

1. We begin by considering the definition of the discrete wavelet transform as shown in equation (35) and (36).
2. By substituting equation (12) into (36), we get

$$W_\psi(j,k)=\frac{1}{\sqrt{M}}\sum_x f(x)2^{j/2}\psi(2^j x-k) \tag{41}$$

3. Replace $\psi(2^j x-k)$ with the right side of equation (40):

$$W_\psi(j,k)=\frac{1}{\sqrt{M}}\sum_x f(x)2^{j/2}\left[\sum_m h_\psi(m-2k)\sqrt{2}\varphi(2^{j+1}x-m)\right] \tag{42}$$

4. Rearrange the summation part of the equation:

$$W_\psi(j,k)=\sum_m h_\psi(m-2k)\left[\frac{1}{\sqrt{M}}\sum_x f(x)2^{(j+1)/2}\varphi(2^{j+1}x-m)\right] \tag{43}$$

where the bracketed quantity is identical to Eq. (41) with $j_0=j+1$.

Therefore,

$$W_\psi(j,k)=\sum_m h_\psi(m-2k)W_\varphi(j+1,m) \tag{44}$$

and similarly the DWT approximation coefficient at scale *j + 1* can be expressed as

$$W_\varphi(j,k)=\sum_m h_\varphi(m-2k)W_\varphi(j+1,m) \tag{45}$$

Equation (44) and (45) demonstrate that both the approximation and the detail coefficients $W_\varphi(j,k)$ and $W_\psi(j,k)$ at scale *j* can be obtained by convolving $W_\varphi(j+1,k)$, approximation coefficients at the scale *j* + 1, with the time-reversed scaling and wavelet vectors, $h_{\varphi}(-n)$ and $h_{\psi}(-n)$ followed by the subsequent subsampling. The equation (44) and (45) can then be expressed in the following convolution formats:

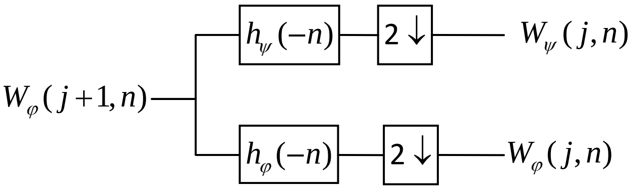


Fig. 5 A forward fast wavelet transform analysis filter bank.

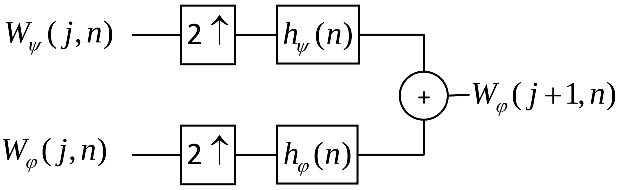


Fig. 6 A synthesis filter bank of the inverse fast wavelet transform.

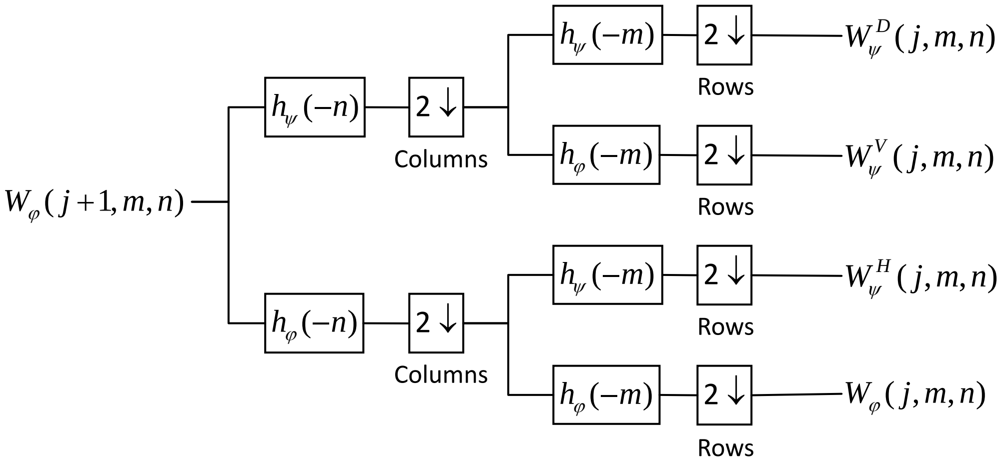


Fig. 7 The analysis filter bank of the two-dimensional FWT.

