Journal of Engineering and Applied Sciences 3 (6): 446-450, 2008
ISSN: 1816-949X

A Novel Computing Method for 3D Double Density DWT

Hadeel N. Abduallah, Ahmed N. Abdallah and Ali Kareem
Department of Electrical and Electronic, Engineering University of Technology, Baghdad, Iraq

Abstract: The relatively new field of framelets shows promise in removing some of the limitations of wavelets. Several applications have benefited from the use of frames, for example, denoising and signal coding. In this study, 3-D double density transform algorithm for computing advance transforms are proposed. The propose method reduces heavily processing time for decomposition of video sequences keeping or overcoming the quality of reconstructed sequences In addition, it cuts heavily the memory demands. Also, the inverse procedures of the above transform for multi-dimensional cases are verified.

Key words: Novel computing method, density, DWT, SWT, WT

INTRODUCTION

Though standard DWT is a powerful tool for analysis and processing of many real-world signals and images, it suffers from 3 major disadvantages, Shift- sensitivity, Poor directionality and Lack of phase information. These disadvantages severely restrict its scope for certain signal and image processing applications (Shukla, 2003).

Other extensions of standard DWT such as Wavelet Packet Transform (WP) and Stationary Wavelet Transform (SWT) reduce only the first disadvantage of shift sensitivity but with the cost of very high redundancy and involved computation. Recent research suggests the possibility of reducing 2 or more above mentioned disadvantages using different forms of Wavelet Transforms (Fernandes, 2002; Spaendonck *et al*., 2003; Fernandes *et al*., 2000) with only limited (and controllable) redundancy and moderate computational complexity.

Frames, or overcomplete expansions, have a variety of attractive features. With frames, better time-frequency localization can be achieved than is possible with bases. Some wavelet frames can be shift invariant, while wavelet bases cannot be. Frames provide more degrees of freedom to carry out design. There are a number of methods of generating practical frames (Selesnick, 2001). The undecimated DWT (UDWT) generates a wavelet frame from an existing wavelet basis by removing the subsampling from an existing critically sampled filter bank (Rioul and Duhamel, 1992). A wavelet frame can be obtained by taking the union of 2 (or more) bases. This can be implemented with 2 independent filter banks operating in parallel. Kingsbury has shown the advantages of dual-tree DWTs (Kingsbury, 1998). A wavelet frame can also be obtained by iterating a suitably designed oversampled filter bank as developed in (Chui and He, 2000), for example. This is the type of frame to be considered in this study.

The 2D wavelet transform has been used for compressing video (Conte *et al*., 2000) as well. However, 3 dimensional (3D) compression techniques seem to offer better results than 2 dimensional (2D) compression techniques which operate in each frame independently. Muraki introduced the idea of using 3D wavelet transform to efficiently approximate 3D volumetric data (Muraki, 1992). However, there are still some aspects of the 3-D geometry-based coding schemes that can be improved. First, the scene geometry information and the image data must be encoded separately. This requirement limits the flexibility of the coding scheme, since the decoding of the 3-D geometry information must be completed prior to the decoding of the image (texture) data. Second, the generally used 3-D geometry representations, such as the mesh model used in existing 3-D geometry-based multi-view coding schemes, are suitable to represent 3-D objects of simple surface but difficult to represent objects of complicated surface, which are often shown in natural scenes. Third, the whole procedure of obtaining 3-D geometry information is computationally complex (Bernab *et al*., 2002).

This study describes new wavelet tight frames based on iterated oversampled FIR filter banks, first introduced in (Selesnick and Sendur, 2000). Selesnick and Sendur (2000) introduce the double-density wavelet transform (DDWT) as the tight-frame equivalent of Daubechies' orthonormal wavelet transform; the wavelet filters are of

Corresponding Author: Hadeel N. Abduallah, Department of Electrical and Electronic, Engineering University of Technology, Baghdad, Iraq

minimal length and satisfy certain important polynomial properties in an oversampled framework. Because the DDWT, at each scale, has twice as many wavelets as the DWT, it achieves lower shift sensitivity than the DWT. New fast computation algorithms for computing discrete double density wavelet transform have been described in this study, in a simple and easy to verify procedure based on iterated FIR filter bank that simplify computation complexity by using simple operations like matrix multiplication and addition.

