

# 메모리 사용을 최소화하는 웨이블릿 영상 부호화기에 관한 연구

정회원 박 성 옥\*, 박 종 옥\*

## A Study of Wavelet Image Coder for Minimizing Memory Usage

Sung-Wook Park\*, Jong-Wook Park\* Regular Members

요 약

본 논문에서는 최소의 메모리 사용량으로 원하는 비트율로 영상 부호화가 가능한 웨이블릿 기반의 영상 부호화기를 제안하였다. 제안된 방법은 부호화 과정시 요구되는 메모리 사용량을 줄이기 위해 웨이블릿 계수들의 비트 레벨 정보를 가지는 2D 중요 계수 배열을 사용하였다. 2D SCA는 웨이블릿 계수의 비트 레벨 정보를 저장하는 이차원 자료 구조로서, 제안된 알고리즘은 이것을 이용하여 중요한 계수에 대한 부호화 과정과 계수들의 비트 레벨 정보의 부호화 과정을 한 번에 수행할 수 있다. 실험 결과 기존의 부호화 방법보다 화질 면에서 비슷하거나 우수한 성능을 보였다. 특히 2D SCA를 이용한 최소의 메모리 사용으로 다양한 비트율에서 영상의 일그러짐 없이 안정적으로 동작함을 확인하였다.

ABSTRACT

In this paper, the wavelet image coder, that can encode the image to various bit rate with minimum memory usage, is proposed. The proposed coder is used the 2D significant coefficient array(SCA) that has bit level information of the wavelet coefficients to reduce the memory requirement in coding process. The 2D SCA is two dimensional data structure that has bit level information of the wavelet coefficients. The proposed algorithm performs the coding of the significance coefficients and coding of bit level information of wavelet coefficients at a time by using the 2D SCA. Experimental results show a better or similar performance of the proposed method when compared with conventional embedded wavelet coding algorithm. Especially, the proposed algorithm performs stably without image distortion at various bit rates with minimum memory usage by using the 2D SCA.

### I. 서론

영상 부호화 기법 중에서 웨이블릿 변환을 이용한 영상 부호화 방법은 JPEG과 같은 블록 기반 부호화기에서 발생하는 블록킹 현상(blocking effect)을 없애주고 고압축이 용이하다는 특징으로 인해서 웨이블릿 기반 부호화 기법에 대한 많은 연구가 이루어지고 있다<sup>[1~4]</sup>. Shapiro는 웨이블릿 분해된 영상이

가지는 같은 방향의 부대역간의 자기 유사성(self-similarity)을 이용한 연속적인 근사 양자화(successive approximation quantization:SAQ)를 사용한 EZW(embedded zerotree wavelet)알고리즘을 제안하였다<sup>[5]</sup>. EZW 부호화 방법은 원하는 비트율로 영상을 정확히 전송할 수 있는 비트율 제어(bit-rate control)가 가능하기 때문에 통신 채널의 대역폭에 따라서 비트율을 조절할 수 있을 뿐만 아니라, SAQ를 사용하여

\* 인천대학교 전자공학과

논문번호 : 020405-0916, 접수일자 : 2002년 9월 16일

크기가 큰 웨이블릿 계수들을 먼저 전송하기 때문에 주어진 비트율에서 보다 우수한 화질을 제공하고, 알고리즘이 빠르고 단순하다는 장점이 있다. Said와 Pearlman은 Shapiro의 방법을 개선하여 SPIHT (Set Partitioning in Hierarchical Trees) 알고리즘을 제안하였다<sup>[6]</sup>. SPIHT는 EZW와 마찬가지로 자기 유사성 탐색이라는 개념을 근거로 한다. 그러나 Shapiro의 연구에서는 각 부밴드의 계수들 중에서 크기가 가장 큰 계수의 절대값들의 집합을 이용하여 부밴드들이 중요한 대역인지 중요하지 않은 대역인지를 판단한 후, 중요한 부밴드만 부호화 하기 때문에 부호화의 효율성이 높다.

본 논문에서는 웨이블릿 계수의 비트 레벨 정보를 가지는 2D SCA를 사용한 웨이블릿 영상 부호화기를 제안한다. 제안한 방법은 빠르고 간단한 EZW 알고리즘과 중요 부밴드만 선택적으로 부호화하는 SPIHT의 개념을 사용함으로써, 웨이블릿 계수들의 반복적인 스캐닝에 의한 부호화 과정의 중복성을 효율적으로 개선하여 제로트리 부호화를 효과적으로 수행할 수 있는 부호화 방법이다. 이를 위해 웨이블릿 계수들의 비트 레벨 정보를 저장하는 2D SCA를 사용하여, 계수의 중요도 검사와 계수의 스캐닝을 동시에 수행할 수 있어서 영상 부호화를 위해 사용되는 메모리 사용량과 처리 속도를 개선할 수 있다. 또한 제안한 방법은 기존의 EZW와 마찬가지로 임베디드 비트 스트림(embedded bitstream)을 생성하기 때문에 정확한 비트율 제어가 가능하며, 중요한 정보를 가지는 웨이블릿 계수들을 먼저 전송하기 때문에 점진적 영상 전송이 가능하다는 장점이 있다.

본 논문은 다음과 같이 구성된다. II장에 웨이블릿 변환 이론에 대해 간략히 소개하고, III장에는 웨이블릿 기반의 부호화에 대하여 설명한다. IV장에서 제안된 부호화 방법에 대하여 기술하고, V장에서는 제안한 방법에 대한 실험 결과를 제시한다. 그리고 마지막 VI장에서 결론을 맺는다.