$$W_{\psi}(j,k)=h_{\psi}(-n)*W_{\varphi}(j+1,n)\Big|_{n=2k,k\geq 0} \tag{46}$$

$$W_{\varphi}(j,k)=h_{\varphi}(-n)*W_{\varphi}(j+1,n)\Big|_{n=2k,k\geq 0} \tag{47}$$

The above equations are illustrated in Fig. 5.

Reconstruction of the original *f*(*x*) can be done by the inverse fast wavelet transform which employs the scaling and wavelet vectors that are used in the forward fast wavelet transform with the level *j* approximation and detail coefficients.

This synthesis process generates the level $j + 1$ approximation coefficients. The synthesis filter bank is depicted in Fig. 6. 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

$g_0(n) = h_0(-n) = h_\phi(n)$ and $g_1(n) = h_1(-n) = h_\psi(n)$ [2].

## 8. Wavelet Transforms in Two Dimensions

A two-dimensional scaling function, $\varphi(x, y)$, and three two-dimensional wavelet $\psi^H(x, y)$, $\psi^V(x, y)$ and $\psi^D(x, y)$ are critical elements for wavelet transforms in two dimensions [2]. These scaling function and directional wavelets are composed of the product of a one-dimensional scaling function $\varphi$ and corresponding wavelet $\psi$ which are demonstrated as the following:

$$\varphi(x, y) = \varphi(x)\varphi(y) \tag{48}$$

$$\psi^H(x, y) = \psi(x)\varphi(y) \tag{49}$$

$$\psi^V(x, y) = \varphi(y)\psi(x) \tag{50}$$

$$\psi^D(x, y) = \psi(x)\psi(y) \tag{51}$$

where $\psi^H$ measures the horizontal variations (horizontal edges), $\psi^V$ corresponds to the vertical variations (vertical edges), and $\psi^D$ detects the variations along the diagonal directions.

The two-dimensional DWT can be implemented using digital filters and downsamplers. The block diagram in Fig. 7 shows the process of taking the one-dimensional FWT of the rows of *f* **(***x***,** *y*) and the subsequent one-dimensional FWT of the resulting columns. Three sets of detail coefficients including the horizontal, vertical, and diagonal details are produced. By iterating the single-scale filter bank process, multi-scale filter bank can be generated. This is achieved by tying the approximation output to the input of another filter bank to produce an arbitrary scale transform. For the one-dimensional case, an image *f* (*x*, *y*) is used as the first scale input. The resulting outputs are four quarter-size subimages: $W_\phi$, $W_\psi^H$, $W_\psi^V$, and $W_\psi^D$ which are shown in the center quad-image in Fig. 8. Two iterations of the filtering process produce the two-scale decomposition at the right of Fig. 8. Fig. 9 shows the synthesis filter bank that is exactly the reverse of the forward decomposition process.

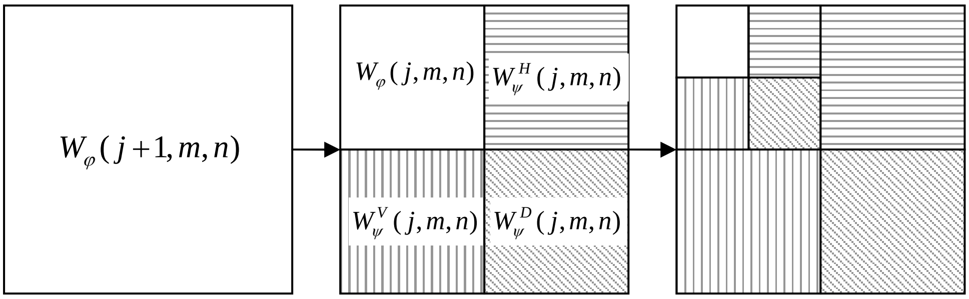


Fig. 8 A two-level decomposition of the two-dimensional FWT.

