

EZW의 하드웨어 설계

이두영, 송문빈, 임재청, 심정섭, 정연모
 경희대학교 전자공학
chung@khu.ac.kr

Hardware Design of EZW

Dooyoung Yi, Moonvin Song, Jaechung Lim, Jungsub Sim, Yunmo Chung
 Dept. of Electronic Engineering, Kyung-Hee University

요약

본 논문은 웨이블릿 변환 과정을 통해서 분해한 영상을 Shapiro가 제안한 효율적인 영상 압축 방법인 EZW(Embedded Zerotree Wavelet) 알고리즘을 하드웨어로 설계 하였다. 이를 위한 하드웨어 구조를 제시하고 VHDL로 모델링 하여 FPGA를 통해 검증 하였다.

1. 서론

JPEG은 DCT 변환을 블록단위로 처리하는 정지영상 압축 알고리즘 중 가장 많이 사용되고 있다. 이 기술은 블록화 현상이 일어나는 단점을 가지고 있다. 그러나 웨이블릿 변환은 영상전체를 동시에 분해하기 때문에 블록화 현상이 발생하지 않는다. 또한 JPEG에서 사용하는 푸리에 변환(Fourier transform)이나 가버 변환(Gabor transform)에서 시간과 주파수 영역을 동시에 표현할 수 없었던 문제를 해결함으로써 여러 분야에서 사용되고 있다. EZW는 웨이블릿 변환을 사용하여 부호화하는 방법으로 제로트리(zerotree)의 개념과 대역간의 유사성을 이용하여 부호화의 효율을 높였다. 본 논문에서는 정지영상을 빠르고 효율적으로 부호화하기 위하여 EZW의 하드웨어 구현을 연구 하였다.

2. EZW 개념

2차원 영상을 웨이블릿 변환할 경우에 [그림 1]의 대역 분할 부분에서 나타난 것과 같이 필터뱅크 구조를 구성한다. 1단계 웨이블릿 분해를 나타낸 그림으로 $g_0(n)$ 은 고주파 필터이고, $h_0(n)$ 은 저주파 필터이다. [그림 1]과 같이 입력영상의 고주파 및 저주파 필터를 각각 통과시킨 두개의 영상을 먼저 만들고, 이들에게 다시 고주파 및 저주파 필터를 각각 통과하면 4개의

결과 영상 HH, HL, LH, LL로 부표분화 된다[1]. 첫 번째 필터링은 수평방향이고, 두번째 필터링은 수직방향이다.

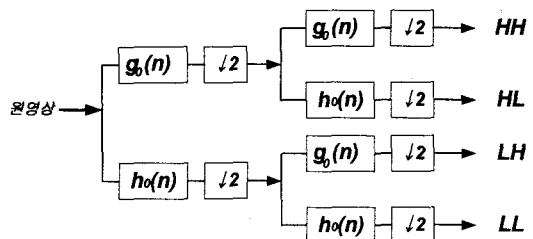


그림 1. 1단계 웨이블릿 분해

[그림 1]에서 분해된 LL 영상에 대해서 제 1 단계와 같이 수평 및 수직방향으로 저주파 및 고주파대역을 통과하면 부표분화되어 [그림 2]와 같이 분해 대역 LL2, LH2, HL2, HH2로 나눈다.

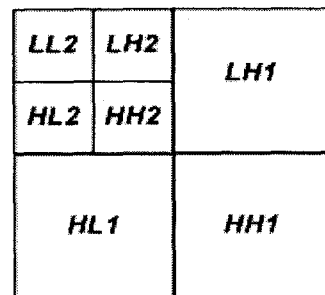


그림 2. 2단계 웨이블릿 분해

이렇게 저주파 대역만을 계속적으로 분해하는 것을 옥타브 분해법(octave decomposition)이라고 한다. 복원은 분해의 역 과정으로서 분해된 대역을 업샘플링(upsampling)한 후 계수를 필터링하여 분해된 대역들로부터 원영상을 복원한다.

[그림 3]은 부호화 및 복호화의 전 과정을 나타낸 그림이다.

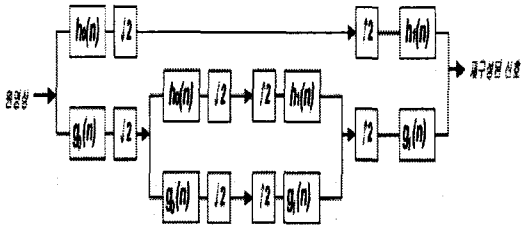


그림 3. 웨이블릿 분해와 복원

이산 웨이블릿 변환(discrete wavelet transform)으로 분해된 웨이블릿 계수를 이용하는 Shapiro가 제안한 EZW는 원하는 임의의 비트율에서 부호화를 멈추거나 복호화 할 수 있다는 장점을 가지고 있다. EZW는 제로트리 부호화를 이용하며 처리 과정은 압축율을 높이기 위해 주부호화과정(dominant pass)과 종속 부호화과정(subordinate pass)으로 나뉜다. 이때 기준값을 정의한 후 이것보다 크면 POS(Positive), 이것보다 작으면 NEG(Negative), 작지만 중요계수가 있으면 IZ(Isolated Zero), 또한 작고 중요계수도 없으면 ZTR(Zero Tree Root)의 4가지 경우를 고려하여 부호화한다.

