

Threshold-adjusted EZW Codec의 설계와 구현

채 희 중[†] · 이 호 석^{††}

요 약

본 논문은 웨이블릿 계수를 활용하여 영상 압축을 수행하는 EZW(Embedded Zerotree Wavelet) 부호화의 성능 개선에 대한 방법을 제시한다. EZW 부호화는 웨이블릿 계수를 그 중요도에 따라 POS, NEG, IZ, ZTR의 4개의 심벌을 사용하여 부호화를 수행한다. 본 연구에서는 EZW 부호화에 웨이블릿 계수 threshold 방법을 적용하여 성능을 개선하였다. 시스템 전체적으로는 계수 threshold 방법을 적용하여 전체 부호화 시간을 단축하였으며, run-length 부호화와 산술 부호화를 적용하여 영상을 최대한으로 압축하였다. 마지막으로 실험을 통하여 EZW 부호화 방법에 적용된 계수 조정 값을 조절하여, 값의 변화에 따른 입력 영상과 복원 영상을 비교 분석한 결과를 제시하였다.

A Design and Implementation of Threshold-adjusted EZW Codec

Hi-jung Chae[†] · Ho Suk Lee^{††}

ABSTRACT

In this paper, we propose a method for the improvement of EZW encoding algorithm. The EZW algorithm encodes wavelet coefficients using 4 symbols such as POS(POSitive), NEG(NEGative), IZ(Isolated Zero), and ZTR(ZeroTreeRoot) which are determined by the significance of wavelet coefficients. In this paper, we applied threshold to wavelet coefficients to improve the EZW algorithm. The coefficients below the threshold are adjusted to zero to generate more ZTR symbols in the encoding process. The overall EZW image compression system is constructed using run-length coding and arithmetic coding. The system shows remarkable results for various images. We finally present experimentation results.

키워드: 웨이블릿 변환(Wavelet transform), 웨이블릿 계수 조정(Wavelet coefficient threshold), EZW

1. 서 론

현재 사용되고 있는 정지 영상 압축 방법은 JPEG(Joint Photographic Experts Group)이다. JPEG은 DCT(Discrete Cosine Transform)와 허프만부호화를 사용하고 있다. JPEG은 한 화소 당 비트율(Bits Per Pixel ; bpp)이 0.3~0.4일 경우에는 사람의 눈에 보기에 부담이 없고 영상의 제곱 평균 오차(Mean Square Error ; MSE)도 우수하게 나타난다. 그러나 그 이하의 bpp로 압축을 수행하게 되면 블록간의 경계가 두드러지게 나타나는 현상이 발생한다[6]. 이것은 JPEG 부호화 방법에서 사용하는 DCT가 4×4혹은 8×8의 블록으로 영상을 분해한 후 각각의 독립된 블록에 대하여 부호화를 수행함으로써 나타나게 되는 현상이다[11]. 즉, DCT의 이러한 블록 기반 부호화 방법은 각각의 블록들을 독립적으로 처리하

기 때문에 부호화률이 높아질수록 인접 블록간의 연속성이 손실되어 블록간의 경계 현상을 유발하게 되는 것이다.

최근 영상을 압축하기 위한 새로운 방법으로서 활발히 연구되고 있는 것이 바로 웨이블릿 변환을 이용한 영상 부호화 방법이다[1]. 웨이블릿 변환은 푸리에 변환(Fourier transform)이나 DCT와는 달리 주파수와 시간에 대한 정보 모두를 가지고 있으므로, 영상의 압축, 컴퓨터 비전(computer vision), 잡음 제거(denoising), 특징소 추출(feature extraction) 등 다양한 분야에서 널리 유용하게 사용되고 있다[5]. 특히 웨이블릿을 이용한 영상의 압축 방법은 JPEG의 DCT를 이용한 영상 압축 방법과는 달리 입력 영상을 블록 단위로 분해하지 않고 처리를 하기 때문에 JPEG과 같은 블록 기반의 부호화에서 발생하는 블록간 경계 현상이 발생하지 않는다. 또한 멀티미디어 정보의 전송에 있어서 스케일러빌리티(scalability)의 기능을 구현하기가 용이하다는 것이 큰 장점이다. 웨이블릿의 이러한 특징으로 인하여 현재 영상 부호화의 새로운 표준안으로서 마련되고 있는 JPEG-2000[13] 및

[†] 정희원 : 연구원

^{††} 정희원 : 호서대학교 컴퓨터공학부 교수

논문접수 : 2001년 5월 11일, 심사완료 : 2001년 12월 7일

MPEG-4 등은 웨이브릿을 이용한 방법을 채택하고 있다.

EZW(Embedded Zerotree Wavelet) 부호화는 웨이브릿 변환된 영상 계수의 특성을 적절히 이용하여 부호화를 수행한다. 웨이브릿 변환된 영상은 주파수 대역별로 밀접한 자기상관 관계를 갖는 다해상도의 영상으로 분할된다. EZW 부호화는 이러한 분할된 영상에서의 대역간 자기-상관성(self-similarity)을 지니고 있는 계수들의 특징과 제로트리(zerotree) 부호화 방법을 적용한 간단하면서도 효율적인 영상 부호화 방법이다[6]. EZW 부호화 방법은 임의의 threshold에 대하여 특정 주파수 대역에 존재하는 영상의 계수가 중요하지 않다면 이 대역에 대응 관계를 갖는 상위 대역에 존재하는 영상 계수들 역시 중요하지 않을 확률이 높다는 개념에서 출발한다. 즉, 낮은 주파수 대역으로부터 높은 주파수 대역에 이르기까지 임의의 threshold보다 작은 계수를 제로트리로 구성함으로써 전송해야 하는 위치 정보의 수를 효과적으로 감소시킬 수 있다. 또한 제로트리를 이용한 부호화 방법은 원하는 비트 수를 모두 만족시킬 수 있으며 영상 내에서 지배적인 에너지를 갖는 정보부터 순차적으로 전송하기 때문에 점진적인 전송에 매우 유리하다[6].

본 논문에서는 웨이브릿 변환을 이용한 영상 서브밴드 분할과 제로트리를 이용한 EZW 부호화 과정을 소개한다. 그리고 EZW 부호화 과정을 수행하기 전 threshold로부터 계산한 값을 이용하여 웨이브릿 계수 값을 조정하여 압축율을 높이고 부호화 시간을 단축시키는 threshold-adjusted EZW 부호화 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 연구와 관련된 기본 이론으로서 이산 웨이브릿 변환, MRA, 영상의 서브밴드 분할에 대하여 소개한다. 3장에서는 EZW 부호화 방법과 엔트로피 부호화 방법, 그리고 본 논문에서 제안한 threshold-adjusted EZW 부호화 방법에 대하여 소개한다. 4장에서는 본 논문에서 제안한 threshold-adjusted EZW 부호화 방법과 EZW 부호화 방법[6]에 대하여 모의 실험을 수행하여 각각의 방법으로 복원된 영상을 비교 분석하였다. 마지막으로 5장에서는 결론을 맺었다.