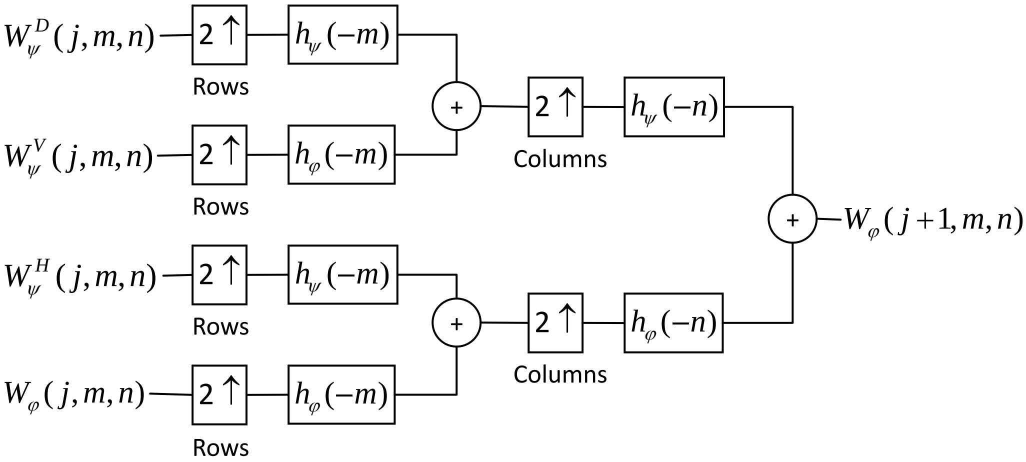


Fig. 9 The synthesis filter bank of the two-dimensional FWT.

# 9. Lifting Scheme

Direct discrete wavelet transform implementation is theoretical invertible. However, due to the finite register length of the computer system, inversion errors could occur and it would result in unsuccessful image reconstruction. In practical cases, the wavelet coefficients will be rounded to the nearest integer in the discrete transformation stage. This makes the lossless compression impossible [5]. An improved implementation called lifting-based wavelet transform which is based on the wavelet theory is proposed and it requires significantly fewer arithmetic computations and memory compared to the convolution based discrete wavelet transform. The lifting-based DWT scheme breaks up the high-pass and low-pass wavelet filters into a sequence of smaller filters. These decomposed filters are then converted into a sequence of upper and lower triangular filters [4]. The derivations of the triangular matrices by lifting factorization are presented in this section. Before we move onto the lifting scheme, there are two essential stages that must be performed:

1. Representing a finite impulse response (FIR) in Laurent polynomial with Z-transform and
2. Factorizing the Laurent polynomial using Euclidean algorithm.

## 9.1 FIR representation in Laurent Polynomials

In the lifting scheme, the finite impulse response filters (FIR) *h* and *g* are expressed in Laurent polynomial with the aid of Z-transform. For instance, the Laurent polynomial representation of filter *h* with Z-transform can be defined as

$$h(z) = \sum_{i=m}^{n} h_i z^{-i} \tag{52}$$

where $m$ and $n$ are positive integers and $\{h_i \in \mathbb{R} \mid i \in \mathbb{Z}\}$ [6]. The degree of the Laurent polynomial $h$ can be found by $|h| = n - m$ and the length of the filter is $|h| + 1$ [4]. Suppose there are two non-zero Laurent polynomials $a(z)$ and $b(z)$ with $|a(z)| \geq |b(z)|$. The product and quotient of these two Laurent polynomials will produce a Laurent polynomials with a degree of $|a(z)| + |b(z)|$ and $|a(z)| - |b(z)|$ respectively. Since the exact division of Laurent polynomials cannot be achieved, the remainder Laurent polynomial $r(z)$ has a degree of $|r(z)| < |b(z)|$. Meanwhile, The resulting polynomial degree will not change if an addition or subtraction operation is performed on the two Laurent polynomials. We can express the $a(z)$ as

$$a(z) = b(z)q(z) + r(z) \tag{53}$$

where the Laurent polynomial for quotient is denoted as $q(z)$ [6].

