

# 리프팅 기반의 3차원 웨이블릿 변환 인코더의 아키텍처 설계

\*조덕은, 송낙은

홍익대학교 공과대학 전자공학과

e-mail : orion0807@hotmail.com, snukeh@hongik.ac.kr

## Architecture Design of 3D-Wavelet Transform encoder based on Lifting Scheme

\*Deok-Eun Cho, Nag-Un Song  
Dept. Electronics Engineering  
HongIk University

### Abstract

In this paper, the encoder architecture of 3-D wavelet transform based on lifting scheme is designed. architecture. Here, 3 level wavelet transform for spatial decomposition and 2 level wavelet transform for temporal decomposition is adopted with efficient computation.

### I. 서론

최근 멀티미디어의 빠른 발전으로 인해 정지영상과 동영상의 서비스가 급격히 늘어나면서 방대한 크기를 가지는 영상 정보를 원하는 대역폭 안으로 보내기 위해 많은 노력을 기하고 있다. 정지영상의 표준인 JPEG과 동영상의 표준인 MPEG1,2의 핵심 압축 방법으로 DCT를 사용하는데 블록효과가 생기는 등의 문제점을 가지고 있어 이를 개선하기 위해 블록으로 나누어 처리하지 않고 전체영상을 처리하는 웨이블릿 변환이 연구되었으며 JPEG2000[1]에서는 표준으로 채택되었다. 이를 확장하여 동영상에 사용되어 질 수 있는 3차원 웨이블릿 변환의 많은 연구가 진행되어져 왔다 [2,3,4]. 3차원 웨이블릿 변환을 이용한 동영상 부호화의 장점은 그것의 압축율과 PSNR, 그리고 시간적 공간적 해석이 가능하다는 것이다.

본 논문은 Daubechies 5/3 필터를 이용하여 3차원 웨이블릿 변환을 위한 아키텍처를 제안하여 이를 C++과 VHDL을 이용하여 알고리즘과 아키텍처를 검증하였다.

### II. 배경 이론

현재 새로운 웨이블릿 변환으로 기존의 컨볼루션 방

식을 대체하여 계산량을 줄일 수 있는 방식인 리프팅 기법이 많이 연구되어 지고 있다. 리프팅 기법은 위스웰튼스에 의해 발견되었다[5]. 리프팅 기법의 기초이론은 상수값을 갖는 diagonal 행렬과 엇갈린 upper triangular 행렬과 lower triangular 행렬로 웨이블릿 필터의 polyphase 행렬을 인수분해하는 것이며 이는 Euclidean 알고리즘에 의하여 이루어지며, 그 결과는 행렬의 곱으로 이루어진다.

$\tilde{h}(z)$ 와  $\tilde{g}(z)$ 가 각각 저대역과 고대역 통과 분해 필터라고 하고  $h(z)$ 와  $g(z)$ 를 각각 저대역과 고대역 통과 합성 필터라고 하자. 필터는 짝수성분과 홀수성분으로 나누어질 수 있는데 이것의 polyphase 행렬은 다음과 같이 정의 되어진다.

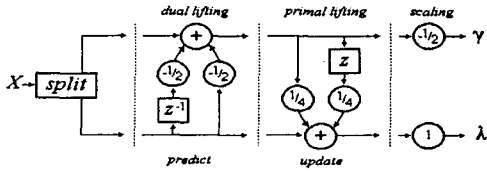
$$\tilde{P}(z) = \begin{bmatrix} \tilde{h}_e(z) & \tilde{h}_o(z) \\ \tilde{g}_e(z) & \tilde{g}_o(z) \end{bmatrix}, P(z) = \begin{bmatrix} h_e(z) & h_o(z) \\ g_e(z) & g_o(z) \end{bmatrix}$$

여기서  $h$ 와  $g$ 가 각각 보수의 관계에 있는 필터라면  $P(z)$ 는 아래와 같이 항상 리프팅 스텝으로 인수분해 되어 질수 있다.

$$P_1(z) = \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix} \prod_{i=1}^m \begin{bmatrix} 1 & s_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_i(z) & 1 \end{bmatrix}$$

여기서  $K_1$ 과  $K_2$ 는 상수이다.

이를 Daubechies 5/3 필터에 적용한 것이 <그림 1>이다.

