A FAST AND EFFICIENT COMPUTATIONAL STRUCTURE FOR THE 2D OVER-COMPLETE WAVELET TRANSFORM

N.F. Law and W.C. Siu

Centre for Multimedia Signal Processing
Department of Electronic and Information Engineering,
The Hong Kong Polytechnic University, Hong Kong.

ABSTRACT

We studied the computational complexity of the over-complete wavelet representation for the commonly used Spline wavelet family with an arbitrary order. By deriving a general expression for the complexity, it is shown that the inverse transform is significantly more costly in computation than the forward transform. In order to reduce the computational complexity, a new spatial implementation is proposed. This new implementation exploits the redundancy between the lowpass and the bandpass outputs that is inherent to the over-complete wavelet scheme. It is shown that the new implementation can greatly simplify the computations, give an efficient inverse structure and allow the use of an arbitrary boundary extension method without affecting the ease of the inverse transform.

1. INTRODUCTION

Mallat and Zhong proposed a representation known as the over-complete wavelet representation (OCWR) for multi-scale edge-based signal characterization [1]. As points of sharp variations are one of the most important features for analyzing signal properties, this OCWR is useful in many applications, ranging from image compression, surface reconstruction to medical image processing [1-4].

Despite its capability for providing a meaningful representation, a major concern with the OCWR is its computational complexity. Unlike the sub-sampled wavelet scheme, the computational time increases linearly with the number of decomposition levels in the OCWR [1-2]. Therefore, complexity becomes a major issue in its practical implementation. In addition, it is conceived that the inverse transform is computationally more expensive than the forward transform since the reconstruction filters are always longer than the forward filters in the Spline wavelet family.

In this paper, we study the computational complexity of the OCWR. A general expression for the complexity of the Spline wavelet family with an arbitrary order is derived. It is found that the inverse transform is significantly more complicated than the forward transform. This is undesirable in applications such as surface reconstruction and image compression where the inverse transform needs to be performed [2-6]. For example, while the encoder in compression application can be very complicated, the decoder is often required to be simple.

In order to reduce computation, we use the fact that the OCWR provides a redundant image representation. This implies that correlation exists between the lowpass and the bandpass outputs. By studying this redundancy, a new spatial interpretation for the OCWR is obtained. Using this interpretation for an alternative implementation, a general expression for the complexity is derived. It is found that the proposed implementation can greatly simplify the computation and give an efficient inverse structure.

2. OVER-COMPLETE WAVELET REPRESENTATION

The OCWR for an image is obtained by applying filters to both the horizontal and the vertical directions. There are three outputs from a single level decomposition: the lowpass approximation of the original image and two bandpass outputs. Mathematically, the lowpass output is given by,

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

Two bandpass outputs are written 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 lowpass filter, the bandpass filter and the original image. In reconstruction, the original image is given as,

$$\begin{aligned} X_0(z_1,z_2) = {} & K(z_1)L(z_2)D_1^1(z_1,z_2)+ \\ & L(z_1)K(z_2)D_1^2(z_1,z_2)+\overline{H}(z_1)\overline{H}(z_2)X_1(z_1,z_2) \end{aligned} \tag{4}$$

where $\overline{H}(z)$ is the time reverse of $H(z)$ and $K(z)$ and $L(z)$ are the bandpass reconstruction filters. To achieve perfect reconstruction, it is required that,

$$\begin{aligned} & K(z_1)L(z_2)G(z_1)+L(z_1)K(z_2)G(z_2)+ \\ & \overline{H}(z_1)\overline{H}(z_2)H(z_1)H(z_2)=1 \end{aligned} \tag{5}$$

Eqn.5 is a necessary and sufficient condition for perfect reconstruction. Unlike the sub-sampling scheme, there is considerable freedom in choosing these four filters.