2. 기본 이론

본 연구의 배경은 앞서도 언급했듯이 JPEG과 같은 블록 기반의 DCT 변환을 이용하는 영상 부호화 방법은 블록간의 연속성이 깨짐으로서 발생하는 블록간 경계 현상이 두드러지게 나타나는 단점이 있다. 하지만 웨이브릿 기반의 영상 부호화 방법은 이러한 현상이 발생하지 않는 특성을 가지고 있다. 그리고 네트워크를 통한 점진적인 전송에 유리하다. 이러한 특징으로 인하여 웨이브릿 기반의 영상 부호화 방법은 영상

부호화의 표준으로서 자리를 잡아가고 있는 추세에 있다[12, 14]. 본 장에서는 EZW 부호화를 수행하기 위한 웨이브릿 변환과 기본 영상 처리 이론에 대하여 소개한다.

2.1 이산 웨이브릿 변환(Discrete Wavelet Transform)

이산 웨이브릿 식은 다음과 같다.

$$f(t) = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} d(k, l) 2^{-k/2} \psi(2^{-k}t - l) \quad (1)$$

식 (1)에서 변수 k 와 l 은 정수 값을 갖는다. 다시 말하면, $k, l \in \mathbb{R}$ 이며 $k, l \in \mathbb{Z}$ 으로서 불연속적인 값을 가지고 있다. 식 (1)에서 dilation으로 2^k 값을 취하므로 translation은 $2^k l$ 을 취한다.

$d(k, l)$ 의 값은 $a=2^k, b=2^k l$ 의 변수를 갖는다. 2차원의 시퀀스를 갖는 $d(k, l)$ 은 $f(t)$ 의 DWT(Discrete Wavelet Transform)이라 한다. DWT는 함수 $f(t)$ 와 웨이브릿 사이의 내적(inner product)으로도 나타낼 수 있다.

$$d(k, l) = \langle f(t), \psi(2^k t - l) \rangle \quad (2)$$

DWT는 여전히 연속-시간 신호의 변환이며 단지 dilation과 translation인 실수 a 와 b 를 정수로 이산화한 형태이다. 이산 웨이브릿 변환 값을 가지고 완벽하게 원래의 신호를 복원할 수 있다.

2.2 MRA(Multi-Resolution Analysis)

본 절에서는 공리적인 정의(axiomatic definition)인 MRA(Multi-Resolution Analysis)를 정의한다[2, 4].

식 (3)은 MRA를 정의하기 위한 내포된 서브스페이스간의 포함 관계를 나타낸다.

$$\dots V_3 \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset V_{-3} \dots \quad (3)$$

식 (3)의 관계식을 고려하여 다음의 관계가 있음을 알 수 있다[2].

서브스페이스의 합집합은 모든 적분 가능한 함수 $L^2(\mathbb{R})$ 을 나타낸다.

$$\bigcup_m V_m = L^2(\mathbb{R}), \quad m \in \mathbb{Z} \quad (4)$$

서브스페이스의 교집합은 0인 함수(all-zero function)만을 포함하는 집합이다.

$$\bigcap_m V_m = \{0\}, \quad m \in \mathbb{Z} \quad (5)$$

스케일 특성은 다음과 같다.

$$f(t) \in V_m \Leftrightarrow f(2^m t) \in V_0 \tag{6}$$

$$f(2^{-1} t) \in V_{m+1}, f(t) \in V_m, f(2^1 t) \in V_{m-1}$$

쉬프트 특성은 다음과 같다.

$$f(t) \in V_0 \Leftrightarrow f(t-n) \in V_0, \text{ for all } n \in Z \tag{7}$$

Basis의 존재는 다음을 통하여 알 수 있다.

즉, $\phi \in V_0$ 의 관계가 성립한다면

$$\{\phi(t-n) \mid n \in Z\} \tag{8}$$

$$\langle \phi(t), \phi(t-n) \rangle = \delta(n)$$

의 관계가 성립한다.

스케일링 함수는 V_0 에 대한 orthonormal basis이다. 또한 위의 MRA를 정의하기 위한 몇 가지 규칙이 있는데 다음과 같다.

$\phi(t)$ 함수에 대한 적분은 1이다.

$$\int_{-\infty}^{+\infty} \phi(t) dt = 1 \tag{9}$$

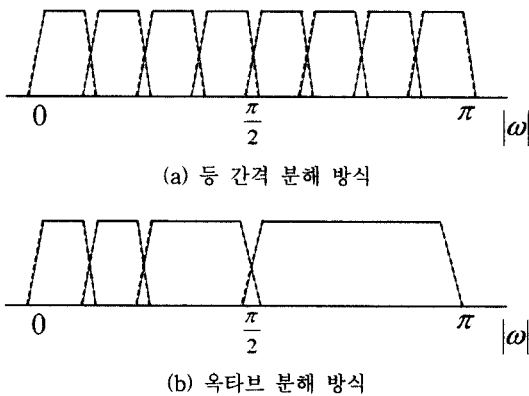
$\phi(t)$ 함수가 지니고 있는 에너지는 1이다.

$$\int_{-\infty}^{+\infty} |\phi(t)|^2 dt = 1 \tag{10}$$

$\{2^{m/2} \phi(2^m t - n) \mid n \in Z\}$ 는 V_{-m} 에 대한 basis이다.

2.3 영상 서브밴드 분할

웨이브릿이나 서브밴드 부호화는 기본적으로 원래의 영상에서 화소 값을 필터링하고 다운 샘플링하는 것이다. (그림 1)은 디지털 영상 데이터의 대역과 웨이브릿 변환에 의해 분해될 수 있는 방식의 종류를 나열하였다[3].

