Wavelet Algorithms for Remote Sensing

Gee Ju CHAE
Telematics Research Division, ETRI
161 Gajeong-dong, Yuseong-Gu, Daejeon, 305-350, KOREA
cbase@etri.re.kr

Kyoung Ho CHOI
Telematics Research Division, ETRI
161 Gajeong-dong, Yuseong-Gu, Daejeon, 305-350, KOREA
khchoietri@etri.re.kr

Abstract: From 1980's, the DWT(Discrete Wavelet Transform) is applied to the data/image processing. Many people use the DWT in remote sensing for diversity purposes and they are satisfied with the wavelet theory. Though the algorithm for wavelet is very diverse, many people use the standard wavelet such as Daubechies D4 wavelet and biorthogonal 9/7 wavelet. We will overview the wavelet theory for discrete form which can be applied to the image processing. First, we will introduce the basic DWT algorithm and review the wavelet algorithm: EZW (Embedded Zerotree Wavelet), SPIHT(Set Partitioning in Hierarchical Trees), Lifting scheme, Curvelet, etc. Finally, we will suggest the properties of wavelet algorithms and wavelet filter for each image processing in remote sensing.

Keywords: Remote Sensing, Wavelets, DWT, Change Detection, EZW, SPIHT

1. Introduction

In remote sensing, we need to transform the satellite images efficiently for change detection, geometric corrections, etc. The standard method is transforming the data in the time domain into frequency domain. The FFT(Fast Fourier Transform) and DCT(Discrete Cosine Transform) is used in diverse field for processing the image data for past years. From 1980's, the DWT(Discrete Wavelet Transform)[1] is applied to the data/image processing. Many people use the DWT in remote sensing for diversity purposes and they are satisfied with the wavelet theory. Though the algorithm for wavelet is very diverse, many people use the standard wavelet such as Daubechies D4 wavelet[1] and biorthogonal 9/7 wavelet [2]. We will overview the wavelet theory for discrete form which can be applied to the image processing. First, we will introduce the basic DWT(which has relation with MRA (MultiResolution Analysis)). We will also review the wavelet algorithm: EZW[3] (Embedded Zerotree Wavelet), SPIHT[4](Set Partitioning in Hierarchical Trees), Lifting scheme[5], Complex wavelet transform(DT CWT)[6], Curvelet[7], etc . Finally, we will suggest the properties of wavelet algorithms and wavelet filter for each image processing in remote sensing.

2. DWT(Discrete Wavelet Trnasform)

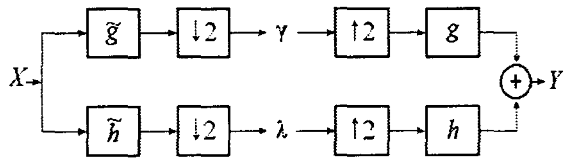

Fig. 1. DWT.

For the filter bank in Figure 1, the conditions for perfect reconstruction(this means that X=Y) are now given by [1] as

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

In geneal, $\tilde{h}$ ($\tilde{g}$) is low-pass(high-pass) filter and $\tilde{g}$ is determined by $\tilde{h}$. If $(\tilde{h},h)$ satisfies above equation, we say $(\tilde{h},h)$ is **biorthogonal filter**. The well-known biorthogonal filter is biorthogonal 9/7 wavelet. If $\tilde{h} = h$, then we say $\tilde{h}$ is **orthogonal filter**. The well-known orthogonal wavelet filter is Daubechies D4 wavelet.

The DWT means that we obtain $\gamma.\lambda$ from X. Many wavelet algorithms process the $\gamma.\lambda$ and obtain the Y from the revised $\gamma.\lambda$.

For the signal(1D signal), we obtain 2 signals from DWT. For the γ, we can apply the DWT and obtain 3 signals from X.

For the image(2D signal), we apply the DWT to row and column of image respectively and obtain the 4 squares from each DWT step.

3. Wavelet Algorithms

1) EZW(Embedded Zerotree Wavelet)

EZW is based on two important observations :

(1) Natural images in general have a low pass spectrum.

(2) Large wavelet coefficients are more important than small coefficients.

