

DWT 계수의 트리구조를 이용한 네트워크-적응적 JPEG2000 컨텍스트 추출방법

정희원 최 현 준*, 서 영 호**, 종신회원 김 동 욱*

A Network-adaptive Context Extraction Method for JPEG2000 Using Tree-Structure of Coefficients from DWT

Hyun-Jun Choi*, Young-Ho Seo** *Regular Members*, Dong-Wook Kim* *Lifelong Members*

요 약

본 논문에서는 JPEG2000의 EBCOT에서 과도한 연산량을 요구하는 컨텍스트 추출과정의 연산량을 감소시키는 방법을 제안하였다. 이 방법은 웨이블릿 변환의 특성인 계수들의 트리구조와 그 계수들의 상관도를 이용하여 특정 임계값을 설정하고 그 임계값보다 작은 계수와 그 자손계수들은 컨텍스트 추출과정을 거치지 않게 하는 것이다. 이 임계값이 증가함에 따라 컨텍스트 추출을 위한 연산량과 출력 데이터양이 줄어드나 화질의 열화가 심해지는 연산량과 화질 또는 데이터량간의 상보적 관계가 성립된다. 따라서 이 임계값을 네트워크의 환경 또는 조건에 따라 설정하면 네트워크에 적응적으로 수행할 수 있는 컨텍스트 추출방법이 가능하다. 이 방법을 실험한 결과 수용할 만한 화질의 범위(30dB 이상)의 임계값은 0에서 4사이이었으며, 이 범위에서 연산량은 평균 3%에서 64%를 감소할 수 있고, 출력데이터는 평균 32%에서 73%의 감소율을 보여 수용할 만한 화질의 열화를 대가로 상당한 연산량 감소와 데이터량 감소를 얻을 수 있을 것으로 판단된다. 따라서 제안한 방법은 무선 환경에서 영상/비디오 데이터의 실시간 통신 등에 매우 유용하게 사용될 것으로 기대된다.

Key Words : JPEG2000, EBCOT, DWT, Network Adaptive, Context Extraction.

ABSTRACT

In EBCOT, the context extraction process takes excessive calculation time and this paper proposed a method to reduce this calculation time. That is, if a coefficient is less than a pre-defined threshold value the coefficient and its descendents skip the context extraction process. There is a trade-off relationship between the calculation time and the image quality or the amount of output data such that as this threshold value increases, the calculation time and the amount of output data decreases, but the image degradation increases. Therefore, by deciding this threshold value according to the network environments or conditions, it is possible to establish a network-adaptive context extraction method. The experimental results showed that the range of the threshold values for acceptable image quality(better than 30dB) is from 0 to 4. The experimental results showed that in this range the resulting reduction rate in calculation time was from 3% to 64% in average, the reduction rate in output data was from 32% to 73% in average, which means that large reduction in calculation time and output data can be obtained with a cost of an acceptable image quality degradation. Therefore, the proposed method is expected to be used efficiently in the application area such as the real-time image/video data communication in wireless environments, etc.

* 광운대학교 전자재료공학과 ({chj,dwkim}@kw.ac.kr), **한성대학교 정보통신공학과 (design@kw.ac.kr)

논문번호 : KICS2005-01-028, 접수일자 : 2005년 1월 13일

※ 본 논문은 교육인적자원부, 산업자원부, 노동부의 출연금 및 보조금으로 수행한 최우수실험실지원사업의 연구결과입니다.

I. 서론

세계적인 물결인 디지털화에 의해 2005년까지 컨텐츠의 60% 이상이 디지털화 될 것으로 예측되며, 유·무선 인터넷의 발달로 이들 컨텐츠들의 전달·보급이 매우 활발해지고 있다. 이들 컨텐츠 중 가장 사용빈도의 증가가 큰 것이 영상/비디오 컨텐츠인데, 이들은 매우 함축적인 정보를 포함하는 장점이고 있는 반면, 그 데이터양이 방대한 단점 또한 갖고 있다. 따라서 영상 데이터양을 줄이기 위한 노력이 지난 20여년간 진행되어오고 있으며, 그 중 정지영상에 대해서는 JPEG(Joint Photographic Expert Group)^[1], 동영상에 대해서는 MPEG(Moving Experts Group)^[2], 그리고 화상회의 등의 양방향 동영상 통신을 위해서는 H.26x^[3]가 국제 표준으로 채택된 바 있다.