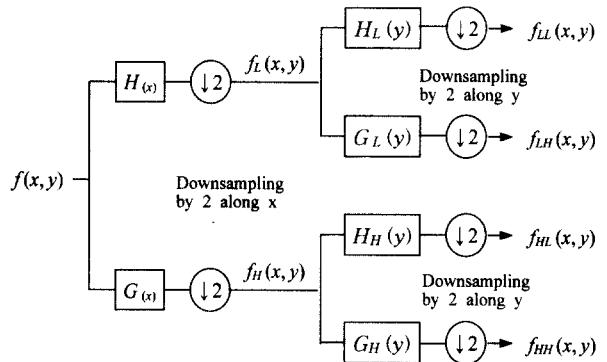


(그림 1) 웨이브릿 변환을 이용한 영상 분해 방식의 종류

디지털 데이터는 $[-\pi, \pi]$ 영역 안에 존재하게 되는데 (a)와 같이 신호를 등 간격으로 분해하여 서브밴드를 만들 수

있다. 그리고 (b)와 같이 저대역 성분을 계속해서 나누어 분해할 수도 있다. (a)와 같은 방법으로 분해하는 웨이브릿 변환 방식을 등 간격 분해 방식이라 하며 (b)와 같은 방법으로 신호를 분해하는 방법을 옥타브 분해 방식이라 한다.

영상은 2차원이기 때문에 필터링 또한 수직과 수평의 두 방향으로 적용한다. 이때 적용하는 방법은 Mallat[4]이 제안한 방법을 주로 사용한다. (그림 2)는 Mallat의 영상 분해 방법을 나타낸다.



(그림 2) 웨이브릿 변환에 의한 영상의 분해

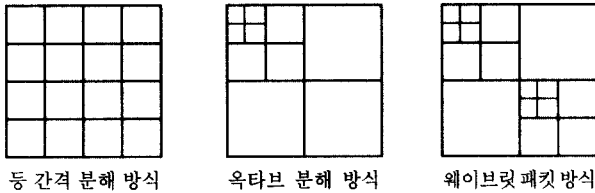
2차원 영상 $f(x, y)$ 를 x 방향으로 저주파 대역 필터링 과정인 $H(x)$ 를 한 후 다운 샘플링을 하여 저주파 성분인 $f_L(x, y)$ 와 고주파 대역 필터링 과정인 $G(x)$ 를 한 후 다운 샘플링을 하여 고주파 성분인 $f_H(x, y)$ 로 분해할 수 있다. 또한 $f_L(x, y)$ 와 $f_H(x, y)$ 를 y 방향으로 각각 저주파 대역 필터링과 고주파 대역 필터링인 $H_L(y), G_L(y), H_H(y), G_H(y)$ 을 수행한 후 다운 샘플링 하여 $f_{LL}(x, y), f_{HL}(x, y), f_{LH}(x, y), f_{HH}(x, y)$ 로 표현되는 4성분의 대역으로 세분화된다[3]. $f_{LL}(x, y)$ 는 시각에 민감하고 영상의 대부분의 정보를 가지고 있는 저주파 성분을 나타내며 $f_{HL}(x, y), f_{LH}(x, y), f_{HH}(x, y)$ 는 각각 수직성분, 수평성분, 대각선 성분이 지배적인 대역을 나타낸다. 그리고 $f_{HL}(x, y)$ 로 표현되는 최저주파 대역을 연속적으로 분해하는 과정을 통해 웨이브릿 분해 영상을 획득할 수 있다.

LL	HL
LH	HH

(그림 3) 웨이브릿 변환된 4 성분의 서브밴드

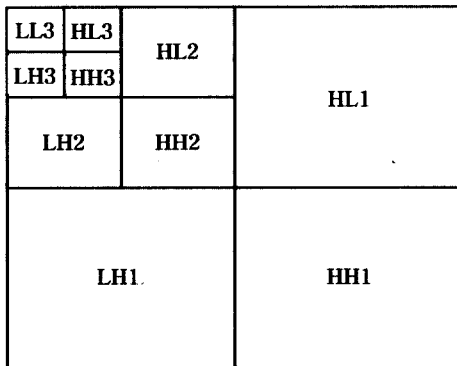
2차원의 이산 웨이브릿 변환을 수행한 후의 원 영상은 (그림 3)에서와 같이 LL, HL, LH, HH의 4가지 성분을 갖는 서

브밴드로 분해된다. L의 대역폭은 $0 < |\omega| < \pi/2$ 이며 H는 $\pi/2 < |\omega| < \pi$ 이다. 2차원 영상에 있어서 분해되어질 수 있는 방법은 아래의 (그림 4)와 같다.



(그림 4) 2차원 영상의 DWT 분해 방식

본 연구에서 사용한 웨이브릿 분해 방법은 (그림 4)의 옥타브 분해 방식이다. 즉, 원 영상에서 저대역의 부분을 반복하여 분해해 나가는 방법이다. (그림 5)는 원래의 영상을 분해 레벨 3으로 옥타브 분해 방식을 적용하여 입력 영상을 분해한 결과이다.



(그림 5) 웨이브릿 옥타브 분해 (level : 3)

3. Threshold-adjusted EZW 부호화

3.1 EZW 부호화

EZW(Embedded Zerotree Wavelet)부호화 방법은 DWT (Discrete Wavelet Transform)을 수행한 영상이 갖는 계수적 특성을 효율적으로 이용하도록 특별하게 고안되어진 영상 부호화 방법이다. EZW 부호화 방법은 영상을 비트스트림으로 압축하기 위하여 점진적인 부호화 방법에 그 기반을 두고 있다. 즉, 많은 양의 비트가 존재하는 스트림으로부터 복원된 영상은 적은 양의 비트가 존재하는 스트림으로부터 복원된 영상보다 더 선명한 화질을 갖게 된다. 그리고 이러한 복원 과정은 어느 시점에서든지 멈출 수 있으며 비트스트림이 가지고 있는 비트의 양에 합당한 영상의 품질을 보장할 수 있도록 한다. 이러한 부호화와 복호화 과정을 가리켜 점진적인 부호화 (progressive coding)라 하며, 임베디드 부호화(embedded coding)라고도 한다.

