An Efficient Computational Scheme for the Two-Dimensional Overcomplete Wavelet Transform

Ngai-Fong Law, *Member, IEEE,* and Wan-Chi Siu, *Senior Member, IEEE*

***Abstract*—We have studied the computational complexity associated with the overcomplete wavelet transform for the commonly used Spline wavelet family. By deriving general expressions for the computational complexity using the conventional filtering implementation, we show that the inverse transform is significantly more costly in computation than the forward transform. To reduce this computational complexity, we propose a new spatial implementation based on the exploitation of the correlation between the lowpass and the bandpass outputs that is inherent in the overcomplete representation. Both theoretical studies and experimental findings show that the proposed spatial implementation can greatly simplify the computations associated with the inverse transform. In particular, the complexity of the inverse transform using the proposed implementation can be reduced to slightly less than that of the forward transform using the conventional filtering implementation. We also demonstrate that the proposed scheme allows the use of an arbitrary boundary extension method while maintaining the ease of the inverse transform.**

***Index Terms*—Computational complexity, over-complete wavelets, spatial implementation, wavelet transform.**

I. Introduction

POINTS of sharp variation such as edges and discontinuities in multiple scales are usually one of the most important features for analyzing properties associated with signals and images. It was conjectured that the basic representation (the primal raw sketch) furnished by the retinal system is a succession of contour sketches at scales that are in geometric progression [1]. The wavelet transform modulus maxima representation proposed by Mallat [2] provides such a multiscale contour representation of an image. This representation is obtained by retaining the local maxima of the continuous dyadic wavelet transform. It has been shown that the wavelet transform modulus maxima correspond to locations of discontinuities in an image. It thus provides a compact but meaningful description of an image. This representation has been used in various applications, including the image compression [3]–[5], edge and discontinuity characterization [6]–[10], contrast enhancement of medical images [11], artifact removal for image restoration [12], [13], texture characterization [14], and object recognition [15], [16].

Despite its ability to provide a meaningful representation, the main concern with the overcomplete representation is its computational complexity. Unlike the subsampling wavelet, where the computational time decreases with the number of decomposition levels [17], [18], the computational time increases linearly with the number of decomposition levels in the overcomplete case [19], [20]. Therefore, computational complexity becomes a major issue in its practical implementation. In addition, it is generally conceived that the inverse wavelet transform is computationally more expensive than the forward wavelet transform since the reconstruction filters are always longer than the forward filters in the Spline wavelet family [2]–[5].

In this paper, we provide an analysis of the computational complexity for the Spline wavelet family with an arbitrary order n and find that it is significantly higher for the inverse transform compared with the forward transform. In fact, it asymptotically approaches five times for a large n. In order to reduce the computations, we use the fact that the overcomplete wavelet transform provides a redundant representation of an image. This implies that a correlation exists between the lowpass and the bandpass outputs at a number of scales. Indeed, this correlation information has been explored in many applications ranging from discontinuity-preserving surface reconstruction, contrast enhancement, and denoising to artifact removal [6]–[14]. We propose to study the correlation between the lowpass and the bandpass outputs to reduce the computations, especially for the inverse transform.

By studying this correlation, a new spatial interpretation for the overcomplete wavelet transform is obtained, which greatly reduces the computational complexity associated with the inverse transform. In fact, the complexity of the inverse transform using the proposed spatial implementation turns out to be even slightly less than that of the forward transform using the filtering implementation. Besides the reduction in the computational complexity, this spatial interpretation provides us with a flexible and straightforward way of dealing with the boundary extension problem. Unlike the conventional filtering approach implementation, any boundary extension scheme can be used in the spatial implementation without complicating the inverse transform.

This paper is organized as follows. Section II describes the overcomplete wavelet transform. The computational complexities for both the forward and the inverse transforms are derived. The correlation between the lowpass and the bandpass outputs is then studied in Section III. By exploiting this correlation, an efficient spatial implementation structure is found. An analysis

Manuscript received June 11, 2001; revised June 18, 2002. This work was supported by the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University. The associate editor coordinating the review of this paper and approving it for publication was Prof. Paulo S. R. Diniz.

N.-F. Law is with the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (e-mail: ennflaw@polyu.edu.hk).

W.-C. Siu is with the Faculty of Engineering Department of Information Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (e-mail: enwcsiu@polyu.edu.hk).

Digital Object Identifier 10.1109/TSP.2002.804094.

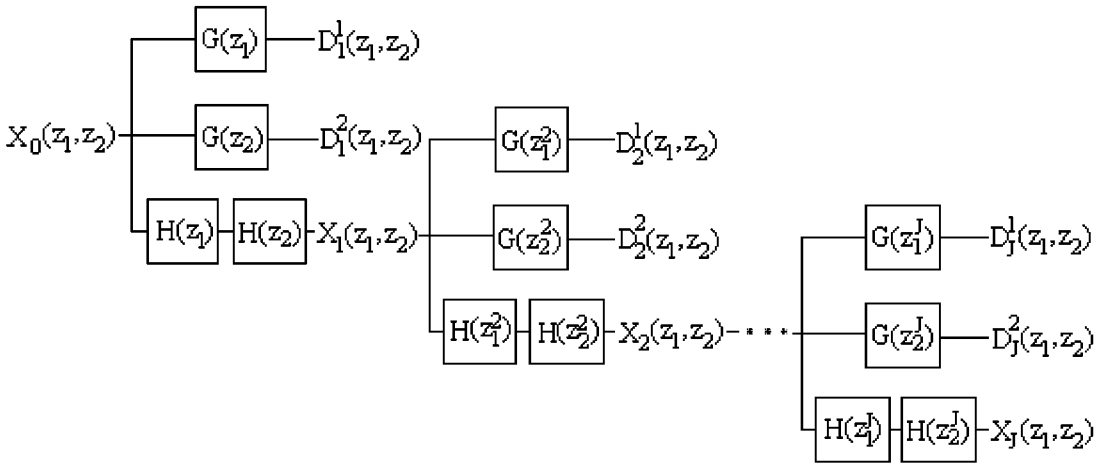


Fig. 1. jth-level forward overcomplete wavelet transform with $J = 2^{j-1}$.

of its computational complexity is also described in Section III. Section IV provides the design examples for commonly used low-order Spline wavelets using the new spatial implementation scheme. An analysis of the computational complexity associated with these low-order Spline wavelets is also provided.

Sections II–IV provide a theoretical analysis of both the conventional filtering implementation and the proposed spatial implementation. Section V consists of an experimental analysis of these two approaches. In particular, we compare their computational times using software implementations. Section VI then concludes the paper.

II. Overcomplete Wavelet Representation

An overcomplete wavelet representation for an image is obtained by applying filters to both the horizontal and the vertical directions [2]–[6]. There are three outputs from a single-level decomposition: the lowpass approximation of the original image and two bandpass outputs. One bandpass output shows the horizontal edges, whereas the other shows the vertical edges in the image. Mathematically, the lowpass output is given by

$$X_1(z_1, z_2) = H(z_1)H(z_2)X_0(z_1, z_2). \tag{1}$$

The two bandpass outputs are written, respectively, as

$$D_1^1(z_1, z_2) = G(z_1)X_0(z_1, z_2) \tag{2}$$
$$D_1^2(z_1, z_2) = G(z_2)X_0(z_1, z_2) \tag{3}$$

where $H(z)$, $G(z)$, and $X_0(z_1, z_2)$ denote, respectively, the one–dimensional (1-D) lowpass filter, the 1-D bandpass filter, and the original image. Fig. 1 shows a jth-level forward overcomplete wavelet transform. It can be seen that the lowpass and the bandpass filters are applied separately to the horizontal and the vertical directions. The lowpass output is obtained by applying the 1-D lowpass filter in both the horizontal and the vertical directions. The bandpass outputs are obtained by applying the bandpass filter in either the horizontal or the vertical directions. This is different from the subsampling scheme in which each subimage is associated with filters in both the horizontal and the vertical directions.

The jth-level inverse overcomplete wavelet transform is shown in Fig. 2. The original signal is reconstructed by

$$X_0(z_1, z_2) = K(z_1)L(z_2)D_1^1(z_1, z_2) + L(z_1)K(z_2) \times D_1^2(z_1, z_2) + \bar{H}(z_1)\bar{H}(z_2)X_1(z_1, z_2) \tag{4}$$

where $\bar{H}(z)$ is the time reverse of $H(z)$, and $K(z)$ and $L(z)$ are the bandpass reconstruction filters. Similar to the forward transform, the inverse filter is applied separately to the horizontal and the vertical directions. By substituting (1)–(3) to (4), the perfect reconstruction constraint can be found as

$$K(z_1)L(z_2)G(z_1) + L(z_1)K(z_2)G(z_2) + \bar{H}(z_1)\bar{H}(z_2)H(z_1)H(z_2) = 1. \tag{5}$$

Equation (5) is a necessary and sufficient condition for perfect reconstruction. There is considerable freedom in choosing these four filters when the orthogonal, the biorthogonal, and the subsampling requirements are dropped from the filter design.