DOUBLE DENSITY DWT

Double density DWT are very similar to wavelets but have some important differences. In particular, whereas wavelets have an associated scaling function ψ (t) and wavelet function ψ (t), double density DWT have one scaling function ψ (t) and 2 wavelet functions ψ1 (t) and ψ2 (t).

The scaling function Φ (t) and the wavelets ψ1 (t) and ψ2 (t) are defined through these equations by the low-pass (scaling) filter h_0 (n) and the 2 high-pass (wavelet) filters h_1 (n) and h_2 (n). Let

$$\phi(t)=\sqrt{2}\sum_{n} h_0(n)\ \ \phi(2t-n),\qquad \psi_i(t)=\sqrt{2}\sum_{n} h_i(n)\ \ \phi(2t-n),\quad i=1,\,2. \tag{1}$$

Any function f (t) could be written as a series expansion in terms of the scaling function and wavelets by (Selesnick, 2001):

$$f(t)=\sum_{k=-\infty}^{\infty} c(k)\phi_k(t)+\sum_{j=0}^{\infty}\sum_{k=-\infty}^{\infty} \begin{matrix} d_1(j,k)\psi_{1,j,k}(t)+ \\ d_2(j,k)\psi_{2,j,k}(t)\end{matrix} \tag{2}$$

where,

$$c(k)=\int f(t)\phi_k(t)dt,\qquad d_i(j,k)=\int f(t)\psi_{i,j,k}(t)dt,\quad i=1,2 \tag{3}$$

In this expansion, the first summation gives a function that is a low resolution or coarse approximation of f (t) at scale j = 0. For each increasing j in the second summation, a higher or finer resolution function is added, which adds increasing details.

The filters h_i (n) and h_i (-n) should satisfy the perfect reconstruction (PR) conditions. From basic multirate identities, the PR conditions are the following (Selesnick and Sendur, 2000):

$$H_0(z).H_0\left(\frac{1}{z}\right)+H_1(z).H_1\left(\frac{1}{z}\right)+H_2(z).H_2\left(\frac{1}{z}\right)=2 \tag{4}$$

and

$$H_0(-z).H_0\left(\frac{1}{z}\right)+H_1(-z).H_1\left(\frac{1}{z}\right)+H_2(-z).H_2\left(\frac{1}{z}\right)=0 \tag{5}$$

Let K_0 denote the number of z eros H_0 (e^{jw}) has at w = π. For i = 1, 2, let K_i denote the number of zeros H_i (e^{jw}) has at w = 0. Then the Z-transform of each h_i (n) factors as follows:

$$H_0(z) = Q_0(z)\,(z+1)^{K_0} \tag{6}$$

$$H_1(z) = Q_1(z)\,(z+1)^{K_1} \tag{7}$$

$$H_2(z) = Q_2(z)\,(z+1)^{K_2} \tag{8}$$

K_0 denotes the degree of polynomials representable by integer translates of ψ (t) and is related to the smoothness of ψ (t). K_1 and K_2 denote the number of zero moments of the wavelets filters h_1 (n) and h_2 (n), provided $K_0>K_1$ and $K_0\geq K_2$. If it is desired for a given class of signals that the wavelets have 2 zero moments (for example), then the remaining degrees of freedom can be used to achieve a higher degree of smoothness by making K_0 greater than K_1 and K_2. Although, the values K_i need not all be equal, there is still the constraint (Selesnick, 2001; Selesnick and Sendur, 2000):

$$\text{Length } h_0 \geq K_0+\min(K_1, K_2) \tag{9}$$

So, the minimum length of h_0 is K_0+ min (K_1, K_2). In the orthonormal case $K_0 = K_1$ and K2 = ∞ (as h_2 = 0), which gives the minimum length of h_0 to be $2K_0$, which is consistent with Daubechies orthonormal filters.