## 9.2 Polyphase Representation of Filters

We first consider the FIR-based discrete wavelet transform. The input image is fed into a low-pass filter $\tilde{h}$ and a high-pass filter $\tilde{g}$ separately. The outputs of the two filters are then subsampled. The resulting low-pass subband $y_L$ and high-pass subband $y_H$ are shown in Fig. 10. The original signal can be reconstructed by synthesis filters *h* and *g* which take the upsampled $y_L$ and $y_H$ as inputs [4].

In order to achieve perfect reconstruction of a signal, the filters shown in Fig. 10 must satisfy the following conditions:

$$\begin{cases} h(z)\tilde{h}(z^{-1}) + g(z)\tilde{g}(z^{-1}) = 2 \\ h(z)\tilde{h}(-z^{-1}) + g(z)\tilde{g}(-z^{-1}) = 0 \end{cases} \tag{54}$$

The analysis and synthesis filters as shown in Fig. 10 are further decomposed into the

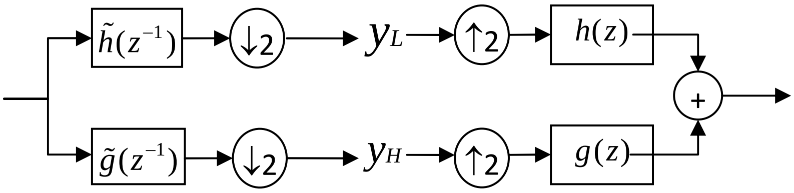


Fig. 10 DWT analysis and synthesis system [6].

polyphase representations which are expressed as

$$h(z) = h_e(z^2) + z^{-1}h_o(z^2) \tag{55}$$

$$g(z) = g_e(z^2) + z^{-1}g_o(z^2) \tag{56}$$

$$\tilde{h}(z) = \tilde{h}_e(z^2) + z^{-1}\tilde{h}_o(z^2) \tag{57}$$

$$\tilde{g}(z) = \tilde{g}_e(z^2) + z^{-1}\tilde{g}_o(z^2) \tag{58}$$

The $h_e$ and $h_o$ in Z-transform can be expressed as

$$\begin{cases} h_e(z) = \sum_k h_{2k} z^{-k} \\ h_o(z) = \sum_k h_{2k+1} z^{-k} \end{cases} \tag{59}$$

The two polyphase matrices of the filter $h$ is defined as

$$P(z) = \begin{bmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{bmatrix}, \qquad \tilde{P}(z) = \begin{bmatrix} \tilde{h}_e(z) & \tilde{g}_e(z) \\ \tilde{h}_o(z) & \tilde{g}_o(z) \end{bmatrix} \tag{60}$$

The parameter $z$ is used since the polyphase representations are derived using Z-transform and the subscript $e$ and $o$ denote the even and odd sub-components of the filters which are split into subsequences. The purpose of the polyphase representation is to reduce the computation time. The perfect reconstruction is ensured only when the following relation is true [4], [6], [5]:

$$P(z)\tilde{P}(z^{-1})^T = I \tag{61}$$

where $I$ is a 2 by 2 identity matrix. Now the wavelet transform can be expressed using the polyphase matrix for forward discrete wavelet transform [4], [6]

$$\begin{bmatrix} y_L(z) \\ y_H(z) \end{bmatrix} = \tilde{P}(z) \begin{bmatrix} x_e(z) \\ z^{-1}x_o(z) \end{bmatrix} \tag{62}$$

and the inverse discrete wavelet transform [4]

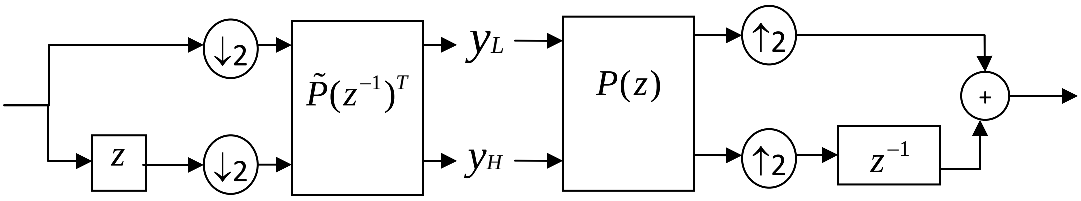


Fig. 11 Polyphase representation of wavelet transform [6].