Mallat and Zhong have constructed the wavelet function in such a way that it is the derivative of a smoothing function [2]. The local extrema of the resultant wavelet representation then characterizes the multiscale edges in the image. This representation allows the processing and manipulation of images with edge-based algorithms. Examples include the edge-based image coding, discontinuity-preserving surface reconstruction, contrast enhancement for medical images, and structural-based texture characterization [3], [6]–[14]. The set of wavelet functions is commonly known as the Spline wavelet family. The 1-D lowpass and bandpass filters for order n are written, respectively, as

$$H(z) = \frac{z^n}{2^{2n+1}}(1 + z^{-1})^{2n+1} \tag{6}$$
$$\text{and } G(z) = 2(z^{-1} - 1). \tag{7}$$

Using the perfect reconstruction constraint from the 1-D framework [2], [22], the reconstruction filter $K(z)$ can be expressed as

$$K(z) = \frac{1 - |H(z)|^2}{G(z)} = \frac{z-1}{8}\sum_{k=0}^{2n}\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^{2k}. \tag{8}$$

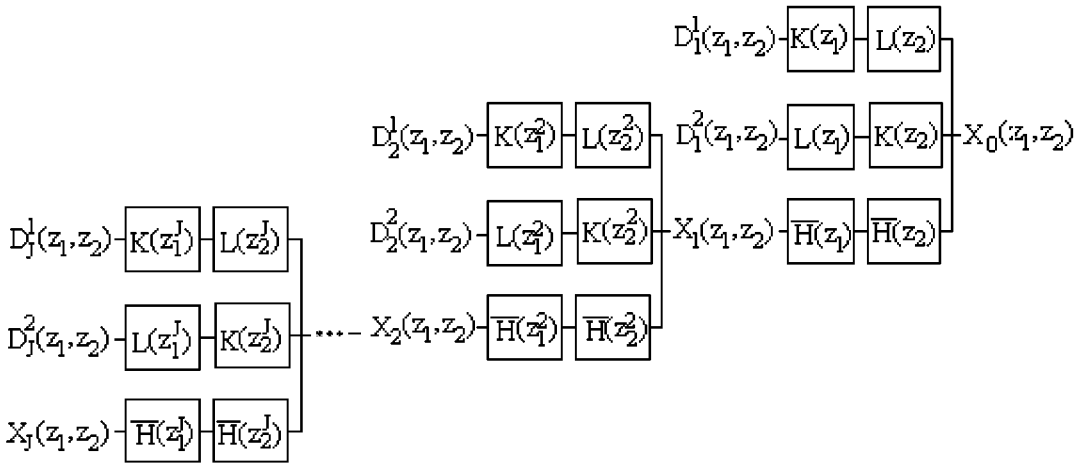


Fig. 2. jth-level inverse overcomplete wavelet transform with $J = 2^{j-1}$.

Substituting (8) to the perfect reconstruction constraint in (5), the expression for $L(z)$ can be obtained. For completeness, its expression is summarized as in Lemma 1.

Lemma 1: To achieve perfect reconstruction, it is required that

$$L(z) = \frac{1+|H(z)|^2}{2}.$$

Proof: Substituting (8) into (5), it can be seen that

$$L(z_2)\left[1-|H(z_1)|^2\right] + L(z_1)\left[1-|H(z_2)|^2\right] + |H(z_1)|^2|H(z_2)|^2 = 1. \quad (9)$$

Rearranging (9), we obtain

$$\left[L(z_2) - \frac{1}{2} - \frac{|H(z_2)|^2}{2}\right]\left[1-|H(z_1)|^2\right] + \left[L(z_1) - \frac{1}{2} - \frac{|H(z_1)|^2}{2}\right]\left[1-|H(z_2)|^2\right] = 0. \quad (10)$$

As outlined by an anonymous reviewer, there are three possible cases for (10) to be satisfied.

Case 1)

$$1-|H(z_1)|^2 \equiv 1-|H(z_2)|^2 \equiv 0.$$

This is not possible since $H(z)$ is not an allpass function.

Case 2)

$$L(z_2) - \frac{1}{2} - \frac{|H(z_2)|^2}{2} \equiv L(z_1) - \frac{1}{2} - \frac{|H(z_1)|^2}{2} \equiv 0.$$

This will give the desired form for $L(z)$.

Case 3) Assume that $f_1(z_2) = L(z_2) - 1/2 - |H(z_2)|^2/2 \neq 0$ and $g_2(z_1) = L(z_1) - 1/2 - |H(z_1)|^2/2$; then, it follows that

$$\frac{f_1(z_2)}{f_2(z_2)} = -\frac{g_2(z_1)}{g_1(z_2)} = c$$

where

$$f_2(z_2) = 1-|H(z_2)|^2 \text{ and } g_1(z_1) = 1-|H(z_1)|^2$$

for some constants c. This suggests that

$$L(z_2) = c + \frac{1}{2} + \left(\frac{1}{2} - c\right)\frac{|H(z_2)|^2}{2}$$

$$\text{and } L(z_1) = -c + \frac{1}{2} + \left(\frac{1}{2} + c\right)\frac{|H(z_1)|^2}{2}.$$

Letting $z = z_1 = z_2$, the requirement becomes $|H(z)|^2 = 1$, which is impossible. Therefore

$$L(z_2) - \frac{1}{2} = \frac{|H(z_2)|^2}{2} \text{ and}$$

$$L(z_1) - \frac{1}{2} = \frac{|H(z_1)|^2}{2}$$

which completes the proof. □

By using Lemma 1 and (6), $L(z)$ can be expanded as

$$L(z) = \frac{1}{2}\left[1 + \frac{\left(z^{(1/2)} + z^{-(1/2)}\right)^{4n+2}}{2^{4n+2}}\right]. \quad (11)$$

Upon comparing the forward and the inverse filters shown in (6)–(8) and (11), it can be seen that the number of filter coefficients for $G(z)$ is always two, regardless of the order of the Spline wavelet. The numbers of filter coefficients for $H(z)$, $K(z)$, and $L(z)$ are $2n+2$, $4n+2$, and $4n+3$, respectively. As the inverse filters are significantly longer than the forward filters, the computational complexity associated with the inverse transform would be much higher than that associated with the forward transform. A detailed analysis of the computational complexity is carried out in Sections II-A and B.

A. Filter Complexity

In order to study the computational complexity associated with the Spline wavelet, we need to expand the filters expressions and find out the numbers of additions and multiplications involved. This complexity metric is of interest for both hardware and software realizations. The form of $G(z)$ given in (7) is simple and requires only one addition and one multiplication. Its complexity is thus given by

$$\text{Complexity } [G] = \text{Cost}_{\text{add}} + \text{Cost}_{\text{multiply}} \quad (12)$$

where Cost_{add} and $\text{Cost}_{\text{multiply}}$ define the cost for an addition and a multiplication operations, respectively. The form of $H(z)$ shown in (6) is expanded using the Binomial theorem as

$$H(z) = \frac{1}{2^{2n+1}} \left\{ \sum_{k=0}^{n} {}^{2n+1}C_k \left[z^{n-k} + z^{-n+k-1} \right] \right\} \quad (13)$$

where

$${}^{m}C_k = \frac{m!}{k!(m-k)!}. \quad (14)$$

Equation (13) shows that the number of additions in $H(z)$ is $2n+1$ and that the number of multiplications is $n+1$, i.e.,

$$\text{Complexity}[H] = (2n+1)\text{Cost}_{\text{add}} + (n+1)\text{Cost}_{\text{multiply}}. \quad (15)$$

In reconstruction, the complexities of $\bar{H}(z)$, $K(z)$, and $L(z)$ need to be determined. $\bar{H}(z)$ is the time reverse of $H(z)$. Its expression can be obtained from (13) simply by replacing z with z^{-1}, i.e.,

$$\bar{H}(z) = \frac{1}{2^{2n+1}} \left\{ \sum_{k=0}^{n} {}^{2n+1}C_k \left[z^{-n+k} + z^{n-k+1} \right] \right\}. \quad (16)$$

Thus, the complexity of $\bar{H}(z)$ is the same as that of $H(z)$ given in (15). In calculating the complexity of $K(z)$, we expand the summations in (8) and use the Binomial theorem [21] to obtain

$$K(z) = B_{2n} \left[z^{2n+1} - z^{-2n} \right] + \sum_{k=1}^{2n} E_k \left[z^{k} - z^{-k+1} \right] \quad (17)$$

where

$$B_m = \frac{1}{8} \sum_{k=m}^{2n} \frac{1}{2^{2k}} {}^{2k}C_{k-m} \quad (18)$$

$$\text{and } E_i = B_{i-1} - B_i. \quad (19)$$

Equation (17) shows that the numbers of additions and multiplications in $K(z)$ are $4n+1$ and $2n+1$, respectively, i.e.,

$$\text{Complexity}[K] = (4n+1)\text{Cost}_{\text{add}} + (2n+1)\text{Cost}_{\text{multiply}}. \quad (20)$$

By employing the Binomial theorem, an expression for $L(z)$ can be obtained by expanding (11) as

$$L(z) = P_1 + \frac{1}{2^{4n+3}} \times \sum_{k=0}^{2n} \left\{ {}^{4n+2}C_k \left[z^{2n+1-k} + z^{-2n-1+k} \right] \right\} \quad (21)$$

where

$$P_1 = \frac{1}{2} \left[1 + \frac{{}^{4n+2}C_{2n+1}}{2^{4n+2}} \right]. \quad (22)$$

An analysis of (21) shows that the numbers of additions and multiplications in $L(z)$ are $4n+2$ and $2n+2$, respectively, i.e.,

$$\text{Complexity}[L] = (4n+2)\text{Cost}_{\text{add}} + (2n+2)\text{Cost}_{\text{multiply}}. \quad (23)$$

B. Computational Complexity

A one-level forward transform involves filtering in both the horizontal and the vertical directions (see Fig. 1). Substituting the filter expression for $H(z)$ in (13) to (1), the lowpass output $X_1(z_1, z_2)$ can be rewritten as

$$X_1(z_1, z_2) = \frac{1}{2^{4n+2}} \sum_{k_1=0}^{n} {}^{2n+1}C_{k_1} \left[z_1^{n-k_1} + z_1^{-n+k_1-1} \right] \times \sum_{k_2=0}^{n} {}^{2n+1}C_{k_2} \left[z_2^{n-k_2} + z_2^{-n+k_2-1} \right] X_0(z_1, z_2). \quad (24)$$