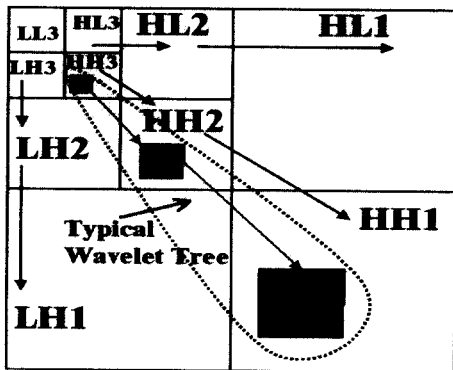
EZW 부호화 방법은 중요한 두 가지 관점에 그 기반을 두고 있다. 첫째로 자연 영상은 저주파수 대역 스펙트럼에 존재한다. 그리고 영상을 웨이브릿 변환 수행하여 대역을 분할하면 서브밴드 내의 에너지는 스케일의 감소에 비례하여 감소한다. 그래서 평균적으로 웨이브릿 계수들은 저주파 서브밴드에서 보다 고주파 서브밴드에서 더 작은 값을 갖게 된다. EZW 부호화 방법에서는 이러한 웨이브릿 변환된 계수가 갖는 특징을 효율적으로 이용하여 영상을 압축하기 위한 방법으로서 점진적인 부호화 방법을 사용하였다. 즉, 영상에서 중요 정보를 포함하는 저주파의 서브밴드를 우선적으로 전송하고, 상대적으로 덜 중요한 정보를 포함하는 고주파의 대역은 보다 더 선명한 영상을 얻기 원할 경우 추가적인 전송을 통하여 그 비트스트림에 비례하는 선명한 영상을 복원할 수 있다. EZW 부호화의 기반이 되는 두 번째 중요한 관점은 값이 큰 웨이브릿 계수는 작은 값을 갖는 웨이브릿 계수의 값보다 영상 내에서 더 중요한 의미를 지닌다는 점이다. 즉, 동일한 서브밴드 내에서 그 계수가 갖는 값의 크기 순으로 배열하여 부호화를 수행한다. EZW 부호화 방법은 이러한 두 가지 중요한 관점에 기반을 두고 일련의 단계를 거치면서 웨이브릿 계수를 내림차순으로 각 계수를 부호화 한다. 각 계수에 대한 부호화 과정은 threshold와의 비교를 통하여 수행하는데, 매 단계에서 사용되는 이러한 threshold는 웨이브릿 변환된 영상 내의 모든 계수를 통하여 구해진다. 계수의 부호화 과정에서는 계수와 threshold를 비교하여 만약 계수가 threshold보다 크다면 부호화를 수행하고 그 계수 값은 입력 계수 매트릭스로부터 제거된다. 그러나, 만약 계수가 threshold에 비해 작은 값을 갖는다면 그 계수 값을 부호화 하지 않고 다음 단계로 넘겨 현재의 단계보다 감소된 threshold를 적용하여 부호화를 수행한다. 이러한 부호화 과정은 웨이브릿 변환된 영상내의 모든 계수를 부호화 할 때까지 혹은, 정해진 최대 비트율 등의 조건을 만족할 때까지 반복하여 수행한다. 이러한 부호화 방법은 또한 계수를 threshold와 비교하여 그보다 작은 값을 갖는 계수를 포함하는 픽셀일 경우, 이를 효율적으로 부호화 하기 위해서 웨이브릿 변환된 영상의 스케일간 존재하는 계수들간의 종속성을 이용한다[6].

웨이브릿 변환은 신호를 시간 영역으로부터 시간-스케일 영역으로 변환한다. 즉, 이것은 웨이브릿 계수들은 2차원의 정보라는 것을 의미한다. 만약 웨이브릿 변환된 신호를 부호화 하고자 한다면 계수 값은 물론 그 계수가 존재하는 위치 값까지 부호화 해야한다. 앞서서도 설명했지만, 웨이브릿 변환을 이용한 원 영상의 분해 과정은 필터링에 이은 다운 샘플링이다. 따라서 각 서브밴드에 있는 영상 계수들은 수평, 수직, 대각선 방향으로 서로 상관성을 지니게 된다. 또한 같은 방향에 놓여 있는 서브밴드들 즉, HL3, HL2, HL1과 LH3,

LH2, LH1 그리고 HH3, HH2, HH1은 각각 서로 위치적인 연관성을 가지고 있다. EZW 부호화 방법은 이렇게 같은 방향에 존재하는 서브밴드간 상관성에 의존하여 제로트리를 구성함으로써 웨이브릿 변환된 계수들이 존재하는 대역과 방향성에 따른 중요도 지도(significance map)를 구성한다. 그리고 대역간 계수들의 중요도 여부에 따라 제로트리 부호화를 수행한다. (그림 6)은 quad-tree 구조로서 서로 다른 서브밴드에서 웨이브릿 계수들간의 상관 관계를 나타낸다.

(그림 6)에서와 같이 저주파의 서브밴드 내에 존재하는 계수들은 각각 다음의 고주파의 서브밴드 내에 4개씩의 자손을 가지고 있다. 그리고 이러한 4개의 자손들도 각각 그 다음의 고주파의 서브밴드 내에 4개씩의 자손들을 가지고 있다.

제로트리는 루트와 같거나 혹은 그보다 작은 모든 노드의 quad-tree 구조를 말한다. 이러한 구조의 트리는 각각 단독의 심벌로 부호화되며 제로 값이 채워진 quad-tree 형태로 다시 복원된다. 즉, EZW 부호화는 스케일의 감소에 따른 웨이브릿 계수의 값이 감소하는 현상에 기반을 둔 제로트리를 이용한 부호화 방법이다.



(그림 6) 대역간 웨이브릿 계수의 상관 관계

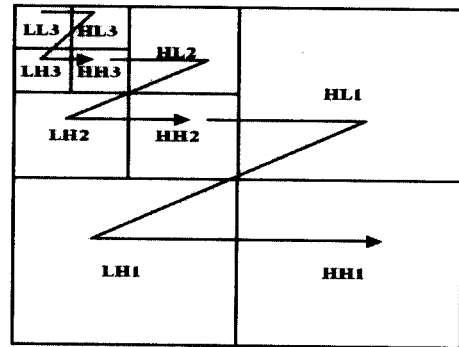
Quad-tree 구조에서 루트에 존재하는 계수의 값이 특정 threshold보다 작다면 quad-tree 구조를 구성하고 있는 모든 계수 값들은 그 threshold보다 작을 가능성이 매우 높다고 가정을 한다. 물론 제로트리 구조에 존재하는 이러한 가능성은 절대적인 것은 아니다. 하지만 일반적으로 그러한 가능성이 높다는 것이다.

EZW 부호화는 각 단계에서는 threshold를 이용하여 부호화를 수행하는데, 웨이브릿 계수의 값이 threshold보다 크지 아니면 작은지에 따라 부호화를 수행하고 비교된 threshold 값을 복호화 과정에 전송하여 모든 계수 값들을 완벽하게 복원할 수 있도록 한다. 그리고 완벽하게 복원하기 위해서 전송하고자 하는 가장 작은 계수 값보다도 더 작은 threshold가 될 때까지 threshold를 감소시키면서 부호화 과정을 반복해 수행한다. 이러한 과정을 거쳐서 생성된 비트스트림을 이용한다면 원하는 만큼의 정밀도로도 복원이 가능하고 완벽한

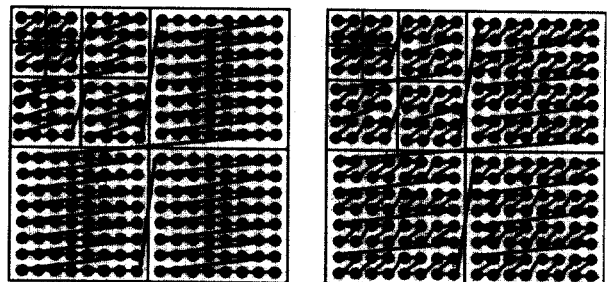
복원도 가능하다. 그리고 이러한 일련의 과정에서 부호화와 복호화 과정 양쪽에 미리 정의된 threshold 값을 사용한다면 부호화 과정에서 적용된 threshold 값을 복원 과정에 전송할 필요도 없다.