## II. Wavelet Transform

웨이블릿 변환의 기저함수는  $\psi(t)$ 로 정의되는 mother wavelet을 시간축에서  $b$ 만큼 변이시키고,  $a$ 만큼 확대/축소하여 얻어낸다. 1-D 신호  $f(t)$ 의 Continuous Wavelet Transform(CWT)는 다음과 같다.

$$W_f(a, b) = \int_{-\infty}^{+\infty} \varphi_{a,b}(t) f(t) dt \quad (1)$$

여기서, 웨이블릿  $\varphi_{a,b}$ 는 모 웨이블릿(mother wavelet)  $\psi$ 의 변이와 확대/축소에 의해서 얻어진다.

$$\varphi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \text{ where } a \in R^+, b \in R, a \neq 0 \quad (2)$$

$R^+$ 는 양의 실수이며,  $R$ 은 실수이다. Discrete Wavelet Transform(DWT)의 경우, 식(2)의  $a, b$ 는 이산화 정수값을 가지게 된다.

$$a = a_0^m, b = nb_0 a_0^{-m} \text{ with } (m, n) \in Z, \text{ and } a_0 > 1, b_0 > 0 \quad (3)$$

위의 식에서  $m$ 과  $n$ 은 팽창과 전이와 관련된 정수 파라미터이다. 따라서 DWT의 모 웨이블릿  $\psi$  다음과 같다.

$$\varphi_{m,n}(t) = a_0^{-m/2} \psi(a_0^{-m} t - nb_0) \quad (4)$$

식(4)에서,  $a_0 = 2, b_0 = 1$ 인 특별한 경우(dyadic decomposition) 위 식을 만족하는 정규직교 기저  $\varphi_{m,n}$ 를 만들 수 있다. 이러한 경우에 DWT의 실제 수행은 적절하게 설계된 Quadrature Mirror Filter(QMF)를 사용하여 효과적으로 수행될 수 있다<sup>[7]</sup>. 각 QMF 쌍은 신호의 대역폭을 반으로 분할하는 저역통과 필터( $H$ )와 고역통과 필터( $G$ )로 구성된다. 2-D DWT는 다음과 같이 분리 가능한 필터 뱅크를 적용하여 계산한다.

$$\begin{aligned} L_n(b) &= [H_x * [H_y * L_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b) \\ D_{n1}(b) &= [H_x * [G_y * L_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b) \\ D_{n2}(b) &= [G_x * [H_y * L_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b) \\ D_{n3}(b) &= [G_x * [G_y * L_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2}(b) \quad , b \in R^2 \end{aligned} \quad (5)$$

여기서,  $R^2$ 는 2차원 유클리디언 공간을, \*는 convolution operator를 나타내며,  $\downarrow 2,1(\downarrow 1,2)$ 은 행(열) 방향으로의 서브 샘플링을 의미한다.  $H$ 와  $G$ 는 각각 저대역 통과 필터(Low-Pass Filter)와 고대역 통과 필터(High-Pass Filter)이다.  $L_n$ 은 LPF에 의해 얻을 수 있으며,  $L_n$ 을 스케일  $n$ 에서의 저해상도 영상으로 부른다.  $D_{ni}$ 는 스케일에서의 상세영상으로 부르며, 수평( $D_{n1}$ ), 수직( $D_{n2}$ ), 대각( $D_{n3}$ ) 방향의 상세 정보를 포함하고, HPF통해 얻을 수 있다. 그러므로 원 영상( $I$ )은 몇몇 스케일에서의 부대역의 집합

으로 표현된다. 그리고 이것은 원영상  $I$ 의 깊이  $d$ 에서의 다중 스케일 표현이다.

$$\{L_d, D_{n,i}\}_{n=1\dots d, i=1,2,3} \quad (6)$$

이러한 분해를 DWT 또는 피라미드 웨이블릿 분해라 부른다. 분해된 모든 부대역은 특정 스케일과 방향 정보를 포함하며, 공간 영역의 정보는 부대역 안에 남아있게 된다<sup>[8]</sup>.

### III. Wavelet 기반의 영상 부호화 기법

#### 3.1 EZW 부호화 방법

EZW 부호화 방법은 웨이블릿 변환 계수들의 대역간 자기 유사성을 이용하여 영상을 압축한다. 그림 1에서 주파수가 낮은 대역에 있는 웨이블릿 계수를 parent라 하고, 이 대역보다 높은 주파수 대역내의 계수들을 descendant라 한다. 이러한 웨이블릿 트리 구조는 parent에 해당하는 계수가 작은 값을 가지면, 이 계수의 descendant들이 대부분 작은 값을 갖는 특징이 있다. EZW 부호화 방법에서는 이러한 트리 구조와 SAQ를 이용하여 영상을 부호화 한다. SAQ는  $T_i = T_{i-1}/2$ 를 만족하는 일련의 임계치  $T_1, \dots, T_N$ 들을 각 웨이블릿 계수들에 반복적으로 적용하면서 웨이블릿 계수들의 중요도를 판단한다. 여기서 초기 임계치  $T_1$ 은 모든 웨이블릿 계수  $c_{i,j}$ 에 대해  $|c_{i,j}| < 2T_1$ 의 조건을 만족하는 값이다. 현재 임계치  $T_i$ 보다 크기가 큰 계수를 중요 계수(significant)라 하고, 작은 계수를 무의미한 계수(insignificant)라 한다. SAQ 과정에서는 웨이블릿 계수들을 저주파 대역에서부터 고주파 대역까지 계층적 순서에 따라 피라미드를 스캔하면서, 각 계수의 크기가 임계치  $T_i$ 보다 큰지, 작은지를 결정한다. 만약 계수가 significant라면 계수의 부호에 따라 POS(positive significant), NEG(negative significant) 기호로 나타내고, 반대로 계수가 insignificant라면 이 계수에 대한 descendant 중에 significant 계수가 있는지를 조사하여 significant 계수가 하나도 없으면 ZTR(zerotree root) 기호를 할당하고, significant 계수가 하나라도 있으면 IZ(isolated zero) 기호를 할당한다. 이와 같이 생성된 기호들은 적응적 산술 부호화를 이용하여 무손실 부호화된다. 이러한 과정을 주어진 비트율을 만족할 때까지 임계치를 낮추어 가면서 반복적으로 수행한다.

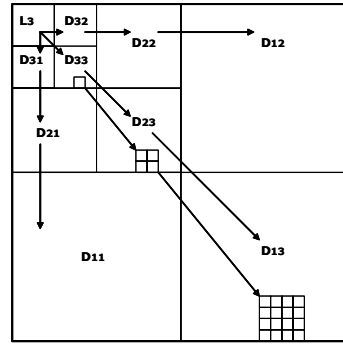


그림 1. 웨이블릿 평면상의 자기 유사성.