Two multiplications in $H(z_1)$ and $H(z_2)$ are merged into one multiplication in (24); thus, the complexity of obtaining $X_1(z_1, z_2)$ equals to two times the complexity of $H(z)$ minus one multiplication, i.e.,

$$\text{Complexity}[X_1(z_1, z_2)] = (4n+2)\text{Cost}_{\text{add}} + (2n+1)\text{Cost}_{\text{multiply}}. \quad (25)$$

The bandpass outputs can be found by substituting (7) with (2) and (3)

$$D_1^1(z_1, z_2) = 2\left(z_1^{-1} - 1\right) X_0(z_1, z_2) \quad (26)$$

$$D_1^2(z_1, z_2) = 2\left(z_2^{-1} - 1\right) X_0(z_1, z_2). \quad (27)$$

The complexity in obtaining $D_1^1(z_1, z_2)$ or $D_1^2(z_1, z_2)$ is then equal to the complexity in obtaining $G(z)$, i.e.,

$$\text{Complexity}[D_1^1(z_1, z_2)] = \text{Complexity}[D_1^2(z_1, z_2)] = \text{Cost}_{\text{add}} + \text{Cost}_{\text{multiply}}. \quad (28)$$

For the inverse transform, its complexity can be obtained by considering the complexities of $K(z_1)L(z_2)D_1^1(z_1, z_2)$, $L(z_1)K(z_2)D_1^2(z_1, z_2)$ and $\bar{H}(z_1)\bar{H}(z_2)X_1(z_1, z_2)$ (see Fig. 2). Using the filters expression in (17) and (21), $K(z_1)L(z_2)D_1^1(z_1, z_2)$ can be expanded as

$$\begin{aligned} &K(z_1)L(z_2)D_1^1(z_1, z_2) \\ &= \left\{ \frac{B_{2n}}{2^{4n+3}} \left[z_1^{2n+1} - z_1^{-2n} \right] + \sum_{k_1=1}^{2n} \frac{E_{k_1}}{2^{4n+3}} \left[z_1^{k_1} - z_1^{-k_1+1} \right] \right\} \\ &\quad \times \left\{ 2^{4n+3} P_1 + \sum_{k_2=0}^{2n} {}^{4n+2}C_{k_2} \left[z_2^{2n+1-k_2} + z_2^{-2n-1+k_2} \right] \right\} \\ &\quad \times D_1^1(z_1, z_2). \end{aligned} \quad (29)$$

The complexity of $K(z_1)L(z_2)D_1^1(z_1, z_2)$ is thus equal to the sum of the complexities in $K(z)$ and $L(z)$, i.e.,

$$\text{Complexity}\left[K(z_1)L(z_2)D_1^1(z_1, z_2)\right] = (8n+3)\text{Cost}_{\text{add}} + (4n+2)\text{Cost}_{\text{multiply}}. \quad (30)$$

The complexity of $L(z_1)K(z_2)D_1^2(z_1, z_2)$ is also equal to the sum of the complexities in $K(z)$ and $L(z)$ and is thus the same as the complexity of $K(z_1)L(z_2)D_1^1(z_1, z_2)$, i.e.,

$$\text{Complexity}\left[L(z_1)K(z_2)D_1^2(z_1, z_2)\right] = (8n+3)\text{Cost}_{\text{add}} + (4n+2)\text{Cost}_{\text{multiply}}. \quad (31)$$

Using (16), the expression for $\bar{H}(z_1)\bar{H}(z_2)X_1(z_1,z_2)$ can be written as

$$\begin{aligned}\bar{H}(z_1)\bar{H}(z_2)X(z_1,z_2) &= \frac{1}{2^{4n+2}}\left\{\sum_{k_1=0}^{n} {}^{2n+1}C_{k_1}\left[z_1^{-n+k_1}+z_1^{n-k_1+1}\right]\right\} \\ &\times\left\{\sum_{k_2=0}^{2n} {}^{2n+1}C_{k_2}\left[z_2^{-n+k_2}+z_2^{n-k_2+1}\right]\right\} \\ &\times X_1(z_1,z_2). \end{aligned} \tag{32}$$

Two multiplications in $\bar{H}(z_1)$ and $\bar{H}(z_2)$ are combined into one multiplication in (32); thus, the complexity of $\bar{H}(z_1)\bar{H}(z_2)X_1(z_1,z_2)$ is equal to two times the complexity in $\bar{H}(z)$ minus one multiplication, i.e.,

$$\begin{aligned}\text{Complexity}\left[\bar{H}(z_1)\bar{H}(z_2)X_1(z_1,z_2)\right] \\ = (4n+2)\,\text{Cost}_{\text{add}} + (2n+1)\text{Cost}_{\text{multiply}}.\end{aligned} \tag{33}$$

From the complexities expression in (25), (28), (30), (31), and (33), we arrive at Theorem 1.

Theorem 1: A one-level overcomplete forward wavelet transform obtained using the filtering approach as shown in Fig. 1 has a complexity of

$$\begin{aligned}\text{Complexity}_{FB}[\text{Forward}] = (4n+4)\text{Cost}_{\text{add}} \\ +(2n+3)\text{Cost}_{\text{multiply}}.\end{aligned}$$

The inverse transform using the filtering approach as shown in Fig. 2 has a complexity of

$$\begin{aligned}\text{Complexity}_{FB}[\text{Inverse}] = (20n+10)\text{Cost}_{\text{add}} \\ +(10n+5)\text{cost}_{\text{multiply}}.\end{aligned}$$

Proof: A one-level forward wavelet transform consists of three outputs:

- $X_1(z_1,z_2)$;
- $D_1^1(z_1,z_2)$;
- $D_1^2(z_1,z_2)$.

The complexity of the forward wavelet transform can thus be obtained by summing their complexities as shown in (25) and (28). The operations in the inverse transform involve three parts:

- $K(z_1)L(z_2)D_1^1(z_1,z_2)$;
- $L(z_1)K(z_2)D_1^2(z_1,z_2)$;
- $\bar{H}(z_1)\bar{H}(z_2)X_1(z_1,z_2)$.

The inverse complexity is thus obtained by summing their complexities shown in (30), (31), and (33). This completes the proof. □

As shown in Theorem 1, the inverse transform is significantly more complex than the forward transform. The inverse transform is nearly five times more complicated than the forward transform in the wavelet transform. This is undesirable, and we need to reduce the computational complexity in the inverse transform.

III. Proposed Spatial Implementation for the Over-Complete Wavelet Representation

The overcomplete wavelet representation provides a redundant representation of an image. There exists a correlation between the lowpass and the bandpass outputs at different scales. Indeed, many applications, such as the discontinuity-preserving surface reconstruction, contrast enhancement, and denoising, have benefited from this correlation in solving their problems [6]–[14]. We propose to study this correlation in the calculation of the wavelet transform. This can provide an alternative implementation structure that is able to reduce the computational complexity associated with the inverse transform.

A. One Stage of Wavelet Transform

We consider a single stage of wavelet transform in this subsection. The first-level lowpass output is given in (24). We could rewrite this equation using Lemma 2.

Lemma 2: The expression

$$Y(z) = \sum_{k=0}^{n} {}^{2n+1}C_k\left[z^{n-k}+z^{-n+k-1}\right]X_0(z) \tag{34}$$

can be rewritten as

$$\begin{aligned}Y(z) = 2^{2n+1}X_0(z) \\ +\frac{1}{2}\left\{2^{2n}+\sum_{k=1}^{n} A_{n-k}\left[z^{-k}-z^{k}\right]\right\}D_1(z)\end{aligned}$$

where

$$D_1(z) = 2\left(z^{-1}-1\right)X_0(z) \tag{35}$$

$$\text{and } A_k = \sum_{m=0}^{k} {}^{2n+1}C_m. \tag{36}$$

Proof: The proof starts by forming two recursion formulae from (35)

$$z^m X_0(z) = X_0(z) - \frac{D_1(z)}{2}\sum_{k=1}^{m} z^k \tag{37}$$

$$\text{and } z^{-m}X_0(z) = X_0(z) + \frac{D_1(z)}{2}\sum_{k=0}^{m-1} z^{-k} \tag{38}$$

for an arbitrary integer m. Expanding the summations in (34) and using (37) and (38), we obtain

$$\begin{aligned}Y(z) = \frac{1}{2^{2n+1}} \\ \times\left\{2X_0(z)\sum_{k=0}^{n} {}^{2n+1}C_k - \frac{D_1(z)}{2}\right. \\ \times\sum_{l=1}^{n} {}^{2n+1}C_{n-l}\sum_{k=1}^{l} z^k \\ \left.+\frac{D_1(z)}{2}\sum_{l=0}^{n} {}^{2n+1}C_{n-l}\sum_{k=0}^{l} z^{-k}\right\}.\end{aligned} \tag{39}$$

As $2^{2n} = \sum_{k=0}^{n} {}^{2n+1}C_k$, (39) becomes

$$Y(z) = X_0(z) + \frac{1}{2^{2n+2}} \left\{ \sum_{l=0}^{n} {}^{2n+1}C_{n-l} \sum_{k=0}^{l} z^{-k} - \sum_{l=1}^{n} {}^{2n+1}C_{n-l} \sum_{k=1}^{l} z^{k} \right\} D_1(z). \quad (40)$$

Upon expanding the summations in (40), it can be shown that

$$\sum_{l=0}^{n} {}^{2n+1}C_{n-l} \sum_{k=0}^{l} z^{-k} = \sum_{k=0}^{n} \sum_{m=0}^{n-k} {}^{2n+1}C_m z^{-k} \quad (41)$$

$$\text{and } \sum_{l=1}^{n} {}^{2n+1}C_{n-l} \sum_{k=1}^{l} z^{k} = \sum_{k=1}^{n} \sum_{m=0}^{n-k} {}^{2n+1}C_m z^{k}. \quad (42)$$

Substituting (41) and (42) into (40) completes the proof. □

Using Lemma 2, it can be seen that

$$\begin{aligned} H(z_2)X_0(z_1,z_2) &= \frac{1}{2^{2n+1}} \left\{ \sum_{k_2=0}^{n} {}^{2n+1}C_{k_2} \left[z_2^{n-k_2} + z_2^{-n+k_2-1} \right] \right\} \\ &\quad \times X_0(z_1,z_2) \\ &= X_0(z_1,z_2) + \frac{1}{2^{2n+2}} \\ &\quad \times \left\{ 2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2} \left[z_2^{-k_2} - z_2^{k_2} \right] \right\} \\ &\quad \times D_1^2(z_1,z_2). \end{aligned} \quad (43)$$

Equation (43) implies that the transform is applied along z_2 (the column) for every row of the image. Substituting (43) into (24), we arrive at Theorem 2. This theorem provides an alternative method for the implementation of the overcomplete wavelet transform.