Mallat and Zhong have constructed the wavelet function in such a way that it is the derivative of a smoothing function [1]. The local extrema of the resultant transform then characterize the multiscale edges in the image. The set of wavelet functions is commonly known as the Spline wavelet family. For an order n, the lowpass and the bandpass filters can be expressed respectively as,

$$H(z)=\frac{z^n}{2^{2n+1}}\left(1+z^{-1}\right)^{2n+1}$$

and $$G(z) = 2\left(z^{-1} - 1\right) \quad (6)$$

The reconstruction filters can be shown to be,

$$K(z) = \frac{z-1}{8} \sum_{k=0}^{2n} \left[\frac{1}{2}\left(z^{1/2} + z^{-1/2} \right) \right]^{2k} \quad (7)$$

$$L(z) = \frac{1}{2}\left[1 + \frac{1}{2^{4n+2}} \left(z^{1/2} + z^{-1/2} \right)^{4n+2} \right] \quad (8)$$

Comparing the forward and the inverse filters shown in eqn.6 to eqn.8, it can be seen that the numbers of filter coefficients for $G(z)$, $H(z)$, $K(z)$ and $L(z)$ are 2, $2n+2$, $4n+2$ and $4n+3$ respectively. As the inverse filters are significantly longer than the forward filters, the computational complexity associated with the inverse is much higher than that with the forward transform.

To analyze the computational complexity, the number of additions and multiplications are calculated. This complexity metric is of interest for both hardware and software realization. The complexity for $G(z)$ is given by,

$$Complexity\ (G) = Cost_{add} + Cost_{multiply} \quad (9)$$

where $Cost_{add}$ and $Cost_{multiply}$ define the costs in having an addition and a multiplication operations respectively. Using the Binomial theorem, $H(z)$ in eqn.6 can be expanded as follows,

$$H(z) = \frac{1}{2^{2n+1}} \sum_{k=0}^{n} C_k^{2n+1} \left[z^{n-k} + z^{-n+k-1} \right] \quad (10)$$

where $C_k^m = m!/k!(m-k)!$. Analyzing eqn.10 gives that,

$$Complexity\ (H) = (2n+1)Cost_{add} + (n+1)Cost_{multiply} \quad (11)$$

In reconstruction, the complexities of $\overline{H}(z)$, $K(z)$ and $L(z)$ need to be determined. $\overline{H}(z)$ can be obtained from eqn.10 simply by replacing z with z^{-1}. The complexity of $\overline{H}(z)$ thus equals to that of $H(z)$. For $K(z)$, we expand the summations in eqn.7 as follows,

$$K(z) = B_{2n}\left[z^{2n+1} - z^{-2n} \right] + \sum_{k=1}^{2n} E_k \left[z^k - z^{-k+1} \right] \quad (12)$$

where

$$B_m = \frac{1}{8} \sum_{k=m}^{2n} \frac{1}{2^{2k}} C_{k-m}^{2k} \quad \text{and} \quad E_i = B_{i-1} - B_i \quad (13)$$

Eqn.12 shows that the complexity of $K(z)$ equals to,

$$Complexity\ [K] = (4n+1)Cost_{add} + (2n+1)Cost_{multiply} \quad (14)$$

Employing the Binomial theorem, an expression for $L(z)$ can be obtained through expanding eqn.8 as follows,

$$L(z) = P_1 + \frac{1}{2^{4n+3}} \sum_{k=0}^{2n} \left\{ C_k^{4n+2} \left[z^{2n+1-k} + z^{-2n-1+k} \right] \right\} \quad (15)$$

where

$$P_1 = \frac{1}{2}\left[1 + \frac{1}{2^{4n+2}} C_{2n+1}^{4n+2} \right] \quad (16)$$

Analyzing eqn.15 shows that,

$$Complexity[L] = (4n+2)Cost_{add} + (2n+2)Cost_{multiply} \quad (17)$$

A one level forward transform involves filtering in both horizontal and vertical directions (cf. eqn.1 to eqn.3). Substituting filters expressions in eqn.7 and eqn.10 to eqn.1, eqn.2 and eqn.3, we obtain,

$$X_1(z_1, z_2) = \frac{1}{2^{4n+2}} \sum_{k_1=0}^{n} C_{k_1}^{2n+1} \left[z_1^{n-k_1} + z_1^{-n+k_1-1} \right] \times \sum_{k_2=0}^{n} C_{k_1}^{2n+1} \left[z_2^{n-k_2} + z_2^{-n+k_2-1} \right] X_0(z_1, z_2) \quad (18)$$