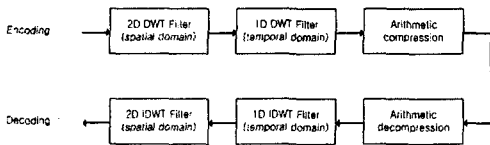


<그림 1> 리프팅을 이용한 Daubechies 5/3 필터



<그림 3> 시간축상의 분해 : (a) 16 GOF, (b) 4 GOF

<그림 2>는 3차원 웨이블릿을 이용한 동영상의 압축 인코더와 디코더를 알기쉽게 나타낸 그림이다. 어떤 동영상의 시퀀스가 입력으로 들어오면 각 프레임들 공간상의 2차원 웨이블릿 변환 한다. 2차원 웨이블릿 변환은 모든 프레임에 적용되고 이를 각 프레임의 같은 위치의 픽셀 값들을 1차원 웨이블릿 변환하여 3차원 웨이블릿 변환이 완성된다. 3차원 웨이블릿 변환된 후의 데이터는 arithmetic coding등의 부호화 기술을 이용하여 압축이 이루어질 수 있다.



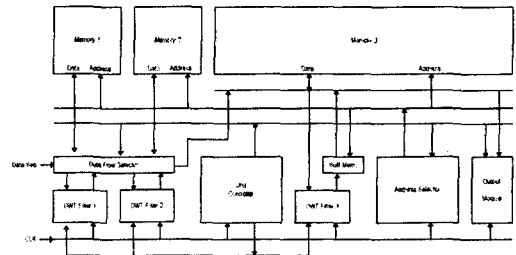
<그림 2> 전통적인 3차원 웨이블릿 분해 기법

전통적인 3차원 웨이블릿 변환은 모든 프레임에의 접근이 요구되어 짐에 따라 실제로 사용되어지는 것이 불가능하다. 이를위해 MPEG등에서 쓰여지는 GOP (Group of Pictures)와 유사한 개념의 GOF(Group of Frame)의 개념을 도입하여 사용한다. GOF는 일정한 길이의 시퀀스를 그룹으로 지정하는 것이다. 각 프레임은 2차원 웨이블릿 변환을 이용하여 분해한 후 시간상으로 고주파 대역과 저주파 대역으로 분해되어진다.

<그림 3>은 각각 16과 4의 GOF 길이를 가진 영상의 시간상의 분해 결과를 보여준다. 3차원 웨이블릿 변환 알고리즘에서 적은 길이의 GOF를 쓰는 경우에는 시간축상 고주파 대역의 프레임이 적어 압축에 있어서 비 효율적이라는 단점을 가진다. 하지만 많은 수의 시간축상 고주파 대역의 프레임을 가진 길이가 16인 GOF에서는 시간상의 분해를 위해 모든 프레임을 저장할 많은 양의 메모리를 필요로 하고 반복되는 웨이블릿 변환으로 인한 Delay가 발생한다. 이로인해 3차원 웨이블릿 변환을 위해서는 메모리와 압축효율간의 적당한 타협을 필요로한다.

### III. 제안된 아키텍처의 설계

본 논문에서 제안된 아키텍처는 <그림 4>에서 보는 바와 같이 각 프레임의 2차원 웨이블릿 변환을 위한 메모리 모듈 두개, 시간상의 웨이블릿 변환을 위해 8개의 프레임을 저장하는 메모리 한개, 웨이블릿 변환시에 필요한 데이터의 흐름을 제어하는 데이터 흐름 선택기, 클럭을 생성하기 위한 클럭 생성기와 시간상의 웨이블릿 변환을 위해 하나의 버퍼 메모리를 추가하였고, 변환속도의 향상을 위해 세 개의 웨이블릿 필터, 전체 모듈의 제어를 위한 제어유닛으로 구성된다.

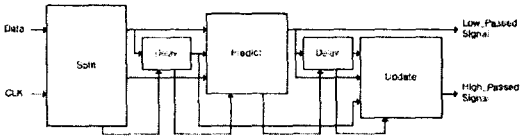


<그림 4> 제안된 3차원 웨이블릿 변환 아키텍처

3차원 웨이블릿 변환 과정을 알아보면 먼저 입력된 프레임이 메모리1과 메모리2, DWT 필터를 통해 3레벨의 웨이블릿 변환하게 된다. 이후 2차원 3레벨 웨이블릿 변환된 각 프레임의 값들은 메모리 3에 저장되고 메모리 3에 8개의 프레임이 저장 되면 이 8개의 프레임을 시간상의 웨이블릿 변환을 위해 각 프레임의 같은 위치를 가진 픽셀 8개를 1차원 웨이블릿 변환한다. 이를 버퍼메모리에 보내어 웨이블릿 변환이 완료된 후 다시 메모리 3에 저장한다. 이 과정이 각 프레임의 같은 위치를 차지하는 모든 픽셀 값들에 적용되어지면 3차원 웨이블릿 변환이 완료된 것이다.