Theorem 2: The first-level lowpass output of the overcomplete wavelet transform using the Spline wavelet family with an arbitrary order n can be rewritten as

$$X_1(z_1,z_2) = X_0(z_1,z_2) + \frac{1}{2^{2n+2}} \left\{ F_1\left[D_1^2(z_1,z_2)\right] + F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right] \right\}$$

where

$$\begin{aligned} F_1\left[D_1^2(z_1,z_2)\right] &= Y_1(z_1,z_2) \\ &= \left\{ 2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2} \left[z_2^{-k_2} - z_2^{k_2} \right] \right\} \\ &\quad \times D_1^2(z_1,z_2) \end{aligned} \quad (44)$$

and

$$\begin{aligned} &F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right] \\ &\quad = \left\{ 2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1} \left[z_1^{-k_1} - z_1^{k_1} \right] \right\} \\ &\qquad \times \left[D_1^1(z_1,z_2) + \frac{(z_1^{-1}-1)}{2^{2n+1}} Y_1(z_1,z_2) \right]. \end{aligned} \quad (45)$$

Proof: The proof starts by substituting (43) into (24), i.e.,

$$\begin{aligned} X_1(z_1,z_2) &= H(z_1)X_0(z_1,z_2) \\ &\quad + \frac{H(z_1)}{2^{2n+2}} \left\{ 2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2} \left[z_2^{-k_2} - z_2^{k_2} \right] \right\} \\ &\quad \times D_1^2(z_1,z_2) \\ &= X_0(z_1,z_2) + \frac{1}{2^{2n+2}} \\ &\quad \times \left\{ 2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1} \left[z_1^{-k_1} - z_1^{k_1} \right] \right\} \\ &\quad \times D_1^1(z_1,z_2) + \frac{D_1^2(z_1,z_2)}{2^{4n+3}} \\ &\quad \times \left\{ \sum_{k_1=0}^{n} {}^{2n+1}C_{k_1} \left[z_1^{n-k_1} + z_1^{-n+k_1-1} \right] \right\} \\ &\quad \times \left\{ 2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2} \left[z_2^{-k_2} - z_2^{k_2} \right] \right\}. \end{aligned} \quad (46)$$

Note that the last term in (46) can be expressed as

$$\begin{aligned} &\frac{D_1^2(z_1,z_2)}{2^{4n+3}} \left\{ \sum_{k_1=0}^{n} {}^{2n+1}C_{k_1} \left[z_1^{n-k_1} + z_1^{-n+k_1-1} \right] \right\} \\ &\quad \times \left\{ 2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2} \left[z_2^{-k_2} - z_2^{k_2} \right] \right\} \\ &\quad = \frac{1}{2^{2n+2}} \Bigg\{ D_1^2(z_1,z_2) \\ &\qquad + \frac{1}{2^{2n+2}} \left[2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1} \left[z_1^{-k_1} - z_1^{k_1} \right] \right] \\ &\qquad \times 2\left(z_1^{-1}-1\right) D_1^2(z_1,z_2) \Bigg\} \\ &\qquad \times \left\{ 2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2} \left[z_2^{-k_2} - z_2^{k_2} \right] \right\} \\ &\quad = \frac{1}{2^{2n+2}} Y_1(z_1,z_2) \\ &\qquad + \frac{1}{2^{4n+4}} \left[2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1} \left[z_1^{-k_1} - z_1^{k_1} \right] \right] \\ &\qquad \times 2\left(z_1^{-1}-1\right) Y_1(z_1,z_2). \end{aligned} \quad (47)$$

Substituting (47) to (46), the first-level lowpass output can be expressed as

$$\begin{aligned} X_1(z_1,z_2) &= X_0(z_1,z_2) \\ &\quad + \frac{1}{2^{2n+2}} \left\{ 2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1} \left[z_1^{-k_1} - z_1^{k_1} \right] \right\} \\ &\quad \times D_1^1(z_1,z_2) + \frac{1}{2^{2n+2}} Y_1(z_1,z_2) \\ &\quad + \frac{1}{2^{4n+4}} \left\{ 2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1} \left[z_1^{-k_1} - z_1^{k_1} \right] \right\} \\ &\quad \times 2\left(z_1^{-1}-1\right) Y_1(z_1,z_2) \end{aligned}$$

$$X_1(z_1,z_2) = X_0(z_1,z_2) + \frac{1}{2^{2n+2}} Y_1(z_1,z_2) + \frac{1}{2^{2n+2}} \left\{ 2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1} \left[z_1^{-k_1} - z_1^{k_1} \right] \right\} \times \left[D_1^1(z_1,z_2) + \frac{(z_1^{-1}-1)}{2^{2n+1}} Y_1(z_1,z_2) \right] \quad (48)$$

which completes the proof. □

Theorem 2 provides a way to relate the first-level lowpass output with the original signal and the two bandpass outputs. Employing the concept of filtering, Theorem 2 can be restated as

$$X_1(z_1,z_2) = X_0(z_1,z_2) + M_1(z_1) D_1^1(z_1,z_2) + M_2(z_1) M_3(z_2) D_1^2(z_1,z_2). \quad (49)$$

For the Spline wavelet family, the filters $M_1(z)$, $M_2(z)$, and $M_3(z)$ can be written as

$$M_1(z) = \frac{1}{2^{2n+2}} \left\{ 2^{2n} + \sum_{k=1}^{n} A_{n-k} \left[z^{-k} - z^{k} \right] \right\} \quad (50)$$

$$M_2(z) = 1 + M_1(z) G(z) \quad (51)$$

$$M_3(z) = M_1(z). \quad (52)$$

Since a separable wavelet scheme is used in the design, all filters are 1-D, as can be seen in (50)–(52). Note that all filters including $G(z)$, $M_1(z)$, $M_2(z)$, and $M_3(z)$ are FIR in structure. This implies that the proposed implementation in (49) or in Theorem 2 will not produce errors or noise amplification and is stable. In fact, (49) can be extended to other wavelet families with different choices of $M_1(z)$, $M_2(z)$, and $M_3(z)$. To see this, we could firstly look at the 1-D case due to the similarity between the designs of the 1-D and the 2-D cases [such as (8)]. In the 1-D case, we have

$$X_1(z) = X_0(z) + M(z) D_1(z) \quad (53)$$

where

$$X_1(z) = H(z) X_0(z) \quad (54)$$

$$D_1(z) = G(z) X_0(z). \quad (55)$$

Placing (54) and (55) in (53), it can be seen that

$$M(z) = \frac{H(z)-1}{G(z)}. \quad (56)$$

A stable implementation is obtained if $M(z)$ is in FIR structure, i.e.,

$$H(z) - 1 \text{ is divisible by } G(z). \quad (57)$$

In the 2-D case, we could consider [cf. (8)]

$$M_1(z) = M(z). \quad (58)$$

Placing (58), (56), and (1)–(3) in (49), it can be seen that

$$H(z_1)[H(z_2) - 1] = M_2(z_1) M_3(z_2) G(z_2). \quad (59)$$

Since the design is separable, (59) implies that

$$M_2(z) = H(z) \quad (60)$$

$$M_3(z) = \frac{H(z)-1}{G(z)} = M_1(z). \quad (61)$$

Therefore, for any other choices of wavelets besides the Spline wavelet family, the overcomplete wavelet representation with $G(z)$ and $H(z)$ that satisfy (57) can be implemented using the proposed spatial implementation as described in Theorem 2 or in (49).

Theorem 2 not only provides an alternative implementation scheme for the forward transform but also simplifies the computation for the inverse transform. In particular, the inverse transform can be easily calculated as

$$X_0(z_1,z_2) = X_1(z_1,z_2) - \frac{1}{2^{2n+2}} \left\{ F_1\left[D_1^2(z_1,z_2) \right] + F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2) \right] \right\}. \quad (62)$$

The inverse transform is very similar to the forward transform in Theorem 2. The proposed implementation scheme for both the forward and the inverse transforms according to Theorem 2 and (62) is shown in Fig. 3. It can be seen that a simple spatial implementation is used for image reconstruction. It greatly simplifies the computation involved in the inverse transform.