THE PROPOSED METHOD

The double density DWT is implemented on discrete-time signals using the over sampled analysis and synthesis filter bank shown in Fig. (1). The analysis filter bank consists of 3 analysis filters- one low pass filter denoted by h_0 (n) and 2 distinct high pass filters denoted h_1 (n) and h_2 (n). As the input signal X (N) travels through the system, the analysis filter bank decomposes it into 3

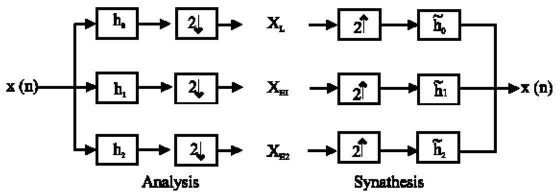

Fig. 1: Analysis and synthesis stages of a 1-D single level double-density DWT

sub bands, each of which is then down-sampled by by 2. From this process X_L (N/2), X_{H1} (N/2) and X_{H2} (N/2) are generated, which represent the low frequency (or coarse) subband and the 2 high frequency (or detail) sub bands, respectively.

The sampled signals are filtered by the corresponding synthesis low pass h_0^* (n) and 2 high pass h_1^* (n) and h_2^* (n) filters and then added to reconstruct the original signal. Note that the filters in the synthesis stage, are not necessary the same as those in the analysis stage. For an orthogonal filter bank, h_i^* (n) are just the time reversals of h_i (n).

Wavelet frames, having the form described above, have twice as many wavelets than is necessary. Yet note that the filter bank illustrated in Fig. 1 is oversampled by 3/2, not by 2. However, if the filter bank is iterated a single time on its lowpass branch (h_0), the total oversampling rate will be 7/4. For a 3-stage filter bank, the oversampling rate will be 15/8. When this filter bank is iterated on its lowpass branch indefinitely, the total oversampling rate increases toward 2, which is consistent with the redundancy of the frame for L_2 (R).

For computing discrete double density transform consider the following transformation matrix for length-6:

$$W=\begin{bmatrix}
h_0(0) & h_0(1) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & 0 & 0 & 0 & \cdots & 0 & 0\\
0 & 0 & h_0(0) & h_0(0) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & 0 & \cdots & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
h_0(2) & h_0(3) & h_0(4) & h_0(5) & 0 & 0 & 0 & 0 & 0 & \cdots & h_0(0) & h_0(1)\\
h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & 0 & 0 & 0 & \cdots & 0 & 0\\
0 & 0 & h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & 0 & \cdots & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
h_1(2) & h_1(3) & h_1(4) & h_1(5) & 0 & 0 & 0 & 0 & 0 & \cdots & h_1(0) & h_1(1)\\
h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & h_1(5) & 0 & 0 & 0 & \cdots & 0 & 0\\
0 & 0 & h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & h_1(5) & 0 & \cdots & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
h_2(2) & h_2(3) & h_2(4) & h_1(5) & 0 & 0 & 0 & 0 & 0 & \cdots & h_2(0) & h_2(1)
\end{bmatrix}_{\frac{3N}{2}xN} \tag{10}$$

Here blank entries signify zeros and for length-10 become:

$$W=\begin{bmatrix}
h_0(0) & h_0(1) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & h_0(7) & h_0(8) & h_0(9) & 0 & 0 & \cdots & 0 & 0\\
0 & 0 & h_0(0) & h_0(1) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & h_0(7) & h_0(8) & h_0(9) & \cdots & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\
h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & h_0(7) & h_0(8) & h_0(9) & 0 & 0 & 0 & 0 & \cdots & h_0(0) & h_0(1)\\
h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & h_1(7) & h_1(8) & h_1(9) & 0 & 0 & \cdots & 0 & 0\\
0 & 0 & h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & h_1(7) & h_1(8) & h_1(9) & \cdots & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\
h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & h_1(7) & h_1(8) & h_1(9) & 0 & 0 & 0 & 0 & \cdots & h_1(0) & h_1(1)\\
h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & h_2(5) & h_2(6) & h_2(7) & h_2(8) & h_2(9) & 0 & 0 & \cdots & 0 & 0\\
0 & 0 & h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & h_2(5) & h_2(6) & h_2(7) & h_2(8) & h_2(9) & \cdots & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\
h_2(2) & h_2(3) & h_2(4) & h_2(5) & h_2(6) & h_2(7) & h_2(8) & h_2(9) & 0 & 0 & 0 & 0 & \cdots & h_2(0) & h_2(1)
\end{bmatrix}_{\frac{3N}{2}xN} \tag{11}$$