또한 웨이블릿의 기본적인 특성인 자기 유사성(self-similarity)과 연속적인 근사 양자화(successive approximation quantization) 방법을 사용하여 부호화한다[2]. [그림 4]는 웨이블릿 계수 사이의 관계를 나타낸 그림이다. 이 그림에서 저주파대역을 나타내는 LL3는 한 단계 낮은 고주파대역 안에서 4개의 노드(node)로 이루어진 quard-tree를 만들며, 고주파대역인 LL2는 다음 고주파대역 안에서 4개의 노드로 이루어져 있음을 알 수 있다. 3옥타브 방식의 LL3이 부모계수일 경우, 상관성을 가지는 하위자손계수의 개수는 총20개이다. 이는 하부로 갈수록 계수 값이 작아지는 특징이 있다. 이것은 상위의 부모계수는 하위의 자손계수와 연관성을 가지고 있음을 알 수 있고, 부모계수가 중요계수이면 자손계수 또한 중요계수일 확률이 높고, 부모계수가 비중요계수일 경우 자손계수도 비중

요계수일 확률이 높다. 그러므로 다른 레벨에 있는 방향성이 같은 대역들 간 중요도를 예측할 수 있는데, 이것을 자기유사성이라 부른다[3].

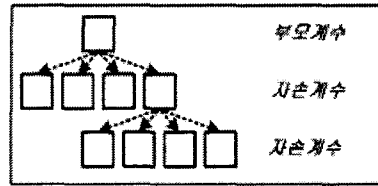
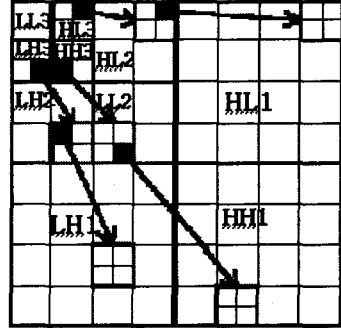


그림 4. 웨이블릿 계수사이의 관계

웨이블릿 변환에서 거의 모든 정보는 저주파대역 안에 존재하며 고주파대역으로 갈수록 '0'에 가까운 값을 가지며, 웨이블릿 계수가 클수록 더 많은 정보를 가진다. 따라서 저주파 대역에서부터 시작하여 고주파 대역으로 스캐닝하면서 하부의 제로트리 루트는 하나의 심볼로 지정하여 모든 비트플랜(bit plan)을 부호화 하는데 효과적이다. 제로 트리 스캐닝 방법은 [그림 5]와 같은 순서에 의해서 이루어진다[4]. 스캐닝 할 때 비교를 수행할 입력 기준값은 다음 식에 의해서 정의할 수 있다.

$$t_0 = 2^{\lfloor \log_2(\text{MAX}(|x,y|)) \rfloor} \quad (1)$$

식(1)에서 MAX(·)은 최대계수 값을 의미하며, $y(x,y)$ 는 계수로 주어져서 최대 값보다 작은 2^n 을 만족하는 최대 값이다.

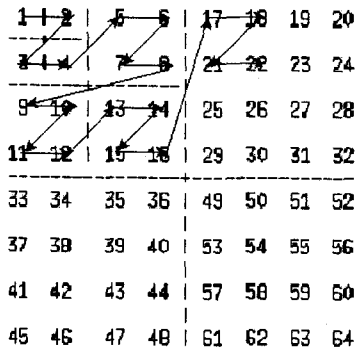


그림 5. 제로 트리 스캐닝 방법

[그림 6]에서 웨이블릿 변환된 계수를 기준값을 32로 양자화 하게 되면 [표 1]과 같은 주부호화 결과를 얻는다.

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

그림 6. 웨이블릿 변환된 계수

표 1. 주부호화

부밴드	계수 값	심볼	재구성 값
LL3	63	POS	48
HL3	-34	NEG	-48
LH3	-31	IZ	0
HH3	23	ZTR	0
HL2	49	POS	48
HL2	10	ZTR	0
HL2	14	ZTR	0
HL2	-13	ZTR	0
LH2	15	ZTR	0
LH2	14	IZ	0
LH2	-9	ZTR	0
LH2	-7	ZTR	0

중요도 판별에 의해 주부호화 과정을 거쳐 중속 부호화를 시작한다. 주부호화에서 중요계수로 판단된 계수들에 대한 절대 값을 저장하여 계수의 절대 값이 높은 순서로 부호화가 진행하면서 원영상과 가까운

값을 가지는 근사 양자화(successive approximation quantization)를 하게 된다. [표 1]에서 중요계수로 판단된 계수를 가지고 [표 2]와 같이 중속부호화를 하게 된다. 처음에 기준값 32로 양자화한 것에 symbol '1'을 보내면 63은 56으로 근사화 되고 34는 40으로 근사화 된다[2].