$$\begin{bmatrix} x_e(z) \\ z^{-1}x_o(z) \end{bmatrix} = P(z)\begin{bmatrix} y_L(z) \\ y_H(z) \end{bmatrix} \tag{63}$$

Fig. 10 can then be schematically redrawn as shown in Fig. 11 based on the equation (62) and (63). Finally, the upper and lower triangular matrices can be obtained by the lifting factorization process. The lifting sequences are generated by employing Euclidean algorithm which factorizes the polyphase matrix for a filter pair [4], [6].

## 9.3 The Lifting Scheme

### 9.3.1 Primal Lifting Scheme

By definition, if the wavelet filter pair *h* and *g* are complementary, then any other finite filter $g^{new}$ [6] that is complementary to *h* can be expressed as

$$g^{new}(z) = g(z) + h(z)s(z^2) \tag{64}$$

where $s(z^2)$ is a Laurent polynomial. With the aid of polyphase decomposition explained in the equations (55)~(58), the above equation can be proved by expanding the polyphase representation which contains the even and odd components:

$$\begin{aligned} g^{new}(z) &= g(z) + h(z)s(z^2) \\ &= \left\{g_e(z^2) + z^{-1}g_o(z^2)\right\} + \left\{h_e(z^2) + z^{-1}h_o(z^2)\right\}s(z^2) \\ &= \left\{g_e(z^2) + h_e(z^2)s(z^2)\right\} + z^{-1}\left\{g_o(z^2) + h_o(z^2)s(z^2)\right\} \end{aligned} \tag{65}$$

Similar to the polyphase representation shown in (60), a new polyphase matrix [6] can be formulated as

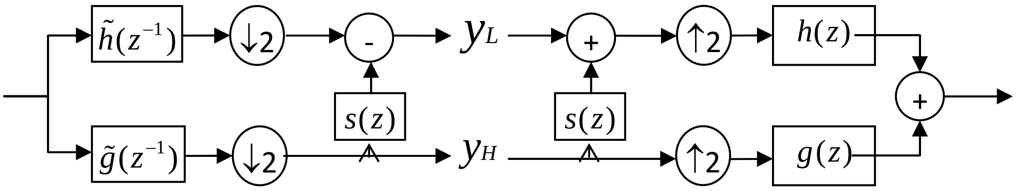


Fig. 12 The primal lifting scheme [6].

$$\begin{aligned} P^{new}(z) &= \begin{bmatrix} h_e(z) & g_e(z)+h_e(z)s(z) \\ h_o(z) & g_o(z)+h_o(z)s(z) \end{bmatrix} \\ &= \begin{bmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{bmatrix} \begin{bmatrix} 1 & s(z) \\ 0 & 1 \end{bmatrix} \\ &= P(z) \begin{bmatrix} 1 & s(z) \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{66}$$

The dual polyphase matrix of $P^{new}$ is derived as

$$\tilde{P}^{new}(z) = \tilde{P}(z) \begin{bmatrix} 1 & 0 \\ -s(z^{-1}) & 1 \end{bmatrix} \tag{67}$$

A new low-pass filter created by primal lifting is given by

$$\tilde{h}^{new}(z) = \tilde{h}(z) - \tilde{g}(z)s(z^{-2}) \tag{68}$$

Fig. 12 depicts the conventional low-pass and high-pass subband filter scheme and lifting of low-pass subband with the aid of high-pass subband.

### 9.3.2 Dual Lifting Scheme

Again, we suppose that $h$ and $g$ are complementary. Any other FIR $h^{new}$ that is complementary to $g$ can be expressed as

$$h^{new}(z) = h(z) + g(z)t(z^2) \tag{69}$$

and the polyphase matrix is given by

$$P^{new}(z) = P(z) \begin{bmatrix} 1 & 0 \\ t(z) & 1 \end{bmatrix} \tag{70}$$

A new high-pass filter created by dual lifting [4], [6] is given by

$$\tilde{g}^{new}(z) = \tilde{g}(z) - \tilde{h}(z)t(z^{-2}) \tag{71}$$

A schematic representation of dual lifting scheme is shown in Fig. 13. The figure demonstrates the conventional low-pass and high-pass subband filter scheme and lifting of high-pass subband with the aid of low-pass subband.