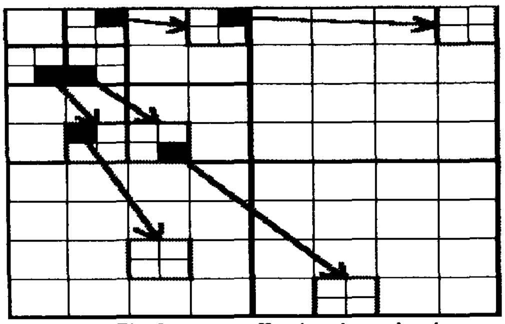

Fig. 2. parent-offpsring dependencies.

These two observations are exploited by encoding the wavelet coefficients in decreasing order, in several passes. For every pass a threshold is chosen against which all the wavelet coefficients are measured. If a wavelet coefficient is larger than the threshold, it is encoded and removed from the image, if it is smaller it is left for the next pass. When all the wavelet coefficients have been visited the threshold is lowered and the image is scanned again to add more detail to the already encoded image. This process is repeated until all the wavelet coefficients have been encoded completely or another criterion has been satisfied (maximum bit rate for instance).

A tree structure, called *spatial orientation tree*(or *quad tree)* , naturally defines the spatial relationship on the hierarchical pyramid. Figure 2 shows how spatial orientation tree is defined in a pyramid constructed with recursive four-subband splitting. Its direct descendants (offspring) corresponds to the pixels of the same spatial orientation in the next finer level of the pyramid. The tree is defined in such a way that each node has either no offspring(the leaves) or four offspring, which always form a group 2 x 2 adjacent pixels. A **zerotree** is a quadtree of which all nodes with root are equal to or smaller than the given threshold.

The first step in the EZW coding algorithm is to determine the initial threshold. If we adopt bitplane coding then our initial threshold t_0 will be $t_0 = 2^{\lfloor \log_2(\max(|c_{i,j}|)) \rfloor}$. With this threshold we enter the main coding loop:

```
threshold = initial_threshold;
do {
        dominant_pass(image);
        subordinate_pass(image);
        threshold = threshold/2;
   }  while (threshold > minimum_threshold);
```

We see that two passes are used to code the image. In the first pass, **the dominant pass**, the image is scanned(we used the scan order as Morton scan in Figure 3) and a symbol is outputted for every coefficient. If the coefficient is larger than the threshold a **P (positive)** is coded, if the coefficient is smaller than minus the threshold an **N (negative)** is coded. If the coefficient is the root of a zerotree then a **T (zerotree)** is coded

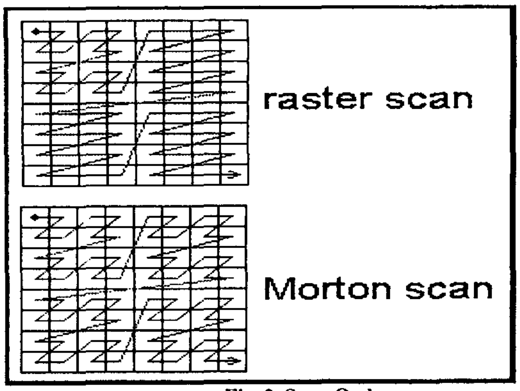

Fig. 3. Scan Order.

and finally, if the coefficient is smaller than the threshold but it is not the root of a zerotree, then a **Z (isolated zero)** is coded. This happens when there is a coefficient larger than the threshold in the subtree. The effect of using the N and P codes is that when a coefficient is found to be larger than the threshold (in absolute value or magnitude) its two most significant bits are outputted (if we forget about sign extension).

All the coefficients that are in absolute value larger than the current threshold are extracted and placed without their sign on the *subordinate list* and their positions in the image are filled with zeroes. This will prevent them from being coded again.

```
/* Dominant pass */

 initialize_fifo();
 while (fifo_not_empty)
 {
    get_coded_coefficient_from_fifo();
  if coefficient was coded as P, N or Z then
      {
      code_next_scan_coefficient();
      put_coded_coefficient_in_fifo();
    if coefficient was coded as P or N then
        {
        add abs(coefficient) to subordinate list;
        set coefficient position to zero;
        }
      }
 }
```

The second pass, the subordinate pass, is the refinement pass.

```
/* Subordinate pass */

 subordinate_threshold = current_threshold*3;
 for all elements on subordinate list do
     {if (coefficient=coeff. > subordinate_threshold)
        {output a one;
         coeff. = coefficient-subordinate_threshold*2;
        }
      else output a zero;
     }
```

	0	1	2	3	4	5	6	7
0	63	-34	49	10	7	13	-12	7
1	-31	23	14	-13	3	4	6	-1
2	15	14	3	-12	5	-7	3	9
3	-9	-7	-14	8	4	-2	3	2
4	-5	9	-1	47	4	6	-2	2
5	3	0	-3	2	3	-2	0	4
6	2	-3	6	-4	3	6	3	6
7	5	11	5	6	0	3	-4	4

Fig. 4. Set of image wavelet coefficients.