#### 3.2 SPIHT 부호화 방법

SPIHT는 EZW와 마찬가지로 자기 유사성 탐색이라는 개념을 근거로 한다. SPIHT 알고리즘에는 두 개의 주된 단계가 있다. 정렬 패스(sorting pass)와 세분 패스(refinement pass)이다. 정렬 패스에서는 계수들을 특정 임계값과 비교해서 중요도(significant or insignificant)를 알아내고 세분 패스에서는 정렬 패스에서 구한 중요한 계수 값을 세밀화 한다. SPIHT에서 사용되는 몇 가지 정의는 다음과 같다.

- $T$  : 특정 임계값
- $\mathcal{O}(i, j)$  : 노드  $(i, j)$ 의 모든 offspring의 좌표들의 집합
- $\mathcal{D}(i, j)$  : node  $(i, j)$ 의 모든 후손들의 좌표들의 집합
- $H$  : 가장 상위 레벨 내에 있는 노드들의 좌표들의 집합.
- $L(i, j) = \mathcal{D}(i, j) - \mathcal{O}(i, j)$

SPIHT 알고리즘은 LSP(list of significant pixel), LIP(list of insignificant pixel), LIS(list of insignificant set)라는 세 개의 리스트를 사용해서 계수들의 좌표를 관리한다. LIP는 중요하지 않은 화소들의 리스트를 의미하며, LSP는 중요한 화소들의 리스트 그리고 LIS는 중요하지 않은 집합들의 리스트를 의미한다. 처음에는 LSP를 비워두고  $H$ 에 속한 계수들을 LIP에 넣고, LIP의 계수들 중에서 descendants를 LIS에 넣는다. 정렬 패스에서 LIP를 차례대로 조사해 가면서  $T$ 보다 큰 값은 LSP에 넣고  $T$ 보다 작은 값은 LIP에 추가시킨다. Descendants에 관한 정보는  $\mathcal{D}(i, j)$ 에 대해서 A형으로,  $L(i, j)$ 에 대해서 B형으로 LIS에 넣는다. 세분 패스에서 LSP를 차례대로 세분하는 정보를 만든 후, 정렬 패스와 세분 패스를 원하는 비트율에 도달할 때까지 반복한다.

#### IV. 제안된 웨이블릿 영상 부호화기

##### 4.1 계수의 비트레벨 정보를 위한 2D SCA

웨이블릿 변환된 모든 계수들의 중요도 검사를 위해 많은 연산 시간이 소비된다. 본 연구에서는 계수의 중요도 검사를 빠르고 효과적으로 수행하기 위해 계수들의 비트 레벨 정보를 저장하기 위한 2D SCA를 제안한다. 제안된 2D SCA는 웨이블릿 계수 피라미드에서 수평 방향의 상세 정보를 저장하는  $D_{nl}$  부대역 피라미드와 같은 크기와 모양을 가지는 이차원 데이터 구조이다. 자료 구조 내의 각 원소(element)는 웨이블릿 피라미드의 동일한 레벨 또는 그 이하의 레벨에서의 동일한 위치에 있는 각 방향 부대역의 가장 큰 계수를 나타내는데 필요한 비트들의 레벨 정보를 가진다.

2D SCA의 생성을 위해 웨이블릿 계수 피라미드의 높이를  $H$ 라 하고, 너비를  $W$ 라 하자. 그리고 저해상도 영상 정보를 가지는 스케일 레벨( $L$ )를 제외한 피라미드 레벨의 수를  $S$ 라 하면, 스케일 레벨( $L$ )의 높이( $H_L$ )와 너비( $W_L$ )의 크기는  $H_L = H/2^S$ 와  $W_L = W/2^S$ 이다. 또한 이차원 자료구조 내에서 각 레벨의 크기는 다음과 같이 정의될 수 있다.

$$\begin{aligned} W_{x,y} &= W_L 2^{\lfloor \log_2(\max\{1, \frac{x}{H_L}\}) \rfloor} \\ H_{x,y} &= H_L 2^{\lfloor \log_2(\max\{1, \frac{y}{H_L}\}) \rfloor} \end{aligned} \quad (7)$$

그리고, 2D SCA의 원소  $m_{x,y}$ 는 그림 6과 같이  $m_t(x,y)$ ,  $m_c(x,y)$  그리고  $m_s(x,y)$ 로 정의될 수 있다.  $m_t(x,y)$ 는 웨이블릿 계수들 중 가장 상위의 레벨( $L$ )에 있는 중요 계수 정보를 가진다.  $m_s(x,y)$ 는 웨이블릿 계수들 중 최하위 레벨(Level 1)에 있는 계수 정보를 가지며,  $m_c(x,y)$ 는 그 이외의 레벨에 존재하는 계수 정보를 포함한다. 따라서 중요계수 배열의 원소  $m_{x,y}$ 에 대하여 정리하면 아래 식과 같다.

$$\begin{aligned} m_t(x,y) &= \max\{m_{x,y+H_S}, \\ &\quad \lfloor \log_2(\lfloor c_{x,y} \rfloor) \rfloor + 1\} \\ m_c(x,y) &= \max\{m_{2x,2y}, m_{2x+1,2y}, \\ &\quad m_{2x,2y+1}, m_{2x+1,2y+1}\} \\ m_s(x,y) &= \lfloor \log_2(\max\{|c_{x,y}|, |c_{x+W_x,y}|, \\ &\quad |c_{x+W_x,y-H_x}| \}) \rfloor + 1 \end{aligned} \quad (8)$$

$$m_{x,y} = \begin{cases} m_t(x,y), & \text{if } x < W_L \wedge y < H_L \\ m_s(x,y), & \text{if } x \geq W/2 \wedge y \geq H/2 \\ \max\{m_s(x,y), m_c(x,y)\}, & \text{otherwise} \end{cases} \quad (9)$$

여기서  $x, y$ 는 각각  $0 \leq x < H$ ,

$0 \leq x < W_L 2^{\lfloor \log_2(\max\{1, \frac{x}{H_L}\}) \rfloor}$ 의 범위를 가진다. 그림 2에 식 8, 9에 의한 2D SCA 생성을 위한 알고리즘을 나타내었고, 그림 3에 2D SCA 생성 과정을 보였다.