본 논문에서 제안된 아키텍처의 주요 부분인 웨이블릿 변환 필터는 Daubechies 5/3 필터를 이용하여 <그림 5>과 같이 구성되었다. 메모리에서 들어온 데이터의 시퀀스는 짝수 성분과 홀수 성분으로 스플릿에서 나누어진후, 업데이트와 프리딕트의 과정을 거쳐 변환

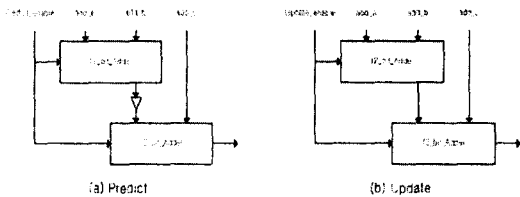
되어진다.



<그림 5. Daubechis 5/3 필터의 구조>

프리딕트와 업데이트는 두개의 12 비트 덧셈기를 통해 <그림 6>과 같이 구현되었다. 덧셈이 이루어질 때 1/2을 곱해주는 것은 단순히 1 비트씩 쉬프트 해 줌으로써 이루어진다. 여기서 구현된 합은 두 번째 12 비트 덧셈기에 전송되기 전에 2's complement를 구하기 위해 인버터를 경유하게 된다. 이는 12 비트 덧셈기를 12비트 뺄셈기로 바꾸어주는 역할을 한다. 이 때 predict\_enable은 두 덧셈기의 수행여부를 부여한다.

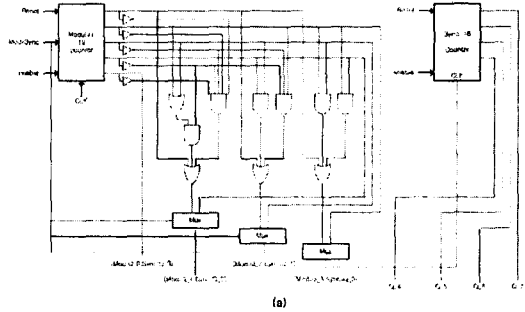
업데이트 역시 프리딕트와 마찬가지로 두 개의 덧셈기로 이루어진다. <그림6>의 (b)가 업데이트의 구조이다. 각 값이 더해지기 전에 1/4의 값을 곱해주어야 하는데 이 역시 프리딕트와 마찬가지로 각 2비트씩 쉬프트 해줌으로써 이루어진다.



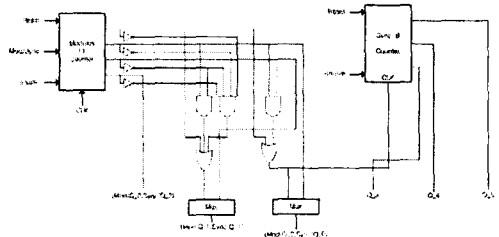
<그림 6. Predict와 Update의 구조>

주소 선택기는 데이터의 웨이블릿 변환시 웨이블릿 변환 필터의 입력신호를 선택하기 위해 사용되고, 또한 웨이블릿 변환 후 이를 다시 메모리에 각 픽셀 값을 알맞은 주소에 저장하기 위해 사용되어진다. 본 논문에서는 변환 속도의 향상을 위해 메모리마다 각기 다른 3개의 주소선택기를 부여하였다.

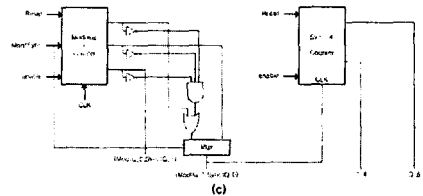
일단 주소 선택기에 들어갈 카운터에 대하여 알아보자. 웨이블릿 변환을 할때는 항상 경계치를 처리하여야 하는데 이를 미러링을 통해 이루어지도록 설계하였다. 이로인해 N개의 데이터 시퀀스를 처리하기 위해서는 N+3개의 클럭 주기가 필요하다. 그리고 웨이블릿 변환 된 데이터를 다시 메모리에 저장하기 위해서는 N개의 클럭 주기가 필요하므로 각 메모리에 배치되는 주소선택기에는 두가지 주기를 모두 만족 할 수 있는 카운터의 설계가 필요하다.