A 2-D separable transform is equivalent to 2 1-D transforms in series. It is implemented as 1-D row transform followed by a 1-D column transform on the data obtained from the row transform. To compute a single-level 2-D discrete double density transform using nonseparable method, the next steps should be followed:

- Checking input dimensions: Input matrix should be of length N×N, where N must be even and N > = length of analysis filters.
- For an N×N matrix input 2-D signal, X, construct a 3N/2×N transformation matrix, W, using transformation matrices given in Eq. (10) and (11).
- Apply Transformation by multiplying the transformation matrix by the input matrix by the transpose of the transformation matrix.

$$Y = W.\ X.\ W^T$$

This multiplication of the 3 matrices result in the final discrete double density transformed matrix.

For a 2-D double density transformation, the algorithm is applied in x-direction first and then in y-direction. Similarly, in 3-D double density transformation the structures are defined in 3-D and the transformation algorithm is applied in x-, y- and z-direction successively. One cycle for an n-dimensional data set is defined as the completion of the algorithm for all n directions. Let's take a general 3-D signal, for example any N×N×M matrix and apply the following steps:

- Checking input dimensions: Input matrix should be of length N×N×M, where N, M are even and min (N, M) > length of analysis filter.
- Apply 2-D double density transform algorithm to each N×N input matrix, which result in a $\left(\frac{3N}{2}\times\frac{3N}{2}\times M\right)$ matrix.
- Apply 1-D double density transform algorithm to each of the $\left(\frac{3N}{2}\times\frac{3N}{2}\right)$ elements in all M matrices in z-direction, which can be done as follows:
- For each i, j construct the Mx1 input vector Y (i, j) = $[a_{i,j}, b_{i,j}, c_{i,j}, d_{i,j}]^t_{1\times M}$ where, i, j = 0, 1, 2, ..., 3N/2
- Construct an (3M/2×M) transformation matrix; using transformation matrices given in Eq. (10) and (11).
- Apply matrix multiplication to the (3M/2×M) constructed transformation matrix by the M×1 input vector.
- Repeat step 3 for all i, j to get YY matrix $\left(\frac{3N}{2}\times\frac{3N}{2}\times\frac{3M}{2}\right)$ matrix.

Fast Computation Method of 3-D Inverse double density transform:

- Let Y be the $\left(\frac{3N}{2}\times\frac{3N}{2}\times\frac{3M}{2}\right)$ double density transformed matrix.
- Construct (M× 3M/2) reconstruction matrix, $T = W^T$, using transformation matrices given in Eq. (10) and (11).
- Apply 1-D inverse double density transform algorithm to eache $\frac{3N}{2}\times\frac{3N}{2}$ element in all 3m/2 matrices in z-direction.
- Construct $N\times\frac{3N}{2}$ reconstruction matrix, $T = W^T$, using transformation matrices given in (10) and (11).
- Apply 2-D Inverse double density transform algorithm to each $\frac{3N}{2}\times\frac{3N}{2}$ matrix result from step 3, which can be done by reconstruction of the input matrix by multiplying the reconstruction matrix by the input matrix by the transpose of the reconstruction matrix. $X= T.\ Y_o\ .T^t$.

A COMPUTER TEST

A general computer program computing a single-level 3-D double density DWT is written using MatLab V.7.0 for a general N×N×M 3-D signal. An example test is applied to yosemite. As shown in Fig. 2a, the original yosemite image dimensions are 256×256×8 (N×N×M). After a single-level 2-D of double density decomposition using a set of filter, image dimensions will be a matrix of 375×375×8 (3N/2×3N/2×M) as shown in Fig. 2b. After a single-level 1-D of double density decomposition using the same filter image dimensions will be a matrix of 375×375×12 (3N/2×3N/2×3M/2) as shown in Fig. 2c. An example test is applied to the decomposed yosemite image to reconstruct the original yosemite image by using a general computer program computing a single-level 3-D double density transform and the result is shown in Fig. 2d.