	0	1	2	3	4	5	6	7
0	Z	Z	Z	P	T	P	N	T
1	Z	Z	P	N	T	T	T	T
2	P	P	T	N	T	T	T	P
3	N	T	N	P	T	T	T	T
4	T	P	T	T			T	T
5	T	T	T	T			T	T
6	T	T	T	T	T	T	T	T
7	T	P	T	T	T	T	T	T

Fig. 5. Result of EZW.

This loop will always end as long as we make sure that the coefficients at the last level, i.e. the highest subbands (HH1, HL1 and LH1) are coded as zerotrees. The followings is the result of EZW using Figure 4.

Threshold : 32
Dominant Pass : PNZT PTTT TZTT TPTT
Subordinate Pass : Skip
Subordinate list {63,34,49, 47}

Threshold : 16
Dominant Pass : ZTNP TTTT TTTT TTTT TTTT
Subordinate Pass : 1010
Subordinate list {63, 34, 49, 47, 31, 23}

Threshold : 8
Dominant Pass : ZZZZ ZPPN PPNT TNNP TPTT
NTTT TTTT TPTT TPTT TTTT
TTTP TTTT TTTT TTTT TTTT
Subordinate Pass : 100010
Subordinate list {63, 34, 49, 47, 31, 23, 10, 14, 13, 15, 14, 9, 12, 14, 8}

2) SPIHT

Given n, if $|c_{i,j}| \geq 2^n$, then we say that a coefficient is *significant* ; otherwise it is called *insignificant*. The sorting algorithm(assume that the encoder and decoder have the same sorting algorithm) divides the set of pixels into partitioning subsets T and we use the function

$$S_n(T) = \begin{cases} 1, & \max_{(i,j)\in T}\{|c_{i,j}|\} \geq 2^n \\ 0, & otherwise \end{cases}$$

To indicate the significance of a set of coordinates T. To simply the notation of single pixel sets, we write $S_n(\{(i,j)\})$ as $S_n(i,j)$.

The following sets of coordinates are used to present the new coding method:

O(i,j) : set of coordinates of all offspring of node (i,j) ;
D(i,j) : set of coordinates of all descendants of the node (i,j);
H : set of coordinates of all spatial orientation tree roots (nodes in the highest pyramid level) ;
L(i,j) = D(i,j) - O(i,j)

Since the order in which the subsets are tested for significance is important, in a practical implementation the significance information is stored in three ordered lists, called list of insignificant sets(**LIS**), list of insignificant pixels (**LIP**), and list of significant pixels (**LSP**). Since LIS represents either the set D(i,j) or L(i,j), we say that a LIS entry is **of type A** if it represents D(i,j), and **of type B** if it represents L(i,j).

The following is the encoding algorithm for SPIHT.

1 **Initialization** :
output $n = \lfloor \log_2(\max_{(i,j)}\{|c_{i,j}|\}) \rfloor$; set the LSP as an empty list, and add the coordinates $(i,j) \in H$ to the LIP, and only those with descendants also to the LIS, as type A entries.

2 **Sorting pass :**
- 2.1 For each entry *(i,j)* in the LIP do :
  - 2.1.1 output $S_n(i,j)$;
  - 2.1.2 if $S_n(i,j)$=1 then move (i,j) to the LSP and output the sign of $c_{i,j}$;
- 2.2 for each entry *(i,j)* in the LIS do :
  - 2.2.1 if the entry is of type A then
    - output $S_n(D(i,j))$;
    - if $S_n(D(i,j))$=1 then
      - ➢ for each $(k,l) \in O(i,j)$ do
        - output $S_n(k,l)$
        - if $S_n(k,l)$ =1 then add (k,l) to the LSP and output the sign of $c_{k,l}$
        - if $S_n(k,l)$=0 then add (k,l) to the end of the LIP
      - ➢ if $L(i,j) \neq \varnothing$ then move (i,j) to the end of the LIS, as an entry of type B and go to Step 2.2.2; else, remove entry (i,j) from the LIS.
  - 2.2.2 if the entry is of type B then

- output $S_n(L(i,j))$;
- if $S_n(L(i,j))$=1 then
  - add each $(k,l) \in O(i,j)$ to the end of the LIS as an entry of type A ;
  - remove (i,j) from the LIS

3 **Refinement pass :** for each entry *(i,j)* in the LSP, except those included in the last sorting pass(i.e., with same n), output the n –th most significant bit of $|c_{i,j}|$;

4 **Quantization-step update :** decrement n by 1 and go to Step 2.