EZW 부호화 과정중 계수의 위치에 대한 정보를 부호화하지 않으면 복호화 과정에 전송되어진 비트스트림을 완전하게 복원할 수 없다. 이와 같은 위치 정보에 대한 부호화는 효율적인 부호화와 비효율적인 부호화를 구분하는 중요한 요소이다. EZW 부호화는 웨이브릿 변환된 영상의 계수를 스캔하여 그 계수 값을 부호화 하는데, 일반적으로 웨이브릿 변환을 사용하여 분할되어진 저주파 대역으로부터 고주파 대역으로 Z자 모양으로 각 서브밴드를 스캔한다. 즉, LL → HL → LH → HH의 순서로 스캔한다. (그림 7)는 웨이브릿 변환된 영상의 서브밴드를 스캔하는 순서를 나타낸다.

또한 각 서브밴드 내에서 각 계수를 순회하는 방법으로는 여러 가지 방법이 있지만 일반적으로 래스터 스캔(Raster scan)과 모튼 스캔(Morton scan)의 두 가지 방법을 많이 사용한다. 참고문헌 [6]에서는 래스터 스캔 방법을 사용하였으며 본 논문에서는 모튼 스캔 방법을 사용하여 각 계수를 부호화 하였다. 계수를 스캔하는 방법에 따른 부호화 결과는 최종적으로 압축된 비트스트림의 내용에 영향을 주며, 성능 면에서는 그다지 큰 차이는 없다. (그림 8)는 각 계수를 스캔하는 방법의 종류를 나타낸다.



(그림 7) 웨이브릿 변환 영상의 서브밴드 스캔



(a) Raster scan order

(b) Morton scan order

(그림 8) 웨이브릿 변환된 계수의 스캔 방법

EZW 부호화는 웨이브릿 변환되어 N-레벨로 분할된 영상을 입력 데이터로 사용하며, 부호화를 통하여 생성되는 비트 스트림은 복호화 과정에서 사용하기 위한 약간의 정보를 필요로 한다. 이 때 필요한 최소의 정보로는 부호화 과정에서 사용된 웨이브릿 변환의 종류와 웨이브릿 변환된 레벨 수준 및 초기 과정에서 사용된 최초의 threshold 값 등이 있다.

EZW 부호화의 첫 번째 과정은 초기화 과정으로서 분할된 영상의 계수 값을 모두 스캔하여 초기 threshold 값을 설정하는 과정이다. 초기의 threshold 값은 식 (11)을 이용하여 결정한다[6].

$$init\ Thr = 2^{\lceil \log_2(MAX(|\gamma(x,y)|)) \rceil} \quad (11)$$

식 (11)에서 $MAX(|\gamma(x,y)|)$ 는 분할된 영상의 전체 계수 중 가장 큰 계수 값을 의미하며 $\gamma(x,y)$ 는 계수 값을 의미한다. 이렇게 결정된 초기의 threshold 값은 부호화 과정 전체에 영향을 준다.

부호화 과정에서 사용할 초기의 threshold 값을 결정한 후 본격적으로 실제 계수를 처리하는 주 처리 과정을 수행하는데 부호화의 주 처리 과정은 크게 dominant 단계와 subordinate 단계인 2단계로 구성되어 있다. Dominant 단계는 영상 내의 계수 값의 위치 정보를 부호화 하는 과정이며 subordinate 단계는 계수의 크기를 부호화 하는 과정이다. (그림 9)은 EZW 부호화의 주요 부분을 의사 코드(pseudo code)를 사용하여 나타낸 것이다.

```

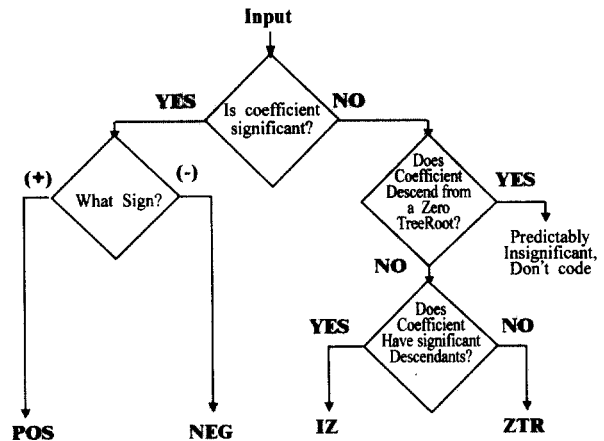
threshold = initial_threshold;
do {
    dominant_pass(image);
    subordinate_pass(image);
    threshold = threshold/2;
} while (threshold > minimum_threshold);
    
```

(그림 9) EZW 부호화 의사 코드

Dominant 단계에서는 입력된 분할 영상 내에서 지배적인 값(significance value)을 갖는 계수에 대해 부호화를 수행하는데 POS(POSitive), NEG(NEGative), IZ(Isolated Zero), ZTR(Zero TreeRoot)의 4 심벌을 사용하여 계수의 위치 값을 부호화 한다. 여기에서 지배적인 값을 갖는 계수라 함은 계수의 절대값이 threshold와 비교하였을 때 큰 값을 갖는 계수를 가리킨다. POS 심벌로서 부호화 되는 경우는 threshold와 비교되는 계수의 절대값이 threshold보다 크기가 크며 그 계수가 지닌 부호가 양수일 때이다. 반면에 POS의 심벌이 부여되는 상황과 동일하게 계수의 절대값은 threshold보다 크지만 그 부호가 음수일 경우에는 NEG의 심벌이 사용된다. POS와 NEG는 지배적인 값으로 분류되며 현재 계수 값과 비교된 threshold 값의 중간 값을 재 생성된 계수로서 저

장한다. IZ의 심벌로서 부호화 되는 경우는 threshold와 비교되는 계수의 절대값이 threshold보다 값의 크기가 작아서 현재의 계수가 영상에서 차지하는 정보는 지배적인 값은 아니지만 동일한 방향성을 갖는 다음의 서브밴드에 지배적인 값 즉, POS나 NEG로 부호화 될 수 있는 계수가 손자로서 존재하는 경우이다. 마지막으로 ZTR의 심벌은 threshold와 비교되는 계수가 지닌 절대값의 크기가 threshold보다 작아서 현재의 계수가 영상에서 차지하는 정보는 지배적인 값이 아님은 물론 동일한 방향성을 갖는 다음의 서브밴드에도 지배적인 값을 갖는 계수가 손자로서 발견되지 않을 경우에 부여된다. IZ나 ZTR로 부호화 된 계수는 0값을 재생성하며 이를 저장한다.