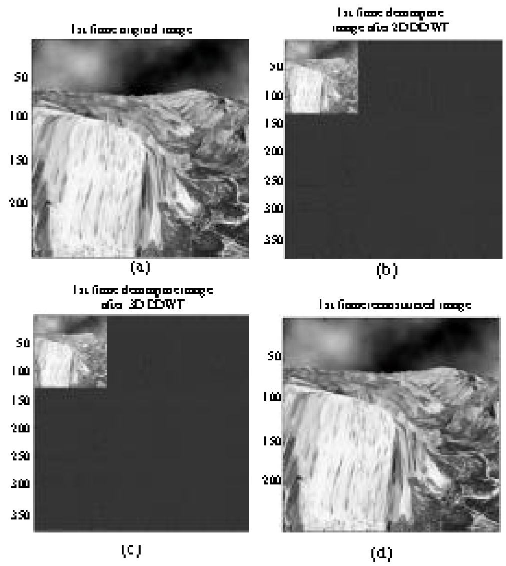

Fig. 2: Yosemite image Sequence, (a) 1'st Frame from Original, (b) After 2D double density transform (c) After 3D double density transform (d) 1'st frame reconstructed image

CONCLUSION

This study presents a 3-D double density transform computation methods that verify the potential benefits of framelets and gain a much improvement in terms of low computational complexity. The new proposed 3-D double density transform algorithm reduces heavily the processing time for decomposition of video sequences keeping or overcoming the quality of reconstructed sequences In addition, it cuts heavily the memory demands.

REFERENCES

Bernab´e, G., J. Gonz´alez, J.M. Garc´ýa and J. Duato, 2002. Memory conscious 3d wavelet transform. In: Proc. 28th Euromicro Conference. Multimedia and Telecommunications. Dortmund, Germany.

Chui, C. and W. He, 2000. Compactly supported tight frames associated with refinable functions. Appl. Comput. Harmon. Anal., 8: 293-319.

Conte, G., S. Tommesani and F. Zanichelli, 2000. The long and winding road to high-performance image processing with MMX/SSE. In: Proc. 5th IEEE Int. Workshop on Computer Architectures for Machine Perception.

Fernandes, F. *et al.*, 2000. Directional complex wavelet processing. In: Wavelet Applications VII, Proc. of SPIE.

Fernandes, F., 2002. Directional, shift-insensitive, complex wavelet transforms with controllable Redundancy. PhD Thesis, Rice University.

Kingsbury, N.G., 1998. The Dual-Tree Complex Wavelet Transform: A new technique for shift invariance and directional filters. In: Proc. Eighth IEEE DSP Workshop, Utah, pp: 9-12.

Muraki, S., 1992. Approximation and rendering of volume data using wavelet transforms. In: Proc. Visualization, pp: 21-28.

Rioul, O. and P. Duhamel, 1992. Fast algorithms for discrete and continuous wavelet transforms. IEEE Trans. Inform. Theory, 38: 569-586.

Selesnick, I.W. and L. Sendur, 2000. Iterated oversampled filter banks and wavelet frames. In: Wavelet Applications in Signal and Image Processing VIII, San Diego, 2000, Proc. SPIE, pp: 4119.

Selesnick, I.W., 2001. Smooth wavelet tight frames with zero moments Applied and Computational Harmonic Analysis (ACHA), 10 (2,): 163-181.

Shukla, P.D., 2003. Complex wavelet transform and their applications. M.Phil. Thesis, Department of Electronic and Electrical Engineering, University of Strathclyde, Scotland.

Spaendonck, R.R. *et al.*, 2003. Orthogonal Hilbert transform filterbanks and wavelets. ICASSP03, pp: 6-10.