(a)



(b)



(c)

<그림 7. 각 레벨별 주소 선택기>

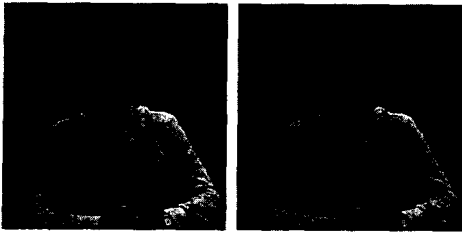
3레벨의 웨이블릿 변환을 위해서 N, N/2, N/4개의 데이터 시퀀스를 처리하는 3개의 주소선택기를 설계하였다. 본 논문의 아키텍처 시뮬레이션에 사용되어질 영상은 16\*16의 정지영상의 시퀀스로 최대 16개의 시퀀스를 웨이블릿 변환하게 된다.

#### IV. 실험 결과 및 검토

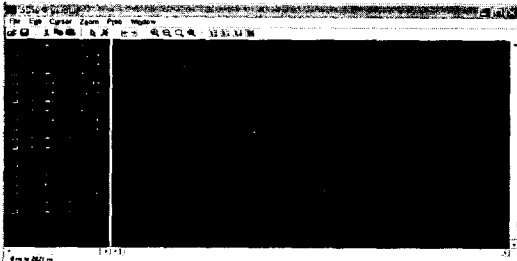
알고리즘의 검증과 시뮬레이션을 위하여 3차원 웨이블릿 변환과 이의 복원을 C++을 이용하여 구현하였고 검증하였다. C++을 이용한 구현에서는 연속된 raw포맷 형식의 256\*256영상을 사용하여 인코딩 및 디코딩을 통하여 알고리즘을 검증하였다.

하드웨어 검증을 위해 16\*16의 작은 영상을 VHDL을 이용하여 검증하였다.

참고문헌



(a) 원영상 (b) 복원된 영상  
<그림 8. 원영상 및 복원된 영상 결과>



<그림 9. Function Simulation 결과>

Daubechies 5/3 필터의 또 다른 장점은 복원된 영상의 각 픽셀값이 소수점을 포함하지 않은 정수 값을 그대로 가질수 있다는 장점이 있다. 다른 필터의 경우에는 변환시의 곱셈에 의해 복원시 약간의 변화가 있을 수 있으나 이 필터는 정수 값을 그대로 복원한다.

실험결과 DCT에 대비하여 블록효과를 제거함으로써 보다 나은 영상 품질을 제공하고[6], 컨벌루션을 이용한 웨이블릿 변환과 비교하여 곱셈연산을 줄임으로써 전체적인 연산량을 줄일 수 있었다.

IV. 결론 및 향후 연구 방향

본 논문에서는 동영상 압축시 DCT의 단점을 보완하여 대체할 수 있는 3차원 웨이블릿 변환 아키텍처를 제안하였고, 이를 C++와 VHDL을 이용하여 설계한 아키텍처를 검증하였다. 이를 이용하여 동영상 압축 알고리즘에 적용한다면 현재 DCT를 채택한 MPEG이나 H.26x 보다 나은 영상품질을 제공할 수 있다.

향후 3차원 웨이블릿변환에 SPIHT알고리즘, motion compensation 알고리즘인[3,6]등을 적용하며, 3차원 웨이블릿 변환에서의 메모리 사용 효율성을 높이는 아키텍처를 연구하고자 한다.

[1] ISO/IEC JTC 1/SC 29/WG 1(ITU-T SG8). Coding of still picture: JPEG 2000.

[2] Christine I. Podilchuk, and N. S. Jayant, N. Farvardin "Three-Dimensional Subband coding of Video", IEEE Transactions on image processing, Vol 4, No 2, pp. 125-139, Feb. 1995

[3] Beong-Jo Kim and W. A. Pearlman, "An Embedded Wavelet Video Coder Using Three-Dimensional Set Partitioning in Hierarchical Trees", Data Compression Conference, 1997. DCC '97. Proceedings, 25-27, pp.251-260, 1997

[4] Jizheng Xu, Zixiang Xiong, S. Li, and Y. Q. Zhang, "Memory-constrained 3-D Wavelet Transform for Video Coding Without Boundary Effects", IEEE Transactions on circuits and systems for video technology, Vol.12, No 9, pp. 812-818 Sep. 2002

[5] I. Daubechies, W. Sweldens 'Factring Wavelet Transforms into Lifting Scheme,' J. Fourier Anal. Appl., Vol 4, pp. 247-269, 1998.

[6] A. Secker and D. Taubman, "Motion-compensated highly scalable video compression using an adaptive 3D wavelet transform based on lifting", IEEE Image Processing, 2001. Proceedings. 2001 International Conference on, Volume: 2, 7-10, pp. 1029-103., Oct. 2001