(그림 10)은 dominant 단계에서 입력된 계수 값이 threshold값과 비교하여 각각의 정의된 심벌로 부호화 되는 조건을 흐름도로 표현하였다[6]. 그리고 (그림 11)은 dominant 단계의 전체 알고리즘을 의사 코드로서 나타낸 것이다.



(그림 10) Dominant 단계에서 심벌이 부호화 되는 과정의 흐름도

```

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

(그림 11) Dominant 단계의 의사 코드

Subordinate 단계에서는 입력 영상 내에서 지배적인 값을 지니고 있는 계수로 결정되어 dominant 단계에서 POS나

NEG 심벌이 부여되었으며 0이 아닌 값을 재생성한 계수들에 대해 부호화를 수행한다.

Dominant 단계에서 POS나 NEG 심벌로 부호화 된 계수들에 한하여 upper 혹은 lower로 구역을 나누어 1 혹은 0의 bit로 부호화 하는 정제(refinement) 과정을 수행한다. 구역은 threshold값, dominant 단계에서 재생성 된 값 그리고 영상내의 최대 계수 값을 이용하여 두 부분으로 분할한다. 분할된 두 구역은 실제 계수가 지닌 절대값이 어느 구역에 속하는지를 구분하는 기준이 되며 실제 계수의 절대값이 upper 구역에 속한다면 1을 부호화하고 lower 구역에 속한다면 0을 부호화 한다. (그림 12)는 subordinate 단계의 전체 알고리즘을 의사 코드로 나타낸 것이다.

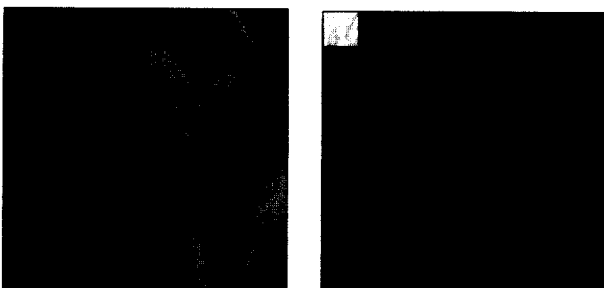
```

subordinate_threshold = current_threshold/2;
for all elements on subordinate list do
{
    if coefficient > subordinate_threshold then
    {
        output a one;
        coefficient = coefficient - subordinate_threshold;
    }
    else output a zero;
}
    