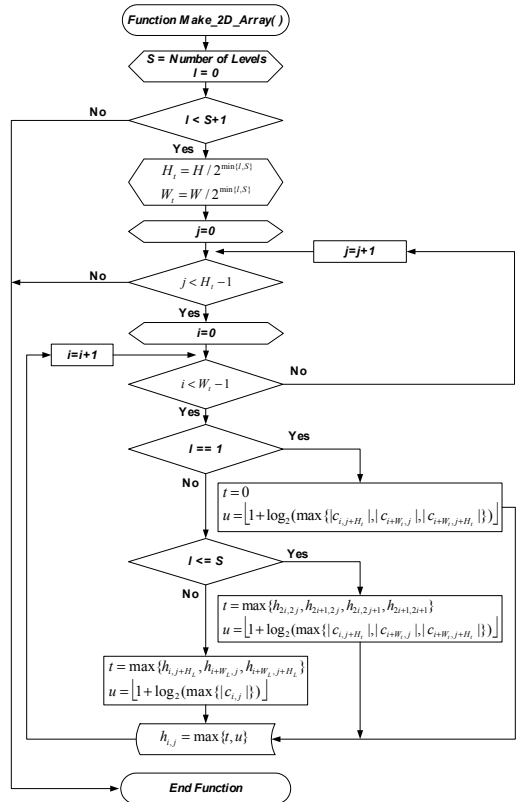


그림 2. 2D 중요계수 배열생성을 위한 흐름도

##### 4.2 부호화 알고리즘

제안된 부호화 알고리즘의 전체적인 내용은 그림 4와 같다. 1) 공간 영역의 영상은 웨이블릿 분해에 의하여 웨이블릿 공간으로 변환된다. 2) 계수들의 비트-레벨 정보 저장을 위한 이차원 중요계수 배열을 생성하고, 3) 적응 이진 산술 부호화기(QM-부호화기)가 초기화 된다. 4) 웨이블릿 계수들은 비트 스케닝 알고리즘에 의해 최상위 비트부터 스캔된다. 복호화 과정은 부호화 과정의 역으로 수행된다.

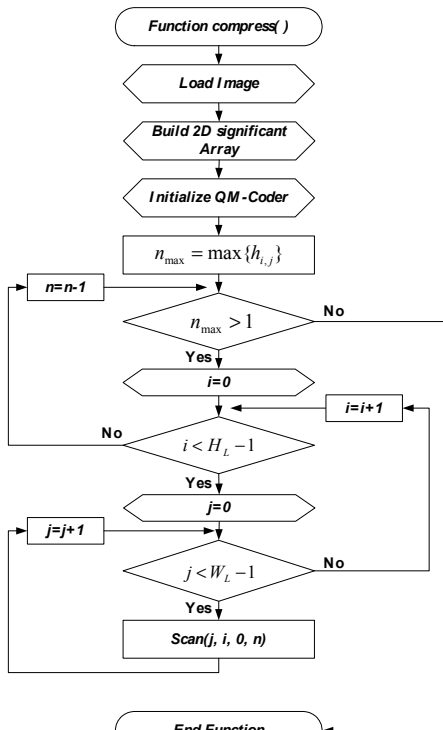
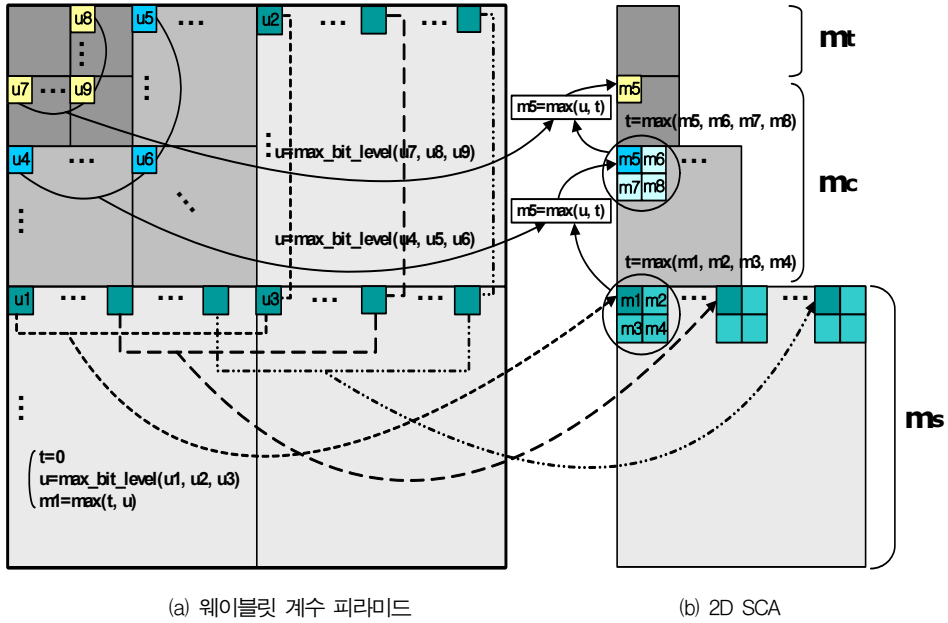


그림 4. 부호화 알고리즘

그림 5의 비트 레벨 계수 스캐닝 알고리즘은 부호화 알고리즘의 주요 부분이다. 이 알고리즘은 제

일 먼저 부호화 완료 조건을 검사한다. 여기서 완료 조건은 부호화 명령시 주어지는 파라미터 값으로 원하는 화질을 나타내는 비트율이다. 그리고 2D SCA를 검색하여 현재의 위치에서 제로트리로 시작하는지를 조사한 후 그 결과를 부호화 한다. 만일 계수가 제로트리로 시작한다면, 현재 피라미드 레벨 이하의 계수에 대한 비트 부호화를 생략할 수 있다. 또한 피라미드의 최상위 레벨인 스케일링 계수(L=0)를 부호화할 경우, 스캐닝 알고리즘은 재귀적(recursive)으로 구조로 되어있어 현재 피라미드 레벨 이하의 다음 레벨에 위치하는 동일 위치의 계수를 순환적으로 검색한다. 그리고 최상위 레벨 이하의 계수들을 부호화 할 경우에는 현재 위치내의 수평, 수직, 대각 방향 부밴드의 세 계수들을 모두 처리하게 되며, 만일 그 다음 하위 레벨이 존재한다면 그 레벨의 네 개의 계수(descendant)들을 모두 재귀적으로 스캐닝된다.