B. Computational Complexity

In analyzing the computational complexity associated with the proposed scheme, the complexities in $F_1\left[D_1^2(z_1,z_2)\right]$ and $F_2[D_1^1(z_1,z_2), Y_1(z_1,z_2)]$ given in (44) and (45) can be written as

$$\text{Complexity}\left[F_1\left[D_1^2(z_1,z_2)\right]\right] = (2n)\,\text{Cost}_{\text{add}} + (n)\text{Cost}_{\text{multiply}} \quad (63)$$

$$\text{Complexity}\left[F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right]\right] = (2n+2)\text{Cost}_{\text{add}} + (n+1)\text{Cost}_{\text{multiply}}. \quad (64)$$

Using both (63) and (64), the complexities for the forward and the inverse wavelet transforms can be defined as in Theorem 3.

Theorem 3: The overcomplete forward wavelet transform using the proposed implementation scheme described in Theorem 2 has a complexity of

$$\text{Complexity}_S[\text{Forward}] = (4n+6)\text{Cost}_{\text{add}} + (2n+4)\text{Cost}_{\text{multiply}}$$

and the inverse transform has a complexity of

$$\text{Complexity}_S[\text{Inverse}] = (4n+4)\text{Cost}_{\text{add}} + (2n+2)\text{Cost}_{\text{multiply}}.$$

Proof: The forward transform has three outputs: the two bandpass outputs $D_1^1(z_1,z_2)$ and $D_1^2(z_1,z_2)$ and the lowpass output $X_1(z_1,z_2)$. The complexities for obtaining the two bandpass outputs are given in (28). According to the new implementation scheme in Theorem 2, the lowpass output involves the calculation of $F_1\left[D_1^2(z_1,z_2)\right]$ and $F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right]$. Its cost would include two

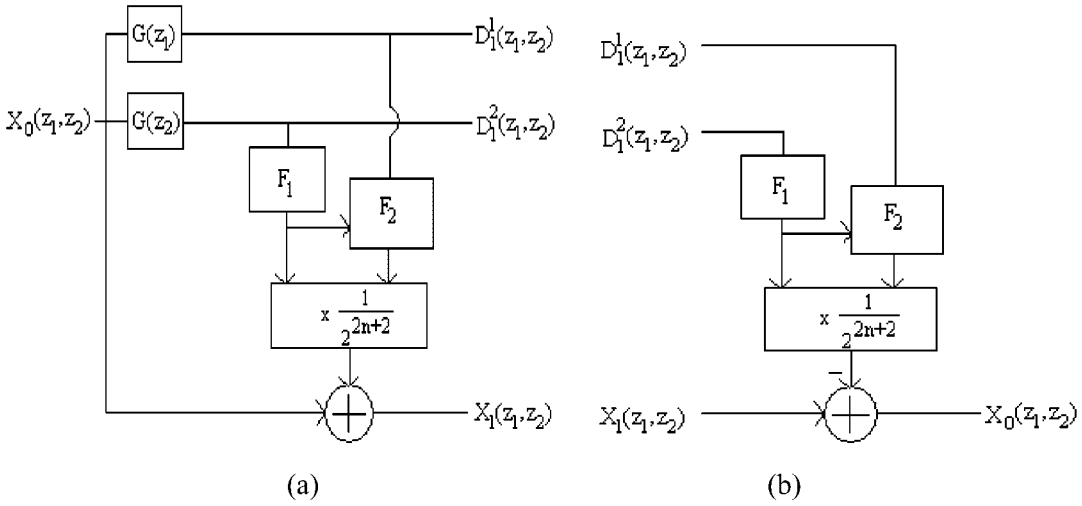


Fig. 3. Proposed spatial domain 2-D overcomplete wavelet transform. (a) Forward transform. (b) Inverse transform.

additions: one multiplication and the complexities in obtaining $F_1\left[D_1^2(z_1,z_2)\right]$ and $F_2\left[D_1^1(z_1,z_2),Y_1(z_1,z_2)\right]$. By summing up these complexities, the forward complexity can be determined. In reconstruction, there is no need to calculate $D_1^1(z_1,z_2)$ or $D_1^2(z_1,z_2)$. Thus, the inverse complexity would only involve two additions: one multiplication and the complexities in obtaining $F_1\left[D_1^2(z_1,z_2)\right]$ and $F_2\left[D_1^1(z_1,z_2),Y_1(z_1,z_2)\right]$ □

In the filtering approach, the complexity of the forward transform is much smaller than that of the inverse transform, as shown in Theorem 1. In contrast, the complexity of the forward transform is slightly higher than that of the inverse transform in the proposed implementation scheme (Theorem 3). On comparing the two approaches, we see that the complexity of the forward transform of the proposed scheme is slightly higher than that of the filtering approach. However, the complexity of the inverse transform of the proposed scheme is much lower than that of the filtering approach since no filtering is required for the reconstruction of the original signal. Instead, a simple spatial implementation is used for the reconstruction, and therefore, its computational complexity is greatly reduced. It can be seen that the inverse transform using our proposed scheme is one multiplication less than the forward transform in the filtering approach.

Besides a decrease in computational complexity of the inverse transform, the proposed implementation handles the boundary in a more flexible way than the filtering approach. As images are of finite length, the boundary needs to be extended in a practical implementation [23]. Under the filtering approach, there are two common ways of dealing with the boundary extension problem for perfect reconstruction: The image is extended before filtering, or the boundary pixels are corrected after the inverse transform. The former would increase the computational time, especially for a large image, whereas the latter involves the design of nontrivial filter-dependent boundary correction rules for different boundary extension methods. However, using our proposed implementation, it can be seen that the prediction terms $F_1\left[D_1^2(z_1,z_2)\right]$ and $F_2\left[D_1^1(z_1,z_2),Y_1(z_1,z_2)\right]$, remain the same in both forward and inverse transforms. As the prediction terms are unchanged, there is no need to do boundary correction after reconstruction. Any boundary extension scheme can be used while maintaining the ease of the inverse transform.

C. Multiple-Level Wavelet Decompositions

The correlation between the lowpass and the bandpass outputs is explored to provide an alternative implementation for the first-level overcomplete wavelet transform. In this section, we extend the proposed scheme to the multiple decomposition framework. As in Fig. 1, the second-level wavelet transform involves $G\left(z^2\right)$, $H\left(z^2\right)$, $K\left(z^2\right)$, and $L\left(z^2\right)$. The two bandpass outputs and the lowpass output are written as

$$D_2^1(z_1,z_2) = G\left(z_1^2\right)X_1(z_1,z_2) \tag{65}$$

$$D_2^2(z_1,z_2) = G\left(z_2^2\right)X_1(z_1,z_2) \tag{66}$$

$$X_2(z_1,z_2) = H\left(z_1^2\right)H\left(z_2^2\right)X_1(z_1,z_2). \tag{67}$$

Let us first analyze the computational complexity involved in calculating the second-level output $D_2^1(z_1,z_2)$ compared with the first-level output $D_1^1(z_1,z_2)$. Equation (65) can be rewritten as

$$D_2^1(z_1,z_2) = G\left(z_1^2\right)X_{1,\text{even}}\left(z_1^2,z_2\right) + G\left(z_1^2\right)X_{1,\text{odd}}\left(z_1^2,z_2\right) \tag{68}$$

where $X_{1,\text{even}}(z_1,z_2)$ and $X_{1,\text{odd}}(z_1,z_2)$ are, respectively, the even and the odd parts of $X_{1,}(z_1,z_2)$. As the length of either $X_{1,\text{even}}(z_1,z_2)$ or $X_{1,\text{odd}}(z_1,z_2)$ is only half of that of $X_1(z_1,z_2)$, the total number of computations involved in obtaining $D_2^1(z_1,z_2)$ would be the same as that of $D_1^1(z_1,z_2)$. In fact, for any arbitrary number of decomposition level i

$$D_i^1(z_1,z_2) = G\left(z_1^{2^{i-1}}\right)X_{i-1}(z_1,z_2) = \sum_{k=0}^{2^i-1} G\left(z_1^{2^{i-1}}\right)X'_{i-1,k}\left(z_1^{2^{i-1}},z_2\right) \tag{69}$$

where

$$x'_{i,k}(n_1,n_2) = x_{i,k}\left(2^i n_1 + k, n_2\right). \tag{70}$$

An analysis of (69) shows that the number of computations involved in obtaining $D_i^1(z_1,z_2)$ remains the same as that of $D_1^1(z_1,z_2)$. Using (66), a similar conclusion can be arrived for $D_i^2(z_1,z_2)$. Therefore, the computational complexities of the two bandpass outputs remain unchanged, regardless of the number of decomposition levels.

The lowpass output $X_2(z_1,z_2)$ in (67) can be obtained by substituting (13) in (67) as

$$\begin{aligned}X_2(z_1,z_2) =& \frac{1}{2^{4n+2}}\sum_{k_1=0}^{n} {}^{2n+1}C_{k_1} \\ &\times \left[z_1^{2(n-k_1)} + z_1^{2(-n+k_1-1)}\right] \\ &\times \sum_{k_2=0}^{n} {}^{2n+1}C_{k_2}\left[z_2^{2(n-k_2)} + z_2^{2(-n+k_2-1)}\right] \\ &\times X_1(z_1,z_2). \end{aligned} \tag{71}$$

Upon comparing (67) and (24), it can be seen that both expressions look very similar if z is replaced by z^2 in (24). The analysis in Theorem 2 can thus be extended to obtain $X_2(z_1,z_2)$ by replacing $D_1^1(z_1,z_2)$ with $D_2^1(z_1,z_2)$, $D_1^2(z_1,z_2)$ with $D_2^2(z_1,z_2)$, and z with z^2. This takes into account the filter interpolation in the subsequent decomposition levels. Mathematically, the second-level lowpass output can be obtained as

$$\begin{aligned}X_2(z_1,z_2) = X_1(z_1,z_2) + \frac{1}{2^{2n+2}}\{&F_1\left[D_2^2(z_1,z_2)\right] \\ &+F_2\left[D_2^1(z_1,z_2), Y_2(z_1,z_2)\right]\}\end{aligned}$$

where

$$\begin{aligned}F_1\left[D_2^2(z_1,z_2)\right] =& Y_2(z_1,z_2) \\ =& \left\{2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2}\left[z_2^{-2k_2} - z_2^{2k_2}\right]\right\} \\ &\times D_2^2(z_1,z_2)\end{aligned}$$

and

$$\begin{aligned}&F_2\left[D_2^1(z_1,z_2), Y_2(z_1,z_2)\right] \\ &= \left\{2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1}\left[z_1^{-2k_1} - z_1^{2k_1}\right]\right\} \\ &\quad\times \left[D_2^1(z_1,z_2) + \frac{\left(z_2^{-2}-1\right)}{2^{2n+1}} Y_2(z_1,z_2)\right].\end{aligned}$$

Using this result, we arrive at the following theorem for any arbitrary number of decompositions.