```

(그림 12) Subordinate 단계의 의사 코드

3.2 Threshold-adjusted EZW

웨이브릿 변환된 입력 영상의 계수를 이용하는 threshold-adjusted EZW 부호화 시스템은 먼저 입력 영상을 이산 웨이브릿을 수행하여 N-레벨로 분해된 출력 영상을 얻어 그 분할된 영상을 입력 데이터로 부호화를 수행한다. (그림 13)은 이산 웨이브릿 변환을 수행한 후의 결과 영상으로서 3-레벨로 분해한 Lena 512*512 영상이다.



(a) 원 영상 (b) 웨이브릿 분해 영상

(그림 13) 3-레벨로 분해한 Lena 512*512 영상

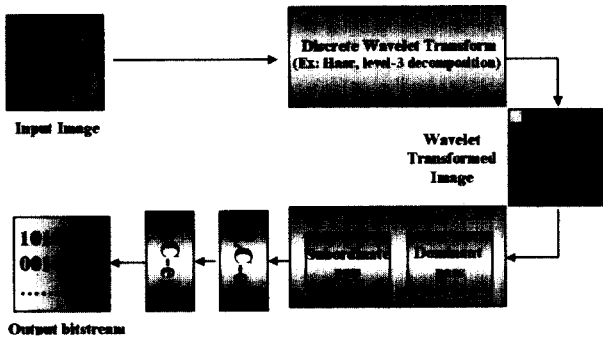
Threshold-adjusted EZW 부호화 시스템은 (그림 13)과 같이 원래의 영상을 N-레벨로 분해함으로써 부호화를 수행하기 위한 전처리 과정을 마친다. Threshold-adjusted EZW

의 부호화 과정은 EZW 부호화 과정에 threshold 조정과 웨이블릿 계수 threshold 개념을 적용한 것이다. 여기서 설명을 분명하게 하기 위하여 `ezw_Threshold`와 `coeff_Threshold`라는 두 개의 threshold를 설정하기로 하자. 즉, threshold-adjusted EZW는 EZW 부호화 과정의 초기에 설정되는 threshold 즉, `ezw_Threshold`로부터 웨이블릿 계수 조정에 필요한 `coeff_Threshold`를 얻고 이값을 이용하여 웨이블릿 계수 값들을 0 값으로 조정함으로써 구현되었다. EZW 부호화 과정에서 첫 번째 dominant 단계를 수행하기 전에 초기 threshold `ezw_Threshold`를 구한다. 그리고 이 `ezw_Threshold` 값에 0.01에서 0.09까지의 값을 곱하여 웨이블릿 계수 조정에 필요한 `coeff_Threshold` 값을 구한다. 다음에 웨이블릿 계수들을 검사하여 계수 값이 `coeff_Threshold` 값보다 낮으면 웨이블릿 계수 값들을 0으로 조정한다. 이 과정을 거치면 초기 threshold `ezw_Threshold`의 1%에서 9% 이하의 웨이블릿 계수 값들은 0이 되는 것이다. 그런 다음 원래의 `ezw_Threshold` 값을 가지고 EZW 부호화 과정을 시작한다. 이러한 방법의 적용 결과로 웨이블릿 계수 속에 0 값이 많이 만들어지게 되며 결과적으로 제로트리가 많이 생성되게 된다. 제로트리가 많이 생성되면 EZW 부호화 과정에서 생성되는 symbol 중에 ZTR (ZeroTreeRoot)가 많이 생성되어 출력되는 비트스트림에 0 값이 많이 생성되어 압축률을 더 높게 된다. 또한 EZW 부호화시에 dominant 단계에서 부호화하여야 하는 웨이블릿 계수들의 수를 줄이게 되어 부호화에 소요되는 시간을 줄이게 되는 것이다. 원래의 EZW 부호화 알고리즘은 dominant와 subordinate의 2 단계를 수행하면서 모든 계수에 대하여 `ezw_Threshold`를 적용하고 부호화한다. 그러나 이로 인해 값이 그다지 크지 않은 웨이블릿 계수들에 대하여도 2 단계의 과정을 적용하게 된다. 이것은 시간적 과부하를 주어 결과적으로 부호화 과정 전체의 처리 속도를 저하시키고 많은 양의 비트스트림을 출력하게 한다. 이러한 현상을 개선하고 복원 영상에 있어서 적절한 품질을 유지하는 범위내에서 원래의 threshold인 `ezw_Threshold`에 0.01에서 0.09까지의 값을 곱하고 이 값 이하의 웨이블릿 계수 값들을 0으로 만들어 줌으로써 제로트리를 많이 만들게 되어 EZW 부호화에 소요되는 시간을 단축하고 또한 생성되는 비트스트림속에 0을 많이 만들게 됨으로서 압축율을 높이는 것이다. 이러한 과정을 통하여 전체 부호화에 소요되는 처리 시간을 매우 많이 단축시킬 수 있다. 그러나 실험에서 0.01부터 0.04까지의 값은 유용한 값으로 확인하였으나 그 이상의 값은 복원 영상에 많은 손상을 주는 것을 확인하였다. 그리고 또한 EZW 부호화를 수행하여 생성된 비트스트림의 압축률을 높이기 위해 run-length 부호화와 산술 부호화를 사용하였다. 특히 run-length 부호화는 산술 부호화의 효율을 극대화하기 위하여 사용하였는데

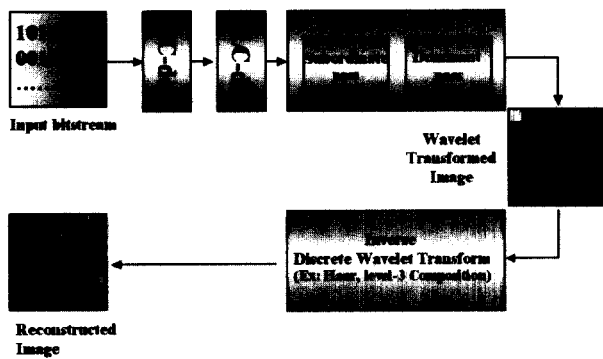
반복되는 0 값들을 그룹화하고 그 결과를 이용하여 다시 산술 부호화를 수행하여 비트스트림을 최대 압축하였다. (그림 14)는 threshold-adjusted EZW 부호화 시스템의 구조를 나타낸다. 그림에 나타나 있듯이 부호화 시스템은 웨이블릿 변환, EZW 부호기, run-length 부호기, 그리고 산술부호기로 구성된다. 그리고 threshold-adjusted EZW의 복호화 과정은 부호화의 역 과정으로 구성된다. (그림 15)는 threshold-adjusted EZW의 복호화 시스템 구조를 나타낸다.

4. 실험 결과와 성능 분석

Threshold-adjusted EZW의 성능에 대한 모의 실험을 하기 위한 테스트 영상은 그레이스케일의 512*512 크기를 갖는 Lena, Barbara, Goldenhill 영상을 사용하였다. 그리고 웨이블릿 변환은 Daubechies 웨이블릿을 사용하였으며 분해 레벨은 3으로 하였다. 그리고 부호화 이전의 원 영상과 복호화를 통하여 생성된 복원 영상간의 영상 품질을 비교하기 위하여 PSNR(Peak Signal-to-Noise Ratio)의 수치를 제시하였다.



(그림 14) Threshold-adjusted EZW 부호화 시스템 구조



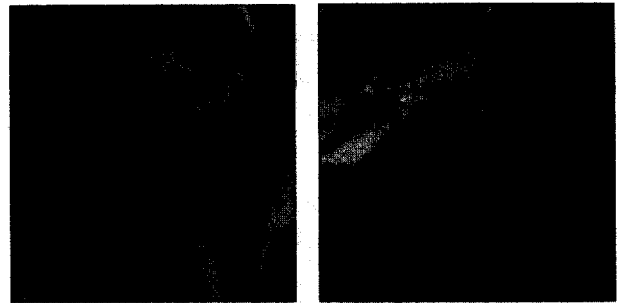
(그림 15) Threshold-adjusted EZW 복호화 시스템 구조

식 (12)는 모의 실험에서 사용한 PSNR에 대한 공식이다. MSE는 원 영상과 복원 영상간의 평균 제곱-오차를 나타낸다.

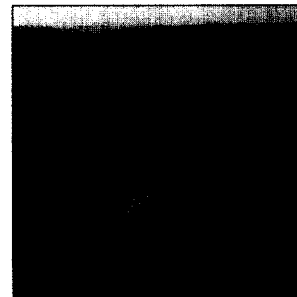
$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) dB \quad (12)$$

(그림 16)은 모의 실험에 사용된 Lena 512*512, Barbara 512*512, Goldenhill 512*512의 영상을 나타낸다.

(그림 17)은 (그림 16)의 3개 실험 영상의 부호화 이전 파일 크기와 EZW 부호화 과정을 거쳐 생성된 파일 크기 및 threshold-adjusted EZW 부호화 과정을 거쳐 생성된 파일들의 크기를 비교한 그래프이다. 결과적으로 (그림 17)의 그래프에서 보는 바와 같이 최대 약 50 : 1의 압축률을 얻을 수 있었다.