그림 6은 계수 부호화를 위한 *Coeff\_Scan* 알고리즘이다. 이 알고리즘에서 *Bit\_Return(x, n)* 함수는 *x*의 절대값에서 *n* 번째 비트값을 반환하며 *s<sub>i,j</sub>*는 (*i, j*)에 위치하는 웨이블릿 계수의 부호이고, *c<sub>x,y</sub>*는 (*x, y*)에 위치하는 웨이블릿 계수를 의미한다. *Bit\_Return(x, n)*에 의해 생성된 비트들은 이진 산술 부호화 기능을 담당하는 *QM\_coder(b, context)*에 의해 산술 부호화된다.

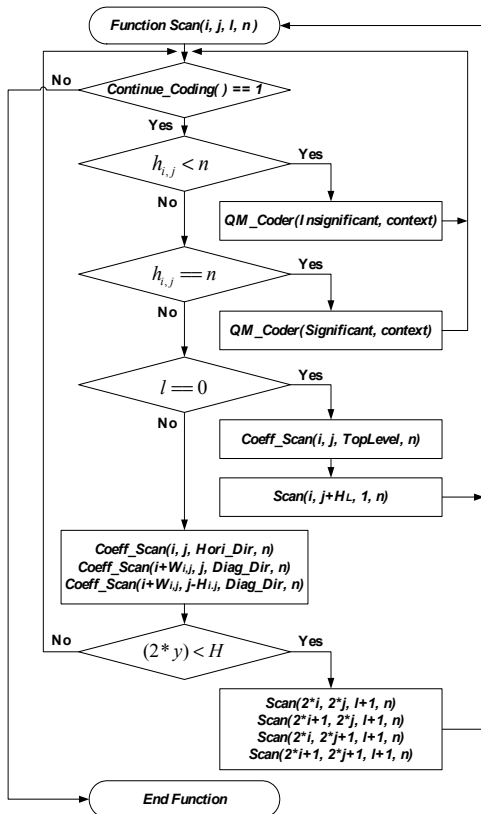


그림 5. 비트 레벨 스캐닝 알고리즘

여기서 인수  $b$ 는 부호화되는 비트를 의미하며  $context$ 는 현재 부호화되는 심볼에 대한 확률을 추정하기 위한 패턴을 의미한다. 부호화 알고리즘의 완료는 함수  $Continue\_Coding()$ 에 의해 결정된다. 이 함수는 완료를 조건을 계산하기 위해 부호화가 진행되는 동안 주어진 비트율을 만족했는지를 계산하여 참 또는 거짓을 반환한다. 만일  $Continue\_Coding()$  함수의 반환값이 참이라면 부호화는 계속되고 거짓이면 부호화는 완료된다. 부호화 알고리즘의 완료 조건을 계산하기 위해 부호화가 진행되는 동안 현재 부호화 단계에서 근사화된(approximated) 영상의 화질이 계산된다. 이것은 계수  $x$ 의  $n$ 번째 비트가 부호화 될 때마다  $Check\_Bpp()$ 를 호출하는 것으로 수행된다. 부호화가 진행되는 동안 수행되는 완료 조건의 계산은 MSE나 Bpp를 사용한다.

$Coeff\_Scan$  알고리즘은 처음 계수가 앞서 중요계수로 판별되었는지에 대한 여부를 조사한다. 만일 계수가 중요계수로 판별되었다면, 계수의 비트 레벨중 현재의 비트를 부호화하고 주어진 평균 제곱 오차(MSE)비트율(bpp)을 만족하는지를 계산한다.

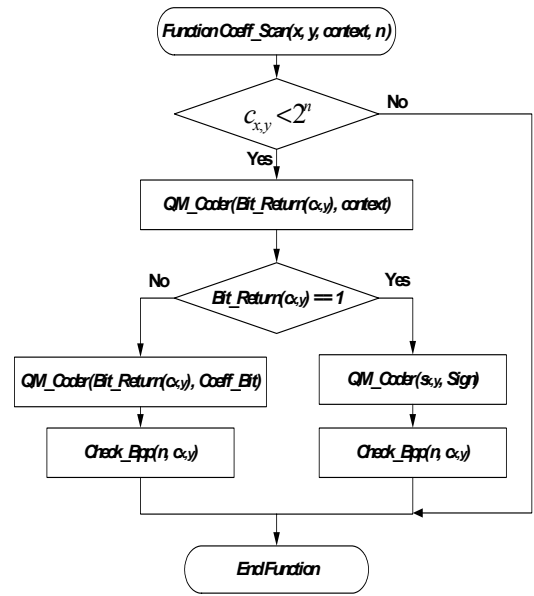


그림 6. 계수의 부호화를 위한 세부 알고리즘

그렇지 않고 만약 계수가  $2^n$ 보다 작다면, 현재 비트 레벨의 비트를 부호화하고 계수의 최상위 비트(MSB)가 0인지 1인지를 조사한다. 만약 MSB가 1이라면 계수의 부호를 부호화하고 주어진 비트율(bpp)을 만족하는지를 계산한다.

### 4.3 부호화 완료조건

제안된 부호화기가 부호화를 완료해야 하는 시점은 다음과 같은 조건이 발생한 경우이다. 첫 번째로 웨이블릿 계수의 모든 비트가 부호화된 경우이고, 두 번째의 경우 QM-부호화기에 의해 생성되는 비트율이 부호화시 파라미터로 주어진 비트율에 도달했을 때이다. 그리고 마지막으로 출력 영상의 화질이 주어진 임계치(threshold) 이하로 나타날 경우에 부호기는 부호화를 완료해야 한다. 먼저, 첫 번째 조건의 경우 계수의 모든 비트들이 부호화되면 부호화 알고리즘이 종료되기 때문에 크게 문제되지 않는다. 두 번째 조건 역시 QM-부호화기의 출력 루틴에서 나오는 비트나 바이트의 수를 계산하는 것으로 쉽게 수행될 수 있다. 그러나 세 번째의 경우 부호화 알고리즘은 복원된 영상의 MSE를 반드시 알고 있어야 하는 문제점이 있다. 복원된 영상의 MSE는 모든 웨이블릿 계수들에 대한 웨이블릿 역변환을 수행해야 하고, 그 결과로부터 MSE를 계산하여야 한다. 반면, 부호화 알고리즘은 매우 빈번하게 완료 조건을 검사