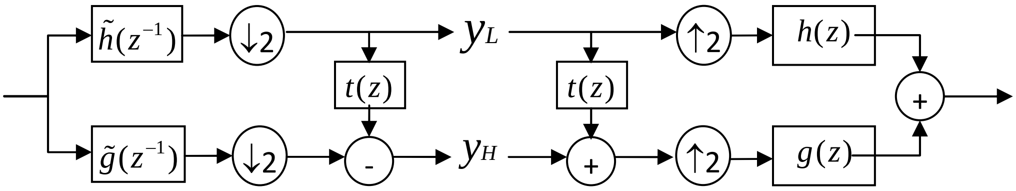


Fig. 13 The dual lifting scheme [6].

## 9.4 Euclidean Algorithm for Laurent Polynomials

The Euclidean algorithm [6], [7] is an important tool that can be used to find the greatest common divisor of two Laurent polynomials. Suppose we have two non-zero Laurent polynomials $a(z)$ and $b(z)$. By making $a_0(z) = a(z)$ and $b_0(z) = b(z)$, we can formulate the Euclidean algorithm as the following steps starting with $i = 0$:

$$a_{i+1}(z) = b_i(z) \tag{72}$$

$$b_{i+1}(z) = a_i(z)\% b_i(z) \tag{73}$$

By iterating the above steps, we would get

$$\begin{bmatrix} a_{i+1}(z) \\ b_{i+1}(z) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & -q_i(z) \end{bmatrix} \begin{bmatrix} a_i(z) \\ b_i(z) \end{bmatrix} \tag{74}$$

The above equation can be rearranged as

$$\begin{bmatrix} a_i(z) \\ b_i(z) \end{bmatrix} = \begin{bmatrix} q_i(z) & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} a_{i+1}(z) \\ b_{i+1}(z) \end{bmatrix} \tag{75}$$

As a result of iteration,

$$\begin{bmatrix} a(z) \\ b(z) \end{bmatrix} = \prod_{i=1}^{n} \begin{bmatrix} q_i(z) & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} a_n(z) \\ 0 \end{bmatrix} \tag{76}$$

where the $a_n$ is denoted as the greatest common denominator of $a(z)$ and $b(z)$. The $n$ is the smallest number for which $b_n(z)$ = 0. The equation (76) will be used to factor lifting sequences as discussed in the next section.

## 9.5 Lifting Factorization

With the basic idea of the Euclidean algorithm being accounted for, we can now apply this algorithm to factor any pair of complementary filters ($h$, $g$) into lifting steps. The greatest common divisor of the even and odd component $h_e$ and $h_o$ of filter $h$ can be computed using the Euclidean algorithm. As a result, the $h_e$ and $h_o$ can be expressed as

$$\begin{bmatrix} h_e(z) \\ h_o(z) \end{bmatrix} = \prod_{i=1}^{n} \begin{bmatrix} q_i(z) & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} K \\ 0 \end{bmatrix} \tag{77}$$

where $K$ is the greatest common divisor of $h_e$ and $h_o$. A complementary filter $g^{new}$ can always be found from a filter $h$ such that

$$P^{new}(z) = \begin{bmatrix} h_e(z) & g_e^{new}(z) \\ h_o(z) & g_o^{new}(z) \end{bmatrix} = \prod_{i=1}^{n} \begin{bmatrix} q_i(z) & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & 1/K \end{bmatrix} \tag{78}$$

Since

$$\begin{bmatrix} q_{2i-1}(z) & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & q_{2i-1}(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{79}$$

and

$$\begin{bmatrix} q_{2i}(z) & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ q_{2i}(z) & 1 \end{bmatrix} \tag{80}$$

we can rewrite (78) as

$$P^{new}(z) = \prod_{i=1}^{n/2} \begin{bmatrix} 1 & q_{2i-1}(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ q_{2i}(z) & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & 1/K \end{bmatrix} \tag{81}$$

