AN IMPROVED PYRAMID ALGORITHM
FOR SYNTHESIZING 2-D DISCRETE
WAVELET TRANSFORMS

Chu Yu and Sao-Jie Chen
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, R.O.C.

Abstract – The Pyramid algorithm (PA) has been shown very suitable for computing 2-D forward and inverse discrete wavelet transforms (DWT). In this paper, we present a new 2-D synthesis PA to improve some defects encountered in the classical PA algorithm that usually requires large latency, long computation time, and big memory space. Unlike the PA algorithm which computes a 2-D IDWT level by level, our proposed algorithm performs a 2-D DWT in word size. Thus, for processing an $N{\times}N$ 2-D IDWT with m levels and L-tap filters, the proposed algorithm needs a latency of $3m+4$, computes only in N^2 clock cycles, and spends $2NL+4(m-1)$ memory space.

1. INTRODUCTION

In recent years, there have been a number of studies on wavelet transforms for signal analysis and synthesis [1-3]. Many algorithms for computing wavelet transforms have been proposed. For instance, Mallat presented 1-D and 2-D DWT/IDWT pyramid algorithms. Later, Vishwanath et al. [5, 6] developed the recursive pyramid algorithm (RPA) for 1-D DWT/IDWT, and proposed the modified recursive pyramid algorithm (MRPA) to improve their preceding algorithm. Soon after that, Chakrabarti et al. [6, 7] also described an extension of 1-D MRPA algorithm in 2-D case.

The 2-D inverse discrete wavelet transform (IDWT), which block diagram is illustrated in Fig. 1, can be implemented by using a pyramid algorithm (PA) [1, 4]. However, this 2-D PA is a separable extension of the 1-D PA in a level-by-level manner for the computations of 2-D IDWT's, thus it induces large latency, long computation time, and big memory space. Owing to these shortcomings, the PA is unsuited not only for hardware realization, but also for application in real-time signal processing. To overcome, we present a new improved pyramid algorithm for the 2-D IDWT, which has low latency, near-optimal computation time, and less memory space. As a result, the proposed algorithm is one of the better choices for scheduling the computations of 2-D IDWT's.

2. The proposed algorithm

The 2-D pyramid algorithm was developed by Mallat [1, 4]. The algorithm can be used to implement the 2-D IDWT as shown in the following pseudocode:

```
begin {2-D synthesis PA}
    input: x(1..N, 1..N);
    for (m = log(N) to 1)
        [Do the separable mth level of 2-D filterings after all the separable
        (m+1)th level of 2-D filterings have been done];
end {2-D synthesis PA}
```

In the above algorithm, we assume that the first level has the finest resolution and the $\log(N)$-th level the coarsest resolution. Clearly, this synthesis algorithm uses a level-by-level manner to accomplish the computations of 2-D IDWT's. The advantage of the PA algorithm is its ease of implementation. However, this algorithm has some defects. For instance, it requires a large number of memory space to store the intermediate results between row and column transforms, and the latency is too long because the first output data is produced only when the previous $\log(N)$-1 levels of 2-D IDWT's have been completely generated. Another defect of this algorithm is its long computation time. For an $N \times N$ input image processed through m levels of 2-D IDWT's, the computational complexity for iterating on the lowpass only filtering is given by:

$$2 \times \left[\left(\tfrac{1}{4} \right)^{m-1} N^2 + \cdots + \left(\tfrac{1}{4} \right)^2 N^2 + \left(\tfrac{1}{4} \right) N^2 + N^2 \right] = \tfrac{8}{3}(1 - 4^{-m})N^2 . \qquad (1)$$

According to the above equation, the upper bound on the number of lowpass (or highpass) operations is $8N^2/3$. Thus, based on this PA algorithm, it is unsuited for real-time signal processing due to its long latency and computation time. Moreover, it is unsuited for single-chip VLSI implementation, because it requires a large number of memory space.