해야 하기 때문에, 이 방법을 그대로 사용할 경우 부호화 알고리즘의 성능을 저하시키는 요인이 된다. 따라서 본 논문에서는 Daubechies 웨이블릿 변환의 정규직교성(orthonormality)을 이용한 수정된 MSE방법을 사용한다. 웨이블릿 변환과 같은 정규직교 변환은 식(10)과 같이 변환되기 이전의 영상 픽셀값과 변환 후의 값이 동일하다.

$$\sum_{i,j} (x_{i,j})^2 = \sum_{i,j} (c_{i,j})^2 \quad (10)$$

여기서  $x_{i,j}$ 는 공간 영역의 픽셀의 명암값을 의미하고,  $c_{i,j}$ 는 웨이블릿 계수를 의미한다. 또한 원 영상과 어떤 단계의 근사화된 영상간의 MSE는 공간 영역과 웨이블릿 영역에서 동등하게 계산될 수 있으므로, MSE를 계산하기 위해서 반드시 웨이블릿 역변환을 수행할 필요는 없다는 것을 알 수 있다. 따라서 본 알고리즘에서는 원 영상과 근사화된 영상간의 MSE를 계산하는 것 대신, 수정된 MSE( $mMSE$ ) 기법을 사용하여 계산 효율을 증대시켰다. 즉, 부호화 초기에 사용되는  $mMSE$  값은 식(11)과 같이 웨이블릿 계수값의 에너지와 같다.

$$mMSE = \sum_{i,j} (c_{i,j})^2 \quad (11)$$

예를 들어 계수  $c$ 의 각 비트가 보내지면, 계수  $c$ 의 이전 approximation에 의해 산출되었던 에러를 빼서  $mMSE$ 를 갱신하고 계수의 새로운 approximation 에러를 추가한다. 계수  $c$ 의 approximation 에러는 마지막 비트의 레벨과 계수  $c$  자체에 의해서만 결정되는데, 만일 계수  $c$ 의  $n$  번째 비트 부호화에 대한  $mMSE$ 는 식(12)와 같은 방법에 의해 갱신된다.

$$mMSE = mMSE_{prev} - \begin{cases} [(|c| \& (2^n - 1)) - 2^{n-1}]^2 - [(|c| \& (2^{n-1} - 1)) - 2^{n-2}]^2 \\ c^2 - (2^{n-1} + 2^{n-2} - c)^2 \\ 0 \end{cases} \quad (12)$$

여기서  $\&$ 는 비트 논리곱을 의미한다. 식(12)는  $n > 1$ 일 경우에만 적용되며, 식에서 첫 번째는  $\lfloor \log_2 c \rfloor > n - 1$ 인 경우이다. 이것은 계수의 부호를 이미 알고있을 때, 부호를 제외한 계수의 비트값이 1인 비트를 찾아내어  $mMSE$ 를 계산하는 방법에 대한 정의이다. 두 번째는  $\lfloor \log_2 c \rfloor = n - 1$ 일 때,

계수의  $n$  번째 비트값이 부호 비트일 때의  $mMSE$  대한 정의이며, 마지막은  $\lfloor \log_2 c \rfloor < n - 1$ 인 경우이며, 이것은 계수의 비트값이 처음 비트 1을 만나기 전 까지의 비트 0에 대한 정의로서, 이때  $mMSE$  값은 변경되지 않음을 의미한다.

#### 4.4 컨텍스트 기반의 적응 산술 부호화기

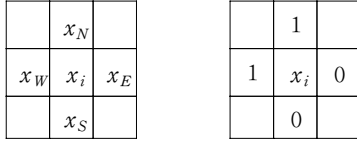
계수 스캐닝 알고리즘에 의해 생성된 비트들은 산술 부호화기 보다 우수한 성능을 보이는 QM-부호화기를 이용하여 부호화한다. 적응 산술 부호화 기법의 일종인 QM-부호화기는 JBIG(Joint Bi-level Image Experts Group)에서 규정한 이진 영상 압축을 위한 엔트로피 부호기이다<sup>9)</sup>.

QM-부호화기의 핵심은 고정된 부호구간  $[0.0, 1.0)$  내에서의 두 개의 확률구간 부활을 기반으로 한다는 것이며, 이것은 부호화가 종료될때까지 재귀적으로 수행한다. 개념적으로 특정 컨텍스트(context)를 기반으로 입력되는 이진 심볼들은 구간  $[0.0, 1.0)$  상의 실수  $r$ 로 사상된다. 컨텍스트는 인접한 화소들의 위치를 표시하는 특정한 패턴을 대변하는 정수값이며, 현재의 화소를 부호화하기 위해 사용되는 QM-부호화기의 확률구간 상태를 색인 형태로 제공한다. 그리고 QM-부호화기에서는 현재 심볼에 대한 확률을 추정하는데 있어서 인접한 픽셀들의 패턴을 사용하는데, 이를 컨텍스트 모델링이라 한다. 컨텍스트 모델링을 어떻게 하느냐에 따라서 QM-부호화기의 성능을 향상시킬 수 있다<sup>10~11)</sup>.  $x_1, x_2, \dots, x_n$ 들로 이루어진 심볼들을 부호화하는데 필요한 최소 코드길이  $l$ 은 다음과 같은 식으로 주어진다.

$$l = -\log_2 \prod_{i=1}^n p(x_i | x_{i-1}, x_{i-2}, \dots, x_1) \quad (13)$$

QM-부호화기에서는 심볼의 분포를 추정하기 위해서  $x_1, x_2, \dots, x_n$ 에 대한 서브시퀀스  $S$ 를 사용하는데, 본 논문에서 사용한 컨텍스트 모델은 그림 7(a)와 같다. 그리고 계수의 부호화를 위해 서로 다른 확률 분포 따라 부호, 완료조건, 계수의 첫 번째 한 비트 다음의 비트값, 최상위 레벨의 스케일링 계수 비트값, 웨이블릿 피라미드에서 각 레벨의 제로트리 정보 그리고 각 방향의 피라미드에서 서로 다른 레벨에 위치하는 insignificant 계수의 중요도 검사와 같은 데이터

들의 컨텍스트가 정의된다. 그림 7(b)는 본 논문에서 사용한 컨텍스트 모델의 예를 나타낸다. 즉, 현재 부호화하고자 하는  $x_i$  비트에 대한 확률 분포는  $S = \{x_N, x_W, x_E, x_S\} = \{1, 1, 0, 0\}$ 를 사용하여 추정한다. 여기서  $x_N$ 와  $x_W$ 는 현재 임계치  $L_i$ 일 때의 SAQ에 의해 생성된 값이고,  $x_E$ 와  $x_S$ 는 바로 이전 임계치인  $L_{i-1}$ 일 때의 SAQ에 의해 생성된 값이다.