Theorem 4: For an overcomplete wavelet transform, the ith-level lowpass output can be written as

$$\begin{aligned}X_i(z_1,z_2) = X_{i-1}(z_1,z_2) + \frac{1}{2^{2n+2}}\{&F_1\left[D_i^2(z_1,z_2)\right] \\ &+F_2\left[D_i^1(z_1,z_2), Y_i(z_1,z_2)\right]\}\end{aligned}$$

where

$$\begin{aligned}F_1\left[D_i^2(z_1,z_2)\right] =& Y_i(z_1,z_2) \\ =& \left\{2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2} \right. \\ &\left. \times \left[z_2^{-2^{i-1}k_2} - z_2^{2^{i-1}k_2}\right]\right\} \\ &\times D_i^2(z_1,z_2)\end{aligned} \tag{72}$$

and

$$\begin{aligned}&F_2\left[D_i^1(z_1,z_2), Y_i(z_1,z_2)\right] \\ &= \left\{2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1}\left[z_1^{-2^{i-1}k_1} - z_1^{2^{i-1}k_1}\right]\right\} \\ &\quad\times \left[D_i^1(z_1,z_2) + \frac{\left(z_2^{-2^{i-1}}-1\right)}{2^{2n+1}} Y_i(z_1,z_2)\right].\end{aligned} \tag{73}$$

Proof: The proof relies on the fact that z is replaced by $z^{2^{i-1}}$ in the filter H for the ith decomposition level. The lowpass output can be written as

$$X_i(z_1,z_2) = H\left(z_1^{2^{i-1}}\right) H\left(z_2^{2^{i-1}}\right) X_{i-1}(z_1,z_2). \tag{74}$$

Using Lemma 2, (74) can be rewritten as

$$\begin{aligned}&X_i(z_1,z_2) \\ &= H\left(z_1^{2^{i-1}}\right) X_{i-1}(z_1,z_2) + \frac{H\left(z_1^{2^{i-1}}\right)}{2^{2n+2}} \\ &\quad\times \left\{2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2}\left[z_2^{-2^{i-1}k_2} - z_2^{2^{i-1}k_2}\right]\right\} \\ &\quad\times D_i^2(z_1,z_2) \\ &= X_{i-1}(z_1,z_2) + \frac{1}{2^{2n+2}} \\ &\quad\times \left\{2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1}\left[z_1^{-2^{i-1}k_1} - z_1^{2^{i-1}k_1}\right]\right\} \\ &\quad\times D_i^1(z_1,z_2) + \frac{D_i^2(z_1,z_2)}{2^{4n+3}} \\ &\quad\times \left\{\sum_{k_1=0}^{n} {}^{2n+1}C_{k_1}\left[z_1^{2^{i-1}(n-k_1)}\right.\right. \\ &\qquad\left.\left. + z_1^{2^{i-1}(-n+k_1-1)}\right]\right\} \\ &\quad\times \left\{2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2}\left[z_2^{-2^{i-1}k_2} - z_2^{2^{i-1}k_2}\right]\right\}.\end{aligned} \tag{75}$$

Following the analysis in (47) and (48), we will arrive at the expressions shown in (72) and (73). □

Theorem 4 provides a general expression for the lowpass output. By comparing Theorem 4 and Theorem 2, it can be seen that the number of computations remains unchanged. Therefore, the computational complexity of the lowpass output is independent of the decomposition level. In summary, the computational complexity of the proposed implementation remains the same as in Theorem 3 for any arbitrary number of decompositions.

TABLE I
FILTER COEFFICIENTS FOR THE QUADRATIC AND THE CUBIC SPLINE WAVELETS

n	-5	-4	-3	-2	-1	0	1	2	3	4	5
Linear											
$h_0(n)$						$\frac{1}{2}$	$\frac{1}{2}$				
g(n)						-2	2				
$k_0(n)$					$\frac{1}{8}$	$-\frac{1}{8}$					
$l_0(n)$					$\frac{1}{8}$	$\frac{6}{8}$	$\frac{1}{8}$				
Quadratic											
$h_1(n)$					$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$			
g(n)						-2	2				
$k_1(n)$			$\frac{1}{128}$	$\frac{7}{128}$	$\frac{11}{64}$	$-\frac{11}{64}$	$-\frac{7}{128}$	$-\frac{1}{128}$			
$l_1(n)$			$\frac{1}{128}$	$\frac{6}{128}$	$\frac{15}{128}$	$\frac{84}{128}$	$\frac{15}{128}$	$\frac{6}{128}$	$\frac{1}{128}$		
Cubic											
$h_2(n)$				$\frac{1}{32}$	$\frac{5}{32}$	$\frac{5}{16}$	$\frac{5}{16}$	$\frac{5}{32}$	$\frac{1}{32}$		
g(n)						-2	2				
$k_2(n)$	$\frac{1}{2048}$	$\frac{11}{2048}$	$\frac{7}{256}$	$\frac{11}{128}$	$\frac{193}{1024}$	$\frac{-193}{1024}$	$\frac{-11}{128}$	$\frac{-7}{256}$	$\frac{-11}{2048}$	$\frac{-1}{2048}$	
$l_2(n)$	$\frac{1}{2048}$	$\frac{10}{2048}$	$\frac{45}{2048}$	$\frac{120}{2048}$	$\frac{210}{2048}$	$\frac{1276}{2048}$	$\frac{210}{2048}$	$\frac{120}{2048}$	$\frac{45}{2048}$	$\frac{10}{2048}$	$\frac{1}{2048}$

IV. DESIGN EXAMPLES

The coefficients for some low-order wavelets in the Spline family are tabulated in Table I. Note that the lengths of the reconstruction filters $K(z)$ and $L(z)$ are always larger than that of $H(z)$ and $G(z)$. Three examples are given. The first one is the linear Spline. Substituting $n = 0$ to (44) and (45), the new implementation for the lowpass output becomes

$$X_1(z_1,z_2) = X_0(z_1,z_2) + \frac{1}{4}\Big\{D_1^1(z_1,z_2) + \frac{1}{2}\left[D_1^2(z_1,z_2) + z_1^{-1}D_1^2(z_1,z_2)\right]\Big\}.$$

By substituting $n = 1$ to (44) and (45), the new implementation for the quadratic Spline wavelet is

$$X_1(z_1,z_2) = X_0(z_1,z_2) + \frac{1}{16}\Big\{F_1\left[D_1^2(z_1,z_2)\right] + F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right]\Big\}$$

where

$$\begin{aligned} F_1\left[D_1^2(z_1,z_2)\right] &= Y_1(z_1,z_2) \\ &= \left(4 + z_2^{-1} - z_2\right) D_1^2(z_1,z_2) \\ F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right] &= \left[4 + z_1^{-1} - z_1\right] \\ &\quad \times \Big[D_1^1(z_1,z_2) + \frac{z_1^{-1}-1}{8} Y_1(z_1,z_2)\Big]. \end{aligned}$$

Similarly, by substituting $n = 2$ to (44) and (45), the new implementation for the cubic Spline wavelet becomes

$$X_1(z_1,z_2) = X_0(z_1,z_2) + \frac{1}{64}\Big\{F_1\left[D_1^2(z_1,z_2)\right] + F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right]\Big\}$$

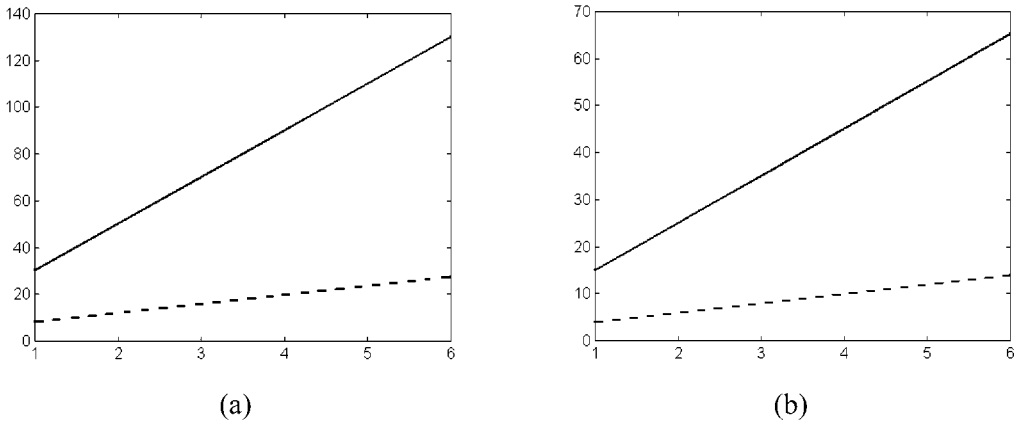


Fig. 4. Plots of (a) the number of additions and (b) the number of multiplications for different orders of Spline wavelet of the 2-D wavelet transform. The solid line represents the filtering approach, and the dotted line represents the proposed approach.