Using image wavelet coefficients as Figure 4., we show the result of SPIHT method for threshold 16.

LIS = {(1,1)A,(0,1)B,(2,0)A,(3,0)A,(3,1)A}
LIP = {(1,0),(1,1),(0,3),(1,2),(1,3),(2,0),(2,1),(3,0),(3,1),(4,2),(5,2),(5,3)}
LSP = {(0,0),(0,1),(0,2),(4,3)}

3) Other Algorithms

Many person suggest another wavelet algorithms which is very effective for many fields : Lifting scheme(Swelden), Complex wavelet transform(or DT CWT(Dual Tree Complex Wavelet Transform))(Kingsbury), Curvelet(Donoho).

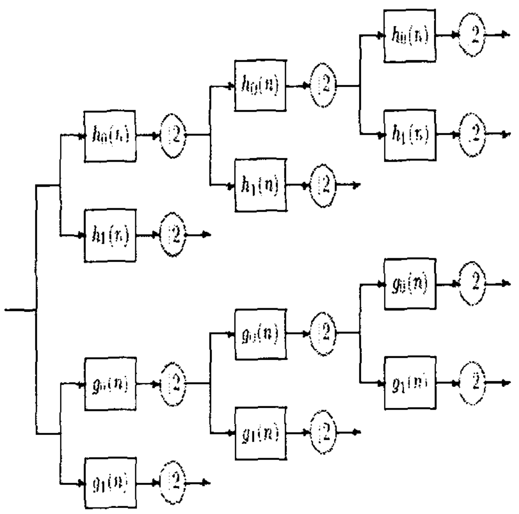

Fig. 6. DT CWT.

4. Applicaton to Remote Sensing

It is very important for the application to remote sensing(e.g, chage detection, mosaic, compression, image fusion, etc) to know the properties of wavelet algorithms. The following is the properties of wavelet algorithms.

EZW : progressive encoding, easy to implementation

SPIHT : progressive encoding, good image quality, high PSNR, especially for color images; fast coding/decoding (nearly symmetric)

Lifting scheme : no requirement of temporary arrays in the calculation steps

DT CWT : Approximate shift invariance, good directional selectivity in 2-dimensions

Curvelet : near-optimal representation of smooth objects having discontinuities along smooth curves

5. Conclusions

We briefly review the wavelet algorithms. We introduce the EZW and SPIHT since it is very widely used in image processing field. Finally, we present the properties of EZW, SPIHT, Lifting scheme, DT CWT, Curvelets.

References

[1] I. Daubechies, 1988. Orthonormal Bases of Compactly Supported Wavelets, *Comm. Pure Appl. Math.* 41 : 909-996

[2] A. Cohen, I. Daubechies, and J. Feauveau, 1992. Biorthogonal Bases of Compactly Supported Wavelets, Comm. Pure Appl. Math., 45 : 485-560

[3] J. M. Shapiro, 1993. , Embedded Image Coding using Zerotrees of Wavelet Coefficients. *IEEE Transactions on Signal Processing*, 41(12) : 3445-3462

[4] A. Said and W. A. Pearlman, 1996. A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees, *IEEE Transactions on Circuits and Systems for Video Technology*, 6 : 243-250.

[5] W. Sweldens, 1997. The lifting scheme: A construction of second generation wavelets, *SIAM J. Math. Anal*, 29(2) : 511-546

[6] N. G. Kingsbury, 1999. Image processing with complex wavelets, *Jphil. Transactions Royal Society London A*, 357: 2543-2560

[7] E. J. Candes and D. L. Donoho, 1999. Curvelets- A surprisingly effective nonadaptive representation for objects with edges, *In* Curve and Surface Fitting : Saint-Malo, A. Cohen, C. Rabut, and L.L.Schumaker, Eds, Nashville, TN : Vanderbilt University Press