(a) 컨텍스트 모델 (b) 컨텍스트의 예  
그림 7. 컨텍스트 모델링

V. 실험 및 결과

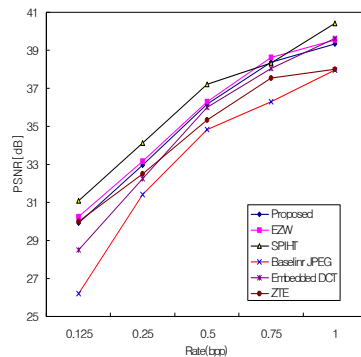
본 논문에서 제안한 부호화의 성능을 평가하기 위해 Pentium-II 450 Linux 환경에서 C언어를 사용하여 구현되었다. 실험에 사용된 영상은 512x512 크기를 가지는 Lena와 Barbara를 이용하였으며 웨이블릿 변환을 위해 Daubechies 9/7 웨이블릿 필터를 사용하였다. 그리고 알고리즘의 성능 평가를 위해 EZW, SPIHT, Baseline JPEG, EZDCT<sup>[12]</sup> 그리고 MPEG-4에서 사용되는 ZTE 방법<sup>[13]</sup>과의 PSNR(Peak Signal to Noise Ratio)을 이용하여 화질을 평가하였다.

표 1과 그림 8는 비트율에 따른 PSNR 성능을 나타낸다. Lena 영상과 Barbara 영상 모두에서 SPIHT 보다는 약간 떨어지지만, JPEG보다는 월등히 높으며 EZW와 비슷한 성능의 결과를 나타내었다. 또한 0.5 bpp 이하의 낮은 비트율에서도 다른 방법들과 유사하게 좋은 성능을 보였다. 그림 9는 제안한 방법에 의한 복원영상과 원영상을 나타내었다. 그림에서 볼 수 있는 바와 같이 낮은 bpp에서 복원된 영상도 블록간 왜곡이 없는 좋은 화질을 보임을 알 수 있다. 또한 부호화시 필요한 메모리 사용량의 경우, 추정 방법의 어려움으로 정확한 정량적인 측정은 할 수가 없었다. 그러나 기존 EZW나 SPIHT를 사용할 경우 이론적으로 약  $8(\text{계수의 자료형이 } 8\text{비트 정수}) \cdot N$  비트의 메모리가 부호화를 위해 필요하지만, 제안한 부호화기의 메모리 사용량은 약  $8 \cdot N/3$  비트의 메모리만 요구된다.

여기서 N은 계수의 개수이다. 즉, 계수 자료형이 16( $2^4$ )비트 크기를 가지는  $2^n \cdot 2^n$  크기의 영상을 부호화 할 경우에 기존의 부호화기가  $2^n \cdot 2^n \cdot 2^4$  비트의 메모리 사용량을 가진다면, 제안된 부호화기는  $\left[ \left( \sum_{i=1}^n 2^{m-1} \cdot 2^{m-1} \right) + (2^3 \cdot 2^3) \right] \cdot 2^4$  비트의 메모리만을 사용한다. 예를 들어 512x512 크기의 Lena 영상을 6-레벨 웨이블릿 변환하여 계수들을 부호화 할 때 기존 부호화기는  $2^9 \cdot 2^9 \cdot 2^4$  (524,288바이트)의 메모리를 사용하지만, 제안된 부호화기의 메모리 사용량은  $[(2^8 \cdot 2^8 + \dots + 2^3 \cdot 2^3) + (2^3 \cdot 2^3)] \cdot 2^4$  (174,848바이트)이다. 따라서 제안된 부호화기는 기존의 부호화기에 비해 약 33.35%의 메모리 사용량만으로 영상을 부호화할 수 있으며, 기존 부호화기에 비해 약 66.65%의 정도의 메모리 사용량을 줄일 수 있다. 따라서 제안된 부호화기는 해상도가 높은 영상을 처리할 경우에 더욱 효과적으로 수행될 수 있음을 알 수 있다.

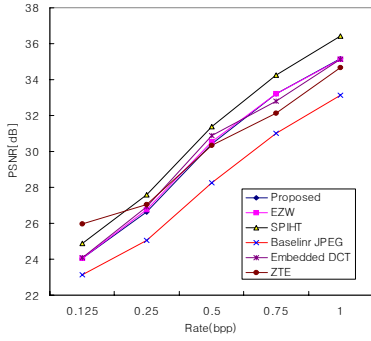
표 1. 비트율에 따른 PSNR 결과

bpp		압축방법				
		0.125	0.25	0.5	0.75	1.0
Lena	Proposed	29.93	32.97	36.15	38.36	39.34
	EZW	30.23	33.17	36.28	38.64	39.55
	SPIHT	31.09	34.11	37.21	38.32	40.41
	JPEG	26.21	31.42	34.84	36.6	37.95
	EZDCT	28.5	32.27	35.98	38.04	39.61
	ZTE	30.01	32.51	35.32	36.75	37.95
Barbara	Proposed	24.04	26.61	30.41	33.19	35.15
	EZW	24.03	26.77	30.53	33.20	35.14
	SPIHT	24.86	27.58	31.39	33.51	36.41
	JPEG	23.11	25.05	28.27	31.01	33.11
	EZDCT	24.07	26.93	30.87	32.81	35.13
	ZTE	25.97	27.04	30.34	32.12	34.68