In order to alleviate these defects, we present a novel algorithm, called the Recursive Quarter-Tree Pyramid Algorithm (RQTPA), to improve the performance of the classical PA algorithm. This RQTPA algorithm also performs the 2-D interpolation filterings based on a separable approach, but it does not synthesize 2-D IDWT level-by-level.

Unlike the classical PA, the proposed algorithm breaks down each subband of data into many subblock units, as shown in Fig. 2, to synthesize the corresponding level. The size of a subblock used to synthesize an nth level in total m levels is equal to 4^{m-n}. Since the algorithm computes 2-D IDWT's from the coarsest level to the finest level in each of the subblock units. Thus, the total storage requirement between levels is $\sum_{n=1}^{m-1} 4^{m-n}$. In order to reduce this large storage requirement, we again break down the larger subblock into only one input datum for each level. Based on this idea, the total storage requirement becomes $4(m-1)$. Finally, the proposed algorithm can be written as follows:

```
begin {2-D synthesis PA}
    input: x(1..N, 1..N);
    for (p = 1 to  N²/4^{log(N)} )
        do RQTPA (log(N));
end {2-D synthesis PA}

RQTPA(m)
begin {recursive quaternary-tree PA}
    if (m>0) then
        [Feed four subbands ( HH^m, HG^m, GH^m, GG^m ) with one
        input datum into a 2-D filter to synthesize four data of
        HH^{m-1}, as shown in Fig. 1];
        RQTPA (m-1);
        RQTPA (m-1);
        RQTPA (m-1);
        RQTPA (m-1);
end {recursive quaternary-tree PA}
```

From the above algorithm, we can expand it as a quaternary tree, as shown in Fig.
3. Root of the tree corresponds to the coarsest level of 2D IDWT's, each leaf node
represents a finest level, and the remaining levels are viewed as the internal nodes
of the tree. In order to visit this tree, the pre-order traversal is used because it
meets the feature of synthesis DWT. For example, if the coarsest level is three, the
traversal sequence for the proposed algorithm becomes as 3-2-1111-2-1111-2-
1111-2-1111, where each number stands for a resolution level. Note that each level,
except that the coarsest level contains four subbands and one input datum, has only
three subbands and one input datum, i.e., the low-high (HG), high-low (GH), and
high-high (GG) subbands. Another subband of input data, the low-low (HH),
comes from the filtering output of a previous coarser level.

For processing an $N \times N$ image, this algorithm needs N^2 clock cycles to
compute m levels of 2-D IDWT's. Since the first output appears only when all the
levels of filterings have been computed, the latency takes $3m+4$ clock cycles. The
memory space requires $2NL+4(m-1)$, where L is the filter length and $2NL$ is the
size of delay line used between row and column filters. Clearly, these performance
data reveal that the proposed algorithm can be applied in real-time signal
processing. In summary, the algorithm has the following features:

(1) Fast computation time.
(2) Low requirement of memory space.
(3) Low latency.
(4) Suitable for real-time signal processing due to features (1) and (3).
(5) Suitable for single-chip VLSI implementation due to feature (2).

3. PERFORMANCE EVALUATION

The performance comparison between our proposed algorithm and the classical pyramid algorithm is summarized in Table 1. This comparison is performed using the same row and column filter structures. The classical pyramid algorithm requires large latency, long computation time, and big memory space, which are all proportional to N^2 for computing an $N \times N$ image. On the other hand, our proposed algorithm requires a low latency of $3m+4$, around N^2 computation clock cycles, and $2NL+4(m-1)$ memory space for m levels of 2-D IDWT computations, each with L filter length.

4. CONCLUSION

A novel and efficient pyramid algorithm for the 2-D inverse DWT has been formulated in this paper. This proposed algorithm overcomes some defects of the classical PA, thus it is suited for scheduling a fast 2-D IDWT computation and providing a low-cost hardware implementation. Based on this algorithm, we will implement a real-time single-chip for DSP processing in the future.

5. ACKNOWLEDGMENTS

Reference

[1] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 11, no. 7, pp. 674-693, July 1989.

[2] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Magazine*, vol. 8, no.4, pp. 14-38, Oct. 1991.