where

$$\begin{aligned} F_1\left[D_1^2(z_1,z_2)\right] &= Y(z_1,z_2) \\ &= \left[16+6\left(z_2^{-1}-z_2\right)\right. \\ &\quad \left.+z_2^{-2}-z_2^2\right] D_1^2(z_1,z_2) \\ F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right] &= \left[16+6\left(z_1^{-1}-z_1\right)\right. \\ &\quad \left.+z_1^{-2}-z_1^2\right] \\ &\quad \times \left[D_1^1(z_1,z_2)\right. \\ &\quad \left.+\frac{z_1^{-1}-1}{32} Y_1(z_1,z_2)\right]. \end{aligned}$$

The computational complexities for the linear $(n=0)$, the quadratic, and the cubic Spline wavelets are shown in Table III. It can be seen that the saving in computation of the inverse transform is significant. For the quadratic Spline wavelet, the number of additions is reduced from 30 to eight, whereas the number of multiplications is reduced from 15 to four. This corresponds to a saving of 73.3% for both the additions and multiplications. For the cubic Spline wavelet, the number of additions is reduced from 50 to 12, whereas the number of multiplications is reduced from 25 to six. This corresponds to a saving of 76.0% for both the additions and multiplications. Fig. 4 shows a comparison of the computational complexity between the filtering approach and the proposed implementation for different orders of Spline wavelets. It can be seen that the saving in computation asymptotically approaches five times for both the additions and multiplications.

V. Experimental Results

The theoretical analysis of both the filtering approach and the proposed implementation has been presented in Sections II–IV. In this section, we will confirm the theoretical findings experimentally by running Visual C++ programs on a PII 333 MHz PC. The first case we considered is the linear Spline wavelet with $n=0$. Its filter coefficients are given in Table I, whereas the numbers of additions and multiplications are given in Table II. Fig. 5 shows a plot of the computation times for the filtering approach and the proposed implementation using different image sizes and by setting the decomposition

TABLE II
Computational Complexity of the Quadratic and the Cubic Spline Wavelets of Two-Dimensional Wavelet Transform

	Filtering Approach		Proposed Implementation	
	Forward	Inverse	Forward	Inverse
Linear				
Additions	4	10	6	4
Multiplications	3	5	4	2
Quadratic				
Additions	8	30	10	8
Multiplications	5	15	6	4
Cubic				
Additions	12	50	14	12
Multiplications	7	25	8	6

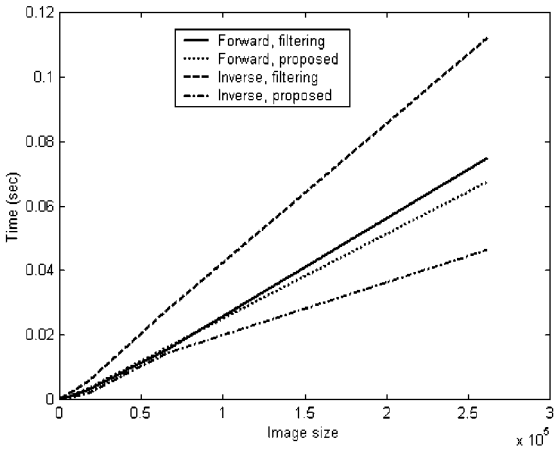


Fig. 5. Plot of the computation times for different image sizes in a single decomposition using the linear Spline wavelet.

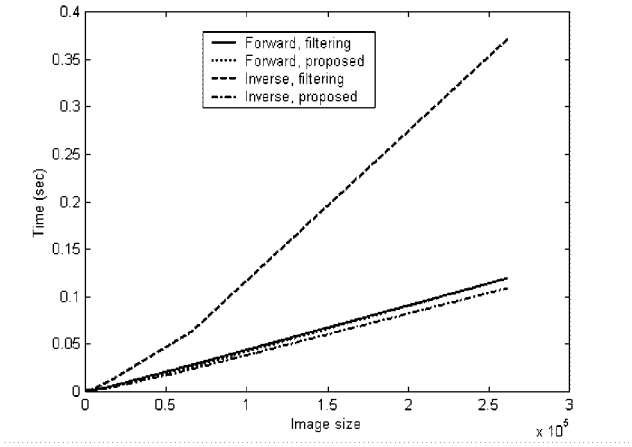


Fig. 6. Plot of the computation times for different image sizes in a single decomposition using the quadratic Spline wavelet.

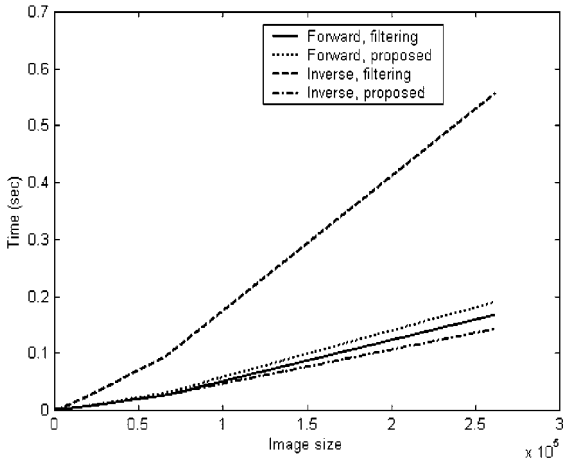


Fig. 7. Plot of the computation times for different image sizes in a single decomposition using the cubic Spline wavelet.

level to one. The experimental results match very well with the theoretical findings. The computation time for the inverse transform that uses the filtering approach is much longer than that used in our proposed implementation. The average time for the computation required in a single pixel can be calculated as

$$T = \frac{1}{K}\sum_{i=0}^{K-1}\frac{\text{Time}_i}{N_i M_i} \tag{76}$$

where Time_i, N_i, and M_i denote, respectively, the computation time, the width, and the height of the ith image. K is the total number of images in the test. This metric denotes the average computation time required in a single pixel.

For the linear Spline wavelet with a single level of decomposition, the average computation time of each of the forward and the inverse transforms using the filtering approach is 2.04×10^{-7} and 3.67×10^{-7} s, respectively. The average computation time of each of the forward and the inverse transforms using our proposed implementation is 2.13×10^{-7} and 1.46×10^{-7} s, respectively. The forward transform using our proposed implementation is slightly slower than that using the filtering approach, whereas the inverse transform using our proposed implementation is much faster. There is a 60.2% speed up in the reconstruction.

Fig. 6 shows the computation times for the Quadratic Spline wavelet. Similar to the linear Spline wavelet case in Fig. 5, the inverse transform from our proposed implementation is much faster than that from the filtering approach. Using the filtering approach, the average computation time calculated according to (76) for each of the forward and the inverse transforms is 4.05×10^{-7} and 9.03×10^{-7}, respectively. Using our proposed implementation, the average computation time for each of the forward and the inverse transforms becomes 3.58×10^{-7} and 2.97×10^{-7}, respectively. We can see that there is a 67.1% speedup in the reconstruction by using our proposed implementation. It is interesting to note that the forward transform using our proposed implementation is slightly faster than that using the filtering approach. Although the numbers of additions and multiplications in our proposed implementation are larger than that in the filtering approach (Table II), there are other factors that affect the computation time in the actual implementation. In this case, the addressing in our proposed implementation is slightly more efficient than that in the filtering approach. This results in a faster execution time.

Another example shown is the cubic Spline wavelet. Fig. 7 shows the computation times of the filtering approach and our proposed implementation. Consistent with the theoretical findings, the inverse transform using our proposed implementation is the fastest. The average computation time for each of the forward and the inverse transforms using the filtering approach is 4.31×10^{-7} and 1.44×10^{-6}, respectively. The average computation time for each of the forward and the inverse transforms using our proposed implementation is 4.78×10^{-7} and 3.77×10^{-7}, respectively. There is a 73.8% speedup in the inverse transform.

The previous results concern a single level of decomposition. The theoretical findings stated in Theorems 1 and 3 concerning the computational complexity of the filtering approach and our proposed implementation have been confirmed experimentally. The theoretical findings for multiple levels of decomposition are presented next. Fig. 8 shows the computation times when the decomposition level is increased to two and five for the linear Spline case. Similar to the case of the single level of decomposition, the inverse transform using our proposed implementation is the fastest. In addition, a linear performance is observed from the result. Table III shows the average computation times calculated from (76). It can be seen that for the forward and the inverse transforms using either the filtering approach or our proposed implementation, the computation times for an ith level of decomposition are approximately equal to i times that for a single level of decomposition. This fits very well with our theoretical findings described in Section III-C.

VI. Conclusion

The computational complexity of the overcomplete wavelet transform for the Spline wavelet family with an arbitrary order is studied in this paper. By deriving general expressions for the

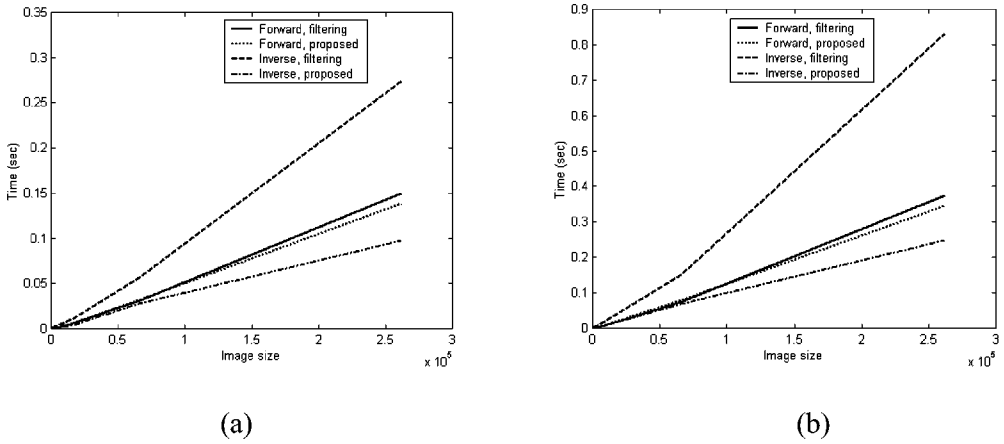


Fig. 8. Plots of the computation times for different image sizes using the linear Spline wavelet for a (a) two and (b) five level of decompositions.