A general formula for generating a lifted filter $g$ by lifting $g^{new}$ is

$$P(z) = P^{new}(z) \begin{bmatrix} 1 & s(z) \\ 0 & 1 \end{bmatrix} \tag{82}$$

## 9.6 Scaled Lifting Scheme

Recently a generalization of the improved lifting scheme [8], [9] has been developed to reduce the number of lifting matrices that is used to decompose the original matrix while the reversibility is preserved through proper reversible quantization. Suppose we have an input to output relation:

$$y = Ax, \quad y = \begin{bmatrix} y[1] \\ y[2] \end{bmatrix}, \quad x = \begin{bmatrix} x[1] \\ x[2] \end{bmatrix}, \quad A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \tag{83}$$

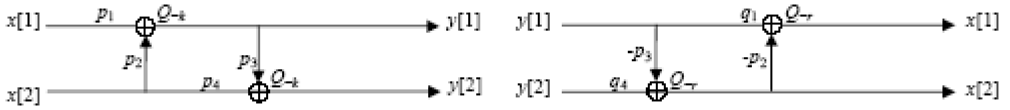
The lifting scheme can be applied to convert this relation into a reversible integer transform. The 2×2 matrix $A$ can be decomposed as

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ c/a & d-bc/a \end{bmatrix} \begin{bmatrix} a & b \\ 0 & 1 \end{bmatrix}, \quad a \neq 0 \tag{84}$$

Thus, the process of the forward integer transforms corresponding to matrix *A* can be expressed as

$$y[1] = Q_{-k}\left[p_1 x[1] + p_2 x[2]\right] \tag{85}$$

$$y[2] = Q_{-k}\left[p_3 y[1] + p_4 x[2]\right] \tag{86}$$



(a) (b)

Fig. 14 Visual representation of processes of the (a) forward and (b) inverse integer transforms.

where

$$p_1 = Q_{-c1}(a), \quad p_2 = Q_{-c2}(b),$$
$$p_3 = Q_{-c3}(c/a), \quad p_4 = Q_{-c4}(d - bc/a) \tag{87}$$

and $c_1$, $c_2$, $c_3$, and $c_4$ are some integers. The function $Q_{-k}$ represents the truncation operation which truncates the bits smaller than $2^{-k}$ and the parameter *k* determines the size of truncate bit (i.e., $Q_{-k}[z] = 2^{-k}\, round(2^k z)$ ). Similarly, the process of the inverse integer transform can recover the input x[1] and x[2] from the output y[1] and y[2]. This is shown as the following:

$$x[2] = Q_{-r}\left[q_4(y[2] - p_3 y[1])\right] \tag{88}$$

$$x[1] = Q_{-r}\left[q_1(y[1] - p_2 x[2])\right] \tag{89}$$

where

$$q_1 = Q_{-d1}(p_1^{-1}), \quad q_4 = Q_{-d4}(p_4^{-1}) \tag{90}$$

and $d_1$ and $d_4$ are some integers. The approximated constraints for the reversibility in equation (88) and (89) are defined as

$$a > 2^{r-k}, \quad d - bc/a > 2^{r-k} \tag{91}$$

The processes of forward and inverse integer transforms as demonstrated in equation (85), (86), (88), and (89) are depicted in Fig. 14. Normally, the original lifting scheme decomposes a 2×2 matrix into 3 matrices and it requires the diagonal entries to be 1. Compared to the original lifting scheme, the scaled lifting scheme can decompose a 2×2 matrix into 2 matrices which is shown in equation (84) and it does not constrain all the diagonal entries to be 1 [9].

Some major advantages [9] of the scaled lifting scheme are summarized in the following points:

1. The scaled lifting scheme has **better flexibility for determinant** because the determinant of matrix *A* shown in equation (83) is not required to be 1. The det(*A*) can be any non-zero value instead.
2. The conventional lifting scheme requires 3 stages for implementation while the scaled lifting scheme only requires 2. The reduction in implementation stage means **fewer time cycles** or **less computation time** is required for implementation.
3. The scaled lifting scheme can achieve **higher accuracy** because it requires **fewer times of quantization operation**.