[3] I. Daubechies, Ten Lectures on Wavelets, vol. 61 of CBMS-NSF Regional Conferences *Series in Applied Mathematics*, SIAM, Philadelphia, PA, 1992.

[4] S. Mallat, "Multifrequency channel decompositions of images and wavelet models," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 12, pp. 2091-2110, Dec. 1989.

[5] M. Vishwanath, "The recursive pyramid algorithm for the discrete wavelet transform," *IEEE Trans. Signal Processing*, vol.42, no. 3, pp. 673-676, Mar. 1994.

[6] C. Chakrabarti and M. Vishwanath, "Efficient realizations of the discrete and continuous wavelet transforms: from single chip implementations to mappings on SIMD array computers," *IEEE Trans. Signal Processing*, vol. 43, no. 3, pp. 759-771, Mar. 1995.

[7] R.M. Owens and M. Vishwanath, "A very efficient storage structure for DWT and IDWT filters," *Journal of VLSI Signal Processing*, vol. 19, no.3, pp.215-225, Aug. 1998.
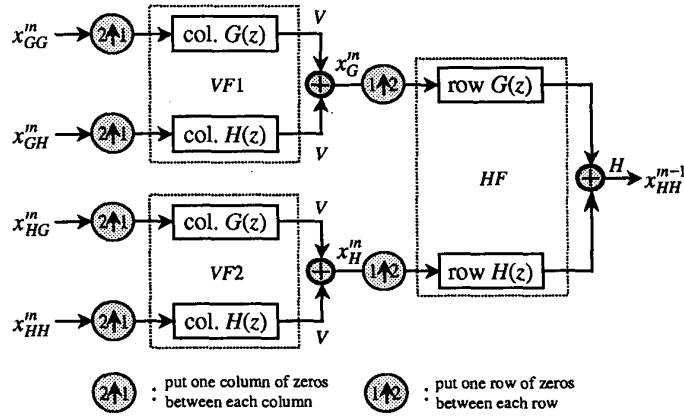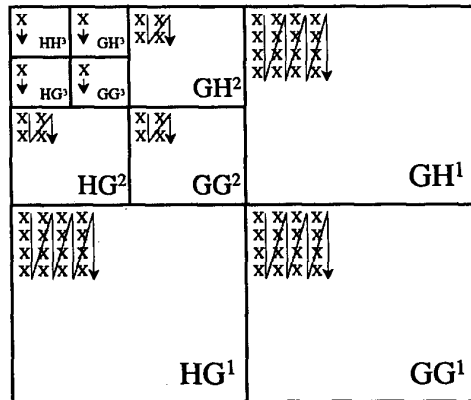
Fig. 1 One level of 2-D IDWT.

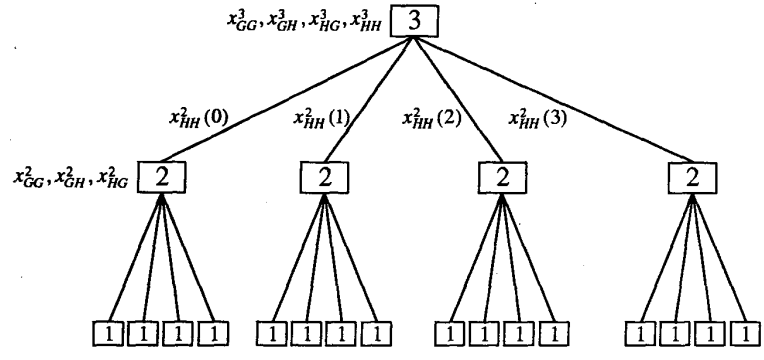Fig. 2 computation scheme with three levels.

Fig. 3 Three levels of quaternary tree.

Table 1: Performance Comparison of Two Algorithms.

Algorithms	Latency	Period	Memory Space	Control Unit
PA	N^2	$\frac{8}{3}N^2$	N^2	simple
RQTPA	$3m+4$	N^2	$2NL+4(m-1)$	complex