(a) Lena 영상





(b) Barbara 영상

그림 8. 비트율에 따른 PSNR 그래프

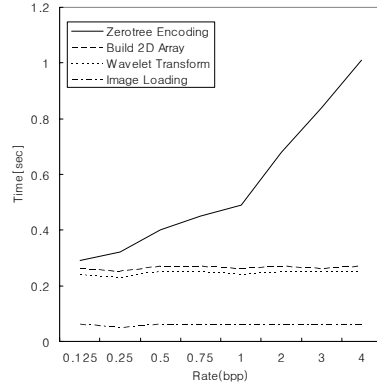
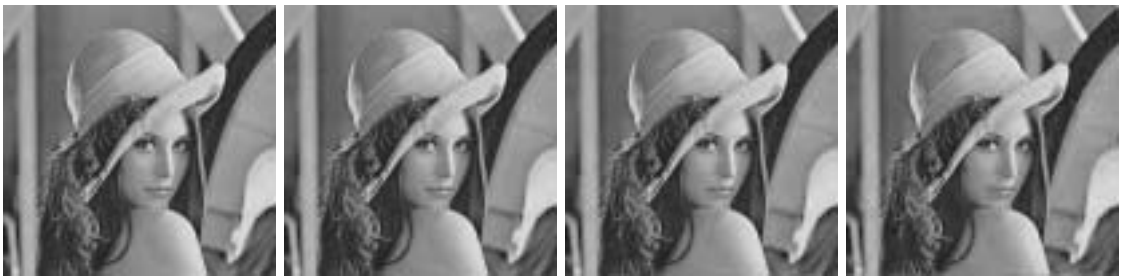


그림 10. Lena 영상의 부호화 소요 시간(단위:sec)

그림 10에 제안한 방법을 이용하여 Lena 영상을 부호화/복호화 하는데 필요한 누적 시간을 나타내었다. 표에서와 제로트리 부호화와 복호화 속도의 경우 bpp가 올라갈수록 소요되는 시간이 늘어났고, 그 이외의 다른 항목들은 영상의 크기에 따라 소비 시간이 크게 달라졌다. 특히 대부분의 시간이 웨이블릿 변환에 소비 되었으며, 본 논문에서 사용된 2D SCA를 구성하는데 소비되는 시간은 크지 않았다. 그리고 기존의 방법과의 부호화 및 복호화 속도 비교는 프로그램 작성 방식과 코드의 최적화 방법에 따라 큰 속도 차이를 보이기 때문에 본 논문에서는 논하지 않았다.

## VI. 결론

본 논문에서는 2D SCA를 사용한 웨이블릿 부호화기를 제안하였다. 제안된 부호화기는 웨이블릿 계수의 비트 레벨 정보를 2D 중요계수 배열에 저장하고, 제로트리 부호화시 이 정보를 이용하여 계수의 중요도 검사와 비트 스캐닝을 한번에 수행시키는 구조이다. 따라서 기존의 알고리즘이 가지는 부호화 과정의 중복된 계수 스캐닝 과정을 줄일 수 있었으며, 부호화 과정에 필요한 메모리 사용량도 기존 방법에 비해 66% 이상 크게 낮출 수 있었다. 또한 비트율 제어에 의한 부호화 완료 조건 검사를 위해 수정된



(a) Lena original image

(b) 0.5bpp

(c) 0.25bpp

(d) 0.125bpp



(a) Barbara original image

(b) 0.5bpp

(c) 0.25bpp

(d) 0.125bpp

그림 9. 제안한 부호화 방법에 의한 복원 영상

MSE 기법을 이용하여 웨이블릿 역변환 과정없이 현재의 계수값만을 이용하여 비트율 제어가 가능하였으며, 이진 심볼을 엔트로피 부호화하는데 있어서 적응 산술 부호화기를 사용하여 부호화 효율을 증대시켰다. 그리고 제안한 방법을 기존의 EZW, SPIHT, JPEG, EZDCT 그리고 ZTE 방법들과 성능을 비교한 결과, 기존 방법들에 비해 비슷하거나 비교적 우수한 성능을 보임을 확인하였다.

### 참고문헌

[1] C. Sidney Burrus, Ramesh A. Gopinath, Introduction to Wavelet and Wavelet Transforms, Prentice-Hall, New Jersey, 1998.

[2] J. Y. Yham, S. Ranganath and A. A. Kassim, "Highly scalable wavelet-based video codec for very low bit-rate environment", IEEE Trans. Selected Areas in Comm., vol. 16, no. 1, pp.12-27, Jan. 1998.

[3] S. N. Efstratiadis, D. Tzovaras and M. G. Strintzis, "Hierarchical partition priority wavelet image compression", IEEE Trans. Image Processing, vol. 5, no. 7, pp. 1111-1123, Jul. 1996.

[4] Z. Xiong, K. Ramchandran and M. T. Orchard, "Space-frequency quantization for wavelet image coding", IEEE Trans. Image Processing, vol. 6, no. 5, pp.677-693, May 1997.

[5] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients", IEEE Trans. Signal Processing, vol. 41, no. 12, pp.3445-3462, Dec. 1993.

[6] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees", IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, pp.243-250, June 1996.

[7] S. Mallat, "A Theory for multiresolution signal decomposition: the wavelet representation", IEEE Trans. Pattern

Analysis and Machine Intelligence, vol. 11, pp.674-693, 1989.

[8] Scheunders P., Livens S., Van de Wouwer G., Vautrot P., and Van Dyck D., "Wavelet-based texture analysis, International Journal on Computer Science and Information Management", vol. 1, no. 2, pp. 22-34, 1998.

[9] CCITT Draft Recommendation T.82 ISO/IEC Draft International Standard 1544 Coded Representation of Picture and Audio Information Progressive Bi-level Image Compression Recommendation, WG9-S1R5.1, Apr. 1992.

[10] V. R. Algazi, R. R. Estaes Jr., "Analysis based coding of image transform and subband coefficients", In SPIE, Applications of Digital Image Processing XVII, pp.11-21, 1995.

[11] C. Chrysafis, A. Ortega, "Efficient context-based entropy coding for lossy wavelet image compression", In Proc. IEEE Data Compression Conf. '97, pp. 241-250, 1997.

[12] Z. Xiong, O. Uleryuz, M. T. Orchard, "A DCT-based Embedded Image Coder", IEEE Signal Processing Letters, Nov. 1996.

[13] S. A. Martucci, I. Sodagar, T. Chiang and Y. Zhang, "A Zerotree Wavelet Video Coder", IEEE Trans. Circuits and Systems for Video Technology, vol.7, n.1, pp. 109-118, Feb. 1997.

박성욱(Sung-Wook Park) 정회원  
한국통신학회논문지 제27권 제9호 참조

박종욱(Jong-Wook park) 정회원  
한국통신학회논문지 제27권 제9호 참조