TABLE III
AVERAGE COMPUTATION TIMES FOR THE LINEAR, THE QUADRATIC, AND THE CUBIC SPLINE WAVELETS

	Filtering Approach		Proposed Implementation	
	Forward	Inverse	Forward	Inverse
Linear				
Level=1	2.04×10^{-7}	3.67×10^{-7}	2.13×10^{-7}	1.46×10^{-7}
Level=2	4.12×10^{-7}	7.86×10^{-7}	4.28×10^{-7}	2.96×10^{-7}
Level=5	1.02×10^{-6}	2.30×10^{-6}	1.07×10^{-6}	8.47×10^{-7}
Quadratic				
Level=1	4.05×10^{-7}	9.03×10^{-7}	3.58×10^{-7}	2.97×10^{-7}
Level=2	8.59×10^{-7}	1.87×10^{-6}	7.65×10^{-7}	6.53×10^{-7}
Level=3	1.36×10^{-6}	2.70×10^{-6}	1.19×10^{-6}	1.04×10^{-6}
Cubic				
Level=1	4.31×10^{-7}	1.44×10^{-6}	4.78×10^{-7}	3.77×10^{-7}
Level=2	8.79×10^{-7}	2.96×10^{-6}	1.06×10^{-6}	8.13×10^{-7}
Level=3	1.33×10^{-6}	4.65×10^{-6}	1.59×10^{-6}	1.23×10^{-6}

computational complexity using the conventional filtering implementation, we found the inverse transform to be significantly more complicated than the forward transform. In fact, it asymptotically approaches five times for a large filter order. In order to reduce the computations, we propose a new spatial implementation based on the exploitation of the correlation between the lowpass and the bandpass outputs inherent in the overcomplete representation. Both theoretical studies and experimental findings reveal that the new spatial implementation results in an efficient inverse structure. We also demonstrated that the computational complexity associated with the inverse transform using the proposed spatial implementation is slightly more efficient than the complexity associated with the forward transform using the filtering approach. Furthermore, we showed that unlike the conventional filtering implementation, the spatial implementation allows the use of an arbitrary boundary extension method and requires no boundary correction.

ACKNOWLEDGMENT

N. F. Law thanks the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, the Hong Kong Polytechnic University for the support she receives under its research fellowship scheme. The authors would like to thank all anonymous reviewers for their suggestions, in particular, to one of the reviewers who contributed to the proof of Lemma 1.

REFERENCES

[1] D. Marr, "Representing visual information," in *Computer Vision System*, A. Hanson and E. M. Riseman, Eds. New York: Academic, 1978, pp. 61–80.

[2] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Trans. Pattern Ana. Machine Intell.*, vol. 14, pp. 710–732, July 1992.

[3] A. W. C. Liew, "Multiscale wavelet analysis of edges: Issues of uniqueness and reconstruction," Ph.D., Univ. Tasmania, Hobart, Australia, 1997.

[4] A. W. C. Liew and N. F. Law, "Reconstruction from 2D wavelet transform modulus maxima using projection," *Proc. Inst. Elect. Eng.—Vision, Image Signal Process.*, vol. 147, pp. 176–184, 2000.

[5] A. W. C. Liew, N. F. Law, and D. T. Nguyen, "Direct reconstruction method for wavelet transform extrema representation," *Proc. Inst. Elect. Eng.—Vision, Image Signal Process.*, vol. 144, pp. 193–198, 1997.

[6] Y. Y. Tang, L. Yang, and J. Liu, "Characterization of Dirac structure edges with wavelet transform," *IEEE Trans. Syst., Man, Cybern. B*, vol. 30, pp. 93–109, Jan. 2000.

[7] S. Mallat and W. L. Hwang, "Singularity detection and processing with wavelets," *IEEE Trans. Information Theory*, vol. 38, pp. 617–643, June 1992.

[8] N. F. Law and R. Chung, "Multiresolution discontinuity-preserving surface reconstruction," *Pattern Recogn.*, vol. 34, no. 11, pp. 2133–2144, Nov. 2001.

[9] Y. Y. Tang, L. H. Tang, and L. Feng, "Characterization and detection of edges by Lipschitz exponent and MASW wavelet transform," in *Proc. 14th Int. Conf. Pattern Recogn.*, Brisbane, Australia, Aug. 1998, pp. 1572–1574.

[10] Q. Shen, X. Liu, and D. Jiang, "Using modulus maximum pair of wavelet transform to detect spike wave of epileptic EEG," in *Proc. 20th Annu. Int. Conf. IEEE*, vol. 3, 1998, pp. 1543–1545.

[11] J. Lu, D. M. Healy Jr, and J. B. Weaver, "Contrast enhancement of medical images using multiscale edge representation," *Opt. Eng.*, vol. 33, pp. 2151–2161, 1994.

[12] T. C. Hsung and D. P. K. Lun, "Application of singularity detection for the deblocking of JPEG decoded images," *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 640–644, May 1998.

[13] A. W. C. Liew and H. Yan, "Blocking artifacts suppression in block-coded images using overcomplete wavelet representation," *IEEE Trans. Circuits Syst. Video Technol.*, Nov. 2000, submitted for publication.

[14] M. Unser, "Texture discrimination using wavelets," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recogn.*, 1993, pp. 640–641.

[15] Q. M. Tieng and W. W. Boles, "Recognition of 2D object contours using the wavelet transform zero-crossing representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 910–916, Aug. 1997.

[16] ——, "Wavelet-based affine invariant representation: A Tool for recognizing planar objects in 3D space," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 846–857, Aug. 1997.

[17] S. Mallat, "A theory of multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1989.

[18] PAI. Daubechies, *Ten Lectures on Wavelets.* Philadelphia: SIAM, 1992.

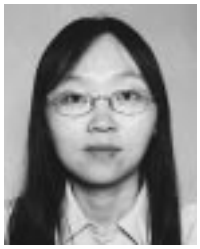
[19] M. J. Shensa, "The discrete wavelet transform: Wedding the a Trous and Mallat algorithms," *IEEE Trans. Signal Processing*, vol. 40, pp. 2464–2482, Oct. 1992.

[20] O. Rioul and P. Duhamel, "Fast algorithms for discrete and continuous wavelet transforms," *IEEE Trans. Inform. Theory*, pt. 2, vol. 38, pp. 569–586, Mar. 1992.

[21] E. Kreyszig, *Advanced Engineering Mathematics*. New York: Wiley, 1993.

[22] N. F. Law and W. C. Siu, "Fast algorithm for over-complete wavelets," *Electron. Lett.*, vol. 37, no. 4, pp. 259–261, 2001.

[23] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Applicat.*, vol. 4, pp. 247–269, 1998.



Ngai-Fong Law (M'98) received the B.Eng. degree with first-class honors from the University of Auckland, Auckland, New Zealand, in 1993 and the Ph.D. degree from the University of Tasmania, Hobart, Australia, in 1997, both in electrical and electronic engineering.

She is currently with the Electronic and Information Engineering Department, Hong Kong Polytechnic University. Her research interests include signal and image processing, wavelet transforms, image enhancement. and compression. Recently, she has also been working on Web-based system design and video searching for internet applications.

Wan-Chi Siu (SM'90) received the Associateship degree from The Hong Kong Polytechnic University (formerly called the Hong Kong Polytechnic), the M.Phil. degree from The Chinese University of Hong Kong, and the Ph.D. degree from Imperial College of Science, Technology, and Medicine, London, U.K., in 1975, 1977, and 1984, respectively.

He was with The Chinese University of Hong Kong between 1975 and 1980. He then joined The Hong Kong Polytechnic University as a Lecturer in 1980 and has became Chair Professor in 1992. He was Head of Department of Electronic and Information Engineering and subsequently Dean of the Engineering Faculty between 1994 and 2002. He is now Director of Centre for Multimedia Signal Processing of the same university. He has published over 200 research papers. His research interests include DSP, fast algorithms, transforms, wavelets, image and video coding, and computational aspects of pattern recognition and neural networks. He is a Member of the Editorial Board of the *Journal of VLSI Signal Processing Systems* for signal, image and video technology and the *EURASIP Journal on Applied Signal Processing*.

Dr. Siu was a Guest Editor of a Special Issue of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, published in May 1998, and was an Associate Editor of the same journal from 1995 to 1997. He was the general chair or the technical program chair of a number of international conferences. In particular, he was the Technical Program Chair of the IEEE International Symposium on Circuits and Systems (ISCAS'97) and the General Chair of the 2001 International Symposium on Intelligent Multimedia, Video, and Speech Processing (ISIMP'2001), which were held in Hong Kong in June 1997 and May 2001, respectively. He is now the General Chair of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'2003), which will be held in Hong Kong. Between 1991 and 1995, he was a member of the Physical Sciences and Engineering Panel of the Research Grants Council (RGC), Hong Kong Government, and in 1994, he chaired the first Engineering and Information Technology Panel to assess the research quality of 19 Cost Centers (departments) from all universities in Hong Kong. He is a Chartered Engineer and a Fellow of both the IEE and the HKIE.