Two multiplications are merged into one in eqn.18, thus the complexity in obtaining $X_1(z_1, z_2)$ equals to,

$$Complexity\ [X_1] = (4n+2)Cost_{add} + (2n+1)Cost_{multiply} \quad (19)$$

The bandpass outputs can be obtained by substituting eqn.6 to eqn.2 and eqn.3, their complexities then equal to,

$$Complexity\left[D_1^1 \right] = Complexity\left[D_1^2 \right] = Cost_{add} + Cost_{multiply} \quad (20)$$

The complexity of the forward transform can thus be calculated by summing up the complexities in eqn.19 and eqn.20, i.e.,

$$Complexity_{FB}[Forward] = (4n+4)Cost_{add} + (2n+3)Cost_{multiply} \quad (21)$$

Similarly, the complexity of the inverse transform is calculated by summing up the complexities in $K(z_1)L(z_2)D_1^1(z_1, z_2)$, $L(z_1)K(z_2)D_1^2(z_1, z_2)$ and $\overline{H}(z_1)\overline{H}(z_2)X_1(z_1, z_2)$. The complexity of $\overline{H}(z_1)\overline{H}(z_2)X_1(z_1, z_2)$ is same as that of $H(z_1)H(z_2)X_0(z_1, z_2)$ and is given in eqn.19. For either $K(z_1)L(z_2)D_1^1(z_1, z_2)$ or $L(z_1)K(z_2)D_1^2(z_1, z_2)$, the complexity equals to the sum of the complexities in $K(z)$ and $L(z)$, i.e.,

$$(8n+3)Cost_{add} + (4n+2)Cost_{multiply} \quad (22)$$

Then the inverse complexity is found to be,

$$Complexity_{FB}[Inverse] = (20n+10)Cost_{add} + (10n+5)Cost_{multiply} \quad (23)$$

Comparing eqn.21 and eqn.23, the inverse transform has a significantly higher complexity (nearly five times) than the forward transform. This is undesirable and we need to reduce its computational complexity.

3. The NEW FORMULATION

The OCWR provides a redundant representation for an image. Correlation exists between the lowpass and the bandpass outputs which can be exploited in our calculation of the wavelet transform. Indeed, many applications, such as discontinuity-preserving surface reconstruction, contrast enhancement and denoising, have benefited from this correlation in solving their problems [1,3-6]. In order to study the correlation, we consider the following expression from the 1D study [7].

1D Formulation: The expression

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

can be rewritten as,

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

where

$$D_1(z) = 2\left(z^{-1} - 1\right)X_0(z) \quad \text{and} \quad A_k = \sum_{m=0}^{k} C_m^{2n+1}$$

Using the 1D formulation, we can derive an alternative implementation structure for the OCWR.

Alternative Implementation: The lowpass output of the over-complete 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

$$F_1\left[D_1^2(z_1,z_2)\right] = Y_1(z_1,z_2) = \left\{2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2}\left[z_2^{-k_2} - z_2^{k_2}\right]\right\}D_1^2(z_1,z_2) \tag{24}$$

and

$$F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right] = \left\{2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1}\left[z_1^{-k_1} - z_1^{k_1}\right]\right\} \times \left[D_1^1(z_1,z_2) + \frac{\left(z_1^{-1} - 1\right)}{2^{2n+1}} Y_1(z_1,z_2)\right] \tag{25}$$

Proof: Using Lemma 1, it can be written that,

$$H(z_2)X_0(z_1,z_2) = X_0(z_1,z_2) + \frac{1}{2^{2n+2}}\left\{2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2}\left[z_2^{-k_2} - z_2^{k_2}\right]\right\}D_1^2(z_1,z_2) \tag{26}$$

then,

$$\begin{aligned} X_1(z_1,z_2) &= X_0(z_1,z_2) + \frac{1}{2^{2n+2}}\left\{2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1}\left[z_1^{-k_1} - z_1^{k_1}\right]\right\}D_1^1(z_1,z_2) + \\ &\frac{D_1^2(z_1,z_2)}{2^{4n+3}}\left\{\sum_{k_1=0}^{n} C_{k_1}^{2n+1}\left[z_1^{n-k_1} + z_1^{-n+k_1-1}\right]\right\} \times \\ &\left\{2^{2n} + \sum_{k_2=1}^{n} A_{n-k_2}\left[z_2^{-k_2} - z_2^{k_2}\right]\right\} \end{aligned} \tag{27}$$