According to [9], both quantization of matrix *A* and the rounding during the truncation operations can lead to approximation errors. The error which is represented in the normalized mean square error (NMSE) is still less than that produced by the original lifting scheme.

# 10. Reversible Integer Wavelet Transforms

An in-depth study on reversible integer wavelet transform was not conducted in this research paper. Therefore, this section will only provide some basic introduction on reversible integer wavelet transform method.

Wavelet transforms are more widely used in the applications of lossy compression due to the fact that most wavelet transforms generate float-point coefficients that are not very suitable for lossless image compression. As a result, integer-to-integer wavelet transforms have been introduced since it is more practical for lossless image coding [10]. Reversible integer wavelet transform is also useful for compression systems that require minimal memory usage and low computational complexity [11]. The initial stage of the derivation of reversible integer wavelet transform involves splitting the input x[n] into even and odd indexed samples. The even indexed samples are defined as

$$s_0[n] \triangleq x[2n] \tag{92}$$

while the odd indexed samples are expressed as

$$d_0[n] \triangleq x[2n+1] \tag{93}$$

The *M* alternating dual lifting and primal lifting steps are then applied so that even samples $s_M[n]$ becomes the low-pass coefficients $s[n]$ after multiplying to a scaling factor *K* and the odd samples $d_M[n]$ becomes the high-pass coefficients $d[n]$ after multiplying to a scaling factor $1/K$. Various types of forward reversible integer transforms are displayed in Table II of [11].

## 11. Conclusions

We have discussed the implementation and theoretical foundation of the time-frequency analysis and the multiresolution analysis. Techniques based on or related to the multiresolution theory such as subband coding and pyramid algorithms are introduced. The mathematic foundation and derivations of the wavelet transforms are also covered starting from the integral wavelet transform. Based on the continuous wavelet transform, the discrete wavelet transform can be derived. Subsequent studies on the fast wavelet transform improved the discrete wavelet transform based on the multiresolution theory and made implementation of the transform feasible using convolution. The discrete wavelet transform can also be extended to two dimensional cases. A relatively new and efficient implementation called lifting-based wavelet transform has been developed and become a very important technique in image compression. The underlying theory and implementation of lifting algorithm is discussed. Future researches should continue on developing more effective transforms.

$s_i(z)$

## 12. References

[1] L. Prasad and S. S. Iyengar, *Wavelet Analysis with Applications to Image Processing*. Boca Raton, FL: CRC Press LLC, 1997, pp.101-115.

[2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing 2/E*. Upper Saddle River, NJ: Prentice-Hall, 2002, pp. 349-404.

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

[4] T. Acharya and A. K. Ray, *Image Processing: Principles and Applications*.

Hoboken, NJ: John Wiley & Sons, 2005, pp. 79-104.

[5] E. Kofidisi, N.Kolokotronis, A. Vassilarakou, S. Theodoridis and D. Cavouras, "Wavelet-based medical image compression," *Future Generation Computer Systems,* vol. 15, no. 2, pp. 223-243, Mar. 1999.

[6] I. Daubechies and W. Sweldens, “Factoring wavelet transforms into lifting steps,” *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp.247-269, May 1998.

[7] W. Sweldens, “The lifting scheme: A new philosophy in biorthogonal wavelet constructions,” *Proc. SPIE*, vol. 2569, pp.68-79, Sept. 1995.

[8] S. C. Pei and J. J. Ding, “Improved reversible integer transform,” *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, pp. 4, 21-24 May 2006.

[9] S. C. Pei and J. J. Ding, "Scaled Lifting Scheme and Generalized Reversible Integer Transform," *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, pp.3203-3206, 27-30 May 2007.

[10] F. Sheng, A. Bilgin, P. J. Sementilli, and M. W. Marcellin, “Lossy and lossless image compression using reversible integer wavelet transforms,” *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, vol.3, no.4-7, pp.876-880, Oct. 1998.

[11] M. D. Adams, and F. Kossentni, "Reversible integer-to-integer wavelet transforms for image compression: performance evaluation and analysis," *Image Processing, IEEE Transactions on,* vol.9, no.6, pp.1010-1024, Jun 2000.