표 2. 중속 부호화의 예

계수값	심볼	재구성값
63	1	56
34	0	40
49	1	56
47	0	40

[표 2]과 같이 중속부호화 과정은 주부호화 과정에서 중요계수로 판단된 계수들의 정밀도를 높여나가는 작업을 한다. 한번 주부호화와 중속부호화를 거치면, 기준값을 반으로 낮추어서 주부호화와 중속부호화를 반복하게 된다.

3. 설계

[그림 7]은 EZW의 하드웨어를 구현하기 위해서 본 논문에서 제시한 구조이다. 이 구조는 웨이블릿 변환된 계수를 저장하는 memory 부분과 Dominant pass 부분, Subordinate 부분, 중요도 맵을 저장하는 Map register와 Control 블록으로 크게 나눌 수 있다. Dominant pass 블록에서는 변환된 웨이블릿 정수계수를 가지고 네 가지의 심볼을 생성하고, 중속 부호화에서는 주부호화한 계수들 중에서 중요한 계수라고 판명된 계수들을 한 번 더 부호화해서 원계수에 근사화 한다.

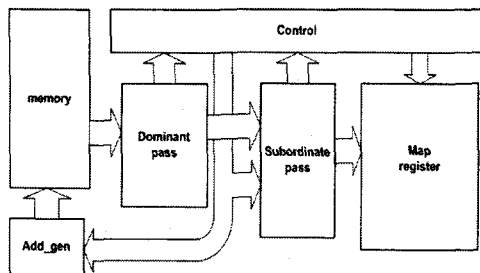


그림.7 EZW의 하드웨어 구조

[그림 8]은 보다 구체적인 EZW의 하드웨어를 제시한

블록 다이어그램이다.

주부호화 블록 중에서 Conversion unit 블록은 음수로 들어온 웨이블릿 계수를 양수로 바꾼다. 양수로 변환된 계수는 Data division을 통해 중요한 계수와 하부 계수로 나누어져 비교를 한 후에 하부에 중요한 계수가 있는지를 찾는다. 이때 만약 하부에 중요한 계수가 존재하면 iz_flag 블록에서 '1'을 발생시켜 하부 계수를 계속 스캔할 수 있도록 설정한다.

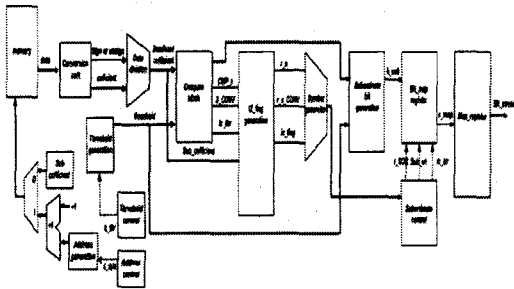


그림 8. EZW의 전체 블록 다이어그램

Symbol generation에서는 POS, NEG, IZ, ZTR로 부호화를 한다. Subordinate bit generation에서는 주부호화 과정에서 중요한 계수로 정의된 계수를 근사화 과정을 통해 Map_register에 저장한다. Control 블록은 부호화 과정을 수행할 계수와 부호화하지 않아야 할 계수를 판별하여 필요에 따라서 제어신호를 발생한다. 또한 중속부호화 신호를 필요에 따라서 발생시키며 기준값이 변화하는 정도에 따라서 스캔의 반복 횟수를 정한다. Address generation에서는 순차적인 스캔 방법을 사용할 수 없으므로 스캔시의 반복적인 패턴에 따른 제로 스캔의 하드웨어 구조를 설계하였다. 어드레스 생성방법은 [표 3]에서 나타내었다. 본 논문에서는 게이트를 최소화하기 위해 하나의 카운터만을 사용하여 구현하였다.

4. 결론

본 논문에서는 효율적인 압축 알고리즘인 EZW의 변환 속도를 높이고 성능을 향상시키기 위한 하드웨어 구조를 제시하였으며 FPGA 상에서 작동을 검증하였다.

5. 참고 문헌

- [1] J. W. Woods and S. O'Neil, "Subband coding of images," *IEEE Transactions on Acoustics, Speech, and Signal Processing*. Vol. 34, pp. 1278-1288, Oct 1986.
- [2] J. M Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, Vol. 41, pp.3445-3463, Dec, 1993.
- [3] K. Wiatr, P. Russek "Embedded zero wavelet coefficient coding method for FPGA implementation of video codec in real-time systems" *International Conference on*, 2000
- [4] V. N Ramaswamy, K. R Namuduri and N. Ranganathan "performance analysis of wavelet in embedded zerotree-based lossless image coding schemes" *IEEE Transactions, Image Processing* Vol. 2 pp. 278-281. Oct 1997
- [5] R. Y. Omaki et al., "Architecture of embedded zerotree wavelet based real-time video coder," *Proceedings of the 12th Annual IEEE International ASIC/SOC Conference*, pp. 137-141, Sep.1999.

표 3. 어드레스 발생순서

순서 경우	0	1	2	3
2×2	0	1	64	65
4×4	0	2	128	130
8×8	0	4	256	260
16×16	0	8	512	520
32×32	0	16	1024	1040
64×64	0	32	2048	2080
.