Note that the last term in eqn.27 can be rewritten as,

$$\frac{1}{2^{2n+2}} Y_1(z_1,z_2) + \frac{1}{2^{4n+4}}\left[2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1}\left[z_1^{-k_1} - z_1^{k_1}\right]\right] 2\left(z_1^{-1} - 1\right) Y_1(z_1,z_2) \tag{28}$$

The lowpass output can then be expressed as,

$$\begin{aligned} X_1(z_1,z_2) &= X_0(z_1,z_2) + \frac{1}{2^{2n+2}} Y_1(z_1,z_2) + \\ &\frac{1}{2^{2n+2}}\left\{2^{2n} + \sum_{k_1=1}^{n} A_{n-k_1}\left[z_1^{-k_1} - z_1^{k_1}\right]\right\} \times \\ &\left[D_1^1(z_1,z_2) + \frac{\left(z_1^{-1} - 1\right)}{2^{2n+1}} Y_1(z_1,z_2)\right] \end{aligned} \tag{29}$$

which completes the proof. The alternative implementation provides a way to relate the lowpass output, the original image and the two bandpass outputs. It provides not only an alternative implementation scheme for the forward transform, but also simplifies the computation of the inverse transform. In particular, the inverse can be easily calculated as follows,

$$X_0(z_1,z_2) = X_1(z_1,z_2) - \frac{1}{2^{2n+2}} \times \left\{F_1\left[D_1^2(z_1,z_2)\right] + F_2\left[D_1^1(z_1,z_2), Y_1(z_1,z_2)\right]\right\} \tag{30}$$

The proposed implementation for both the forward and the inverse transforms is shown in Figure 1. In analyzing the complexity, the complexities in F_1 and F_2 can be written as,

$$Complexity\ [F_1] = (2n)Cost_{add} + (n)Cost_{multiply} \tag{31}$$

$$Complexity[F_2] = (2n+2)Cost_{add} + (n+1)Cost_{multiply} \tag{32}$$

Using eqn.31 and eqn.32, the forward and the inverse complexities are found to be,

$$Complexity_S[Forward] = (4n+6)Cost_{add} + (2n+4)Cost_{multiply} \tag{33}$$

and

$$Complexity_S[Inverse] = (4n+4)Cost_{add} + (2n+2)Cost_{multiply} \tag{34}$$

4. ANALYSIS

In the filtering approach, the complexity of the forward transform is much lower than that of the inverse transform (eqn.21 and eqn.23). In contrast, the complexity of the forward transform is slightly higher than that of the inverse transform in the proposed approach (eqn.33 and eqn.34). Comparing the filtering and the proposed approaches, we can 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.

In the proposed approach, no filtering is required for the reconstruction of the original image. Rather, a simple spatial implementation is used for the reconstruction. The computational complexity of the inverse transform is thus 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.

Figure 2 shows a comparison of the computational complexity between the filtering and the proposed spatial approaches for different *n* orders. The saving in computation is significant. For the quadratic Spline wavelet (n=1), the number of additions reduces from 30 to 8 whereas the number of multiplications reduces from 15 to 4. This gives a saving of 73.3% for both the additions and multiplications. For the cubic Spline wavelet (n=2), the number of additions reduces from 50 to 12 whereas the number of multiplications reduces from 25 to 6. This gives a saving of 76.0% for both the additions and multiplications.

Besides the decrease in computational complexity, the proposed scheme handles the boundary in a nice way. Using the filtering approach, there are two ways to deal with the boundary extension problem: the signal is either symmetrically extended before filtering or the boundary is corrected after the inverse transform. The former would increase the computational time especially for a large image while the latter involves the design

of non-trivial boundary correction rules and is filter dependent. However, using our proposed scheme, any arbitrary boundary extension scheme can be used without affecting the ease of the inverse transform. The prediction terms, F_1 and F_2, always remain the same in both the forward and the inverse transforms. There is thus no need for boundary correction after reconstruction. The image can always be reconstructed perfectly.

5. CONCLUSIONS

By deriving a general expression for the computational complexity of the over-complete wavelet representation, it is found that the inverse transform is significantly more complicated than the forward transform using the filtering implementation structure. In order to reduce the computational complexity, a new spatial implementation is proposed. This implementation exploits the correlation between the lowpass and the bandpass outputs. It is shown that the new implementation can greatly simplify the computation, give an efficient inverse transform and allow the use of an arbitrary boundary extension method.