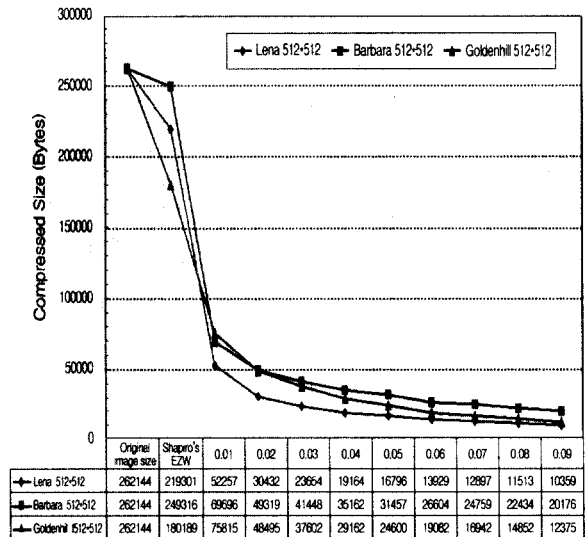


(a) Lena 512*512 원 영상 (b) Barbara 512*512 원 영상



(c) Goldenhill 512*512 원 영상

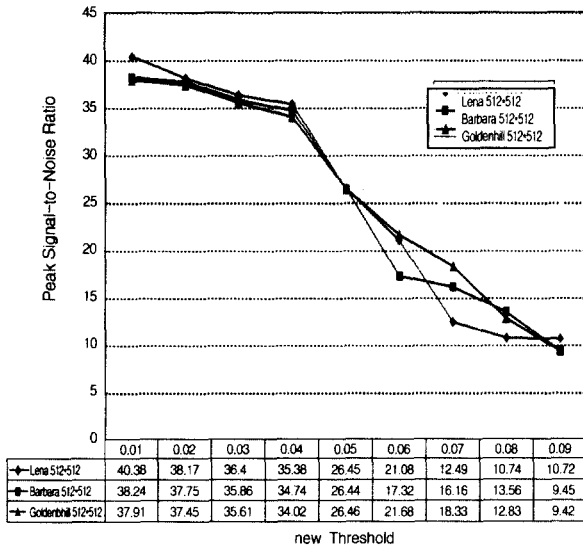
(그림 16) 실험에 사용된 원 영상



Original Image Size, Shapiro's EZW and Proposed EZW

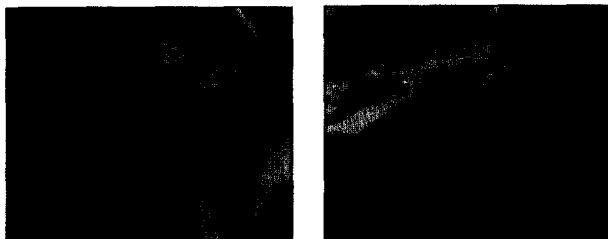
(그림 17) 원 영상과 압축된 영상의 파일 크기 비교

그리고 (그림 18)은 영상의 품질을 비교하기 위하여 부호화의 입력으로 사용된 원 영상과 복호화 과정을 통하여 복원된 영상의 PSNR을 산출한 결과이다.



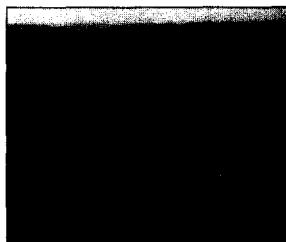
(그림 18) 원 영상과 복원 영상의 PSNR 비교

아래의 (그림 19)는 threshold-adjusted EZW를 사용하여 원 영상에서 1%에 해당하는 계수 값을 손실시키고 부호화하고 복호화 과정을 거쳐 복원시킨 영상이다.



(a) 복원된 Lena 512*512 (PSNR : 40.34dB)

(b) 복원된 Barbara 512*512 (PSNR : 38.25 dB)



(c) 복원된 Goldenhill 512*512 (PSNR : 37.92 dB)

(그림 19) 복원한 영상

5. 결 론

본 논문에서는 EZW 알고리즘의 성능을 개선한 threshold-adjusted EZW 부호화와 복호화 시스템의 구현에 대하여 소

개하였다.

Threshold-adjusted EZW 부호화는 EZW 부호화 과정의 초기에 설정되는 threshold 값을 기반으로 하여 웨이블릿 계수를 조절하여 EZW 부호화 과정에 소요되는 시간을 대폭 단축시켰으며 또한 압축율을 향상시켰다. 전체 시스템으로는 run-length 부호화와 산술 부호화를 적용하여 비트스트림의 압축률을 높였다. 개발된 시스템은 원래의 EZW 알고리즘에 비교하여 속도와 압축률 면에서 많은 향상을 나타내었다.

개발된 시스템의 출력 영상은 원 영상에 비하여 손색이 없었다.

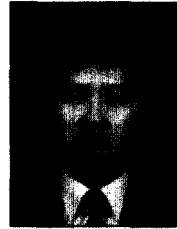
참 고 문 헌

- [1] Raughveer M. Rao and Ajit S. Bopardikar, Wavelets Transforms-Introduction to Theory and Applications, Addison-Wesley, 1998.
- [2] C. Sidney Burrus, Ramesh A. Gopinath and Haitao Guo, Introduction to Wavelets and Wavelet Transforms : A Primer, Prentice-Hall, 1998.
- [3] Martin Vetterli and Jelena Kovacevic, Wavelets and Sub-band Coding, Prentice-Hall, 1995.
- [4] Stephan G. Mallat, "A Theory for Multi-resolution Signal Decomposition : The Wavelet Representation," IEEE Trans. Pattern Anal. Machine Intell., Vol.11, pp.674-693, July, 1989.
- [5] O. Rioul and Martin Vetterli, "Wavelets and Signal Processing," IEEE Signal Processing Magazine, Vol.8, No.4, pp. 14-38, October, 1991.
- [6] Jerome M. Shapiro, "Embedded Image Coding Using Zero-tree Of Wavelet Coefficients," IEEE Trans. On Signal Processing, Vol.41, No.12, pp.3445-3462, December, 1993.
- [7] A. Moffat, R. Neal and I. H. Witten, "Arithmetic Coding Revisited," Proc. IEEE Data Compression Conference, March, 1995.
- [8] Harley R. Myler and Arthur R. Weeks, Image Processing Algorithms In C, Prentice-Hall.
- [9] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Addison-Wesley, 1992.
- [10] Gregory A. Baxes, Digital Image Processing : Principles and Application, John Wiley&Sons, 1994.
- [11] N. Ahmed, T. Natrajan and K. R. Rao, "Discrete Cosine Transform," IEEE Trans. On Computers, Vol.24, pp.88-93, 1974.
- [12] A. Said, W. Pearlman, "A New, Fast and Efficient Image Codec based on Set Partitioning in Hierarchical Trees," IEEE Trans. CSVT, Vol.6, No.3, pp.243-250, June, 1996.
- [13] D. Taubman, "High Performance Scalable Image Compression with EBCOT," IEEE Trans. on Image Processing, Vol.9, No.7, pp.1158-1170, July, 2000.
- [14] Z. Xiong, K. Ramchandran, and M. Orchard, "Space-Frequency Quantization for Wavelet Image Coding," IEEE Trans. on Image Processing, 1997.



채 희 중

e-mail : hjchae77@hotmail.com
1992~1999년 호서대학교 컴퓨터공학부
학사 졸업
1999~2001년 호서대학교 컴퓨터공학부
석사 졸업
관심분야 : 영상압축, ARS, 프로그래밍 언어



이 호 석

hslee@office.hoseo.ac.kr
1979~1983년 서울대학교 전자계산기공학과
학사 졸업
1983~1985년 서울대학교 컴퓨터공학과
석사 졸업
1985~1989년 한국전기통신공사 근무
1989~1993년 서울대학교 컴퓨터공학과 박사 졸업
1994~현재 호서대학교 컴퓨터공학부 근무
관심분야 : 영상압축, 영상부호화, 영상처리, 웨이블릿 응용, 프
로그래밍 언어