6. ACKNOWLEDGMENTS

This work is supported by the Research Grant Council of the HKSAR, under Grant PolyU5133/02E and the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University.

7. REFERENCES

[1] S. Mallat and S. Zhong, "Characterization of Signals from Multiscale Edges", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 14, 710-732, 1992.

[2] A.W.C. Liew and N.F. Law, "Reconstruction from 2D Wavelet Transform Modulus Maxima Using Projection", *IEE Proceedings - Vision, Image and Signal Processing,* Vol. 147, 176-184, 2000.

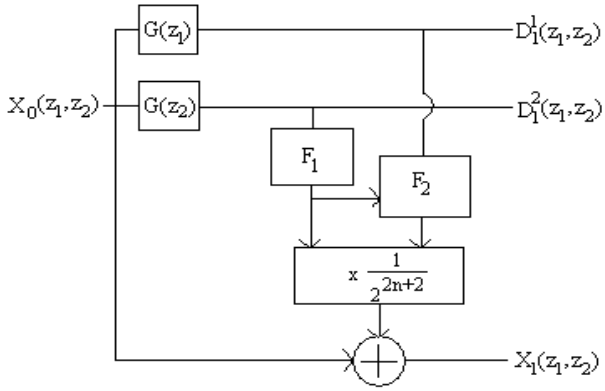
[3] N.F. Law and R. Chung, "Multiresolution Discontinuity-Preserving Surface Reconstruction", *Pattern Recognition*, Vol.34, 2133-2144, 2001.

[4] J. Lu and D.M. Healy, Jr. and J.B. Weaver, "Contrast Enhancement of Medical Images Using Multiscale Edge Representation", *Optical Engineering*, Vol. 33, 2151-2161, 1994.

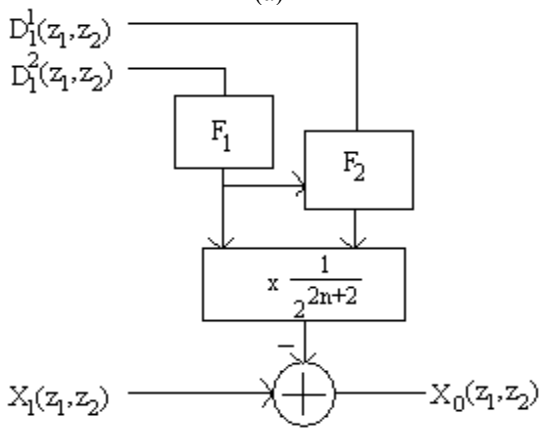
[5] Quang Minh Tieng and W.W. Boles, "Recognition of 2D Object Contours Using the Wavelet Transform Zero-Crossing Representation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, 910-916, 1997.

[6] A.W.C. Liew, N.F. Law and D.T. Nguyen, "Direct Reconstruction Method for Wavelet Transform Extrema Representation", *IEE Proceedings - Vision, Image and Signal Processing*, Vol. 144, 193-198, 1997.

[7] N.F. Law and W.C. Siu, 'Fast Algorithm for Over-Complete Wavelets', *Electronics Letters*, Vol. 37, No. 4, 259-261, 15th Feb 2001, U.K.

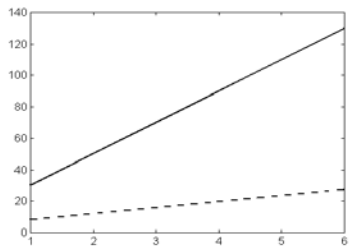


(a)

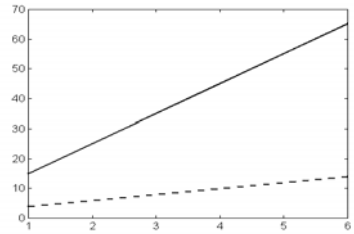


(b)

Figure 1: New spatial domain approach, (a) forward and (b) inverse.



(a)



(b)

Figure 2: Plots of (a) the number of addition and (b) the number of multiplication for different order *n*. The solid line is the filtering approach and the dotted line is the new spatial approach.