이 중 JPEG은 구현이 쉬운 반면 저 비트율에서 뚜렷한 화질의 열화를 보이고, 잡음 환경에서 성능이 크게 저하된다는 단점을 가지고 있다. 이러한 단점을 보완하기 위해 2000년에 JPEG2000^[4]의 정지영상 표준이 채택되기에 이르렀다. JPEG2000은 DCT(Discrete Cosine Transform) 대신 DWT(Discrete Wavelet Transform)를 기반으로 하며, 자연영상, 과학영상, 의료영상, 원격탐지영상, 문서영상, 그래픽영상 등과 같이 서로 다른 특성을 갖는 다양한 형태의 정지영상에서 압축효율이 뛰어난 방식이다.

전체영상을 대상으로 하는 DWT 이 외에 JPEG2000은 EBCOT(Embedded Block Coding with Optimized Truncation)^[5-7]이라는 특별한 엔트로피 부호화 방법을 채택하고 있다. 이 방법 중 Tier 1에 속해 있는 컨텍스트(context) 추출 알고리즘은 주위 데이터들과의 상관도에 따라 부호화하는 방법으로 그 효율성이 매우 뛰어나나 전체 JPEG2000 연산의 50%를 차지할 만큼 많은 양의 연산이 필요하다^[8]. 정지영상을 압축하는 JPEG2000의 응용분야는 영상의 추출 후 사용까지 이러한 연산을 충분히 수행할 시간이 있는 분야도 있으나, 추출 즉시 압축하여 전송하여야 하는 등의 응용분야도 최근 들어 많이 고려되고 있어 과도한 연산량을 줄일 필요가 있다^[8-11].

본 논문에서는 네트워크 등의 환경에 따라 컨텍스트 추출을 적응적으로 수행하여 연산시간을 적응적으로 조절할 수 있는 컨텍스트 추출방법과 이와 관련된 Tier 1 전체 알고리즘을 제안한다. 이 방법은 연산시간과 화질의 상보적 관계를 사용하며, DWT

결과 생성되는 부대역들의 공간정보와 부대역간 상관관계를 이용한다^[12].

본 논문은 다음과 같이 구성된다. 2장에서는 JPEG2000 시스템과 EBCOT 알고리즘을 설명하고, 3장에서는 제안하는 Tier 1 방법에 대하여 설명한다. 4장에서는 이 방법에 대한 실험 및 그 결과를 분석하고, 이를 토대로 5장에서 결론을 맺는다.

II. JPEG2000 압축 시스템

본 장에서는 JPEG2000 정지영상 압축시스템과 이 시스템의 표준 엔트로피 부호화 방법인 EBCOT 알고리즘의 개요에 대해 설명한다. 이 중 컨텍스트 추출방법은 좀더 상세히 설명하며, 본 논문의 주제인 연산시간 감소를 위한 기존의 연구 등을 다룬다.

2.1 JPEG2000 영상 압축 시스템

JPEG2000 영상 압축 시스템은 간략히 그림 1과 같이 나타낼 수 있다. 다른 영상 압축 시스템과 같이 JPEG2000은 데이터의 주파수대역 변환(FDWT), 손실압축(Quantizer), 무손실 압축의 과정(EBCOT)을 거쳐 압축을 수행한다. 여기에 입력된 영상크기를 효율적인 처리를 위한 크기로 분할하는 타일링(Tiling)과정과 영상의 중요한 부분을 설정하여 그 부분을 부각시키는 ROI(Region of Interest) 처리가 추가되어 있다.

이 중에서 JPEG으로부터 가장 큰 변화는 DWT를 수행한다는 것과 엔트로피 코딩이 EBCOT로 바뀐 것이다. DWT는 DCT에 비해 고압축 시 블로킹 효과(blocking effect)가 발생하지 않으며, 변환 후의 부대역들이 원 영상의 위치정보를 그대로 보유함으로써 주파수 정보 뿐 아니라 위치정보를 데이터 압축과정에 사용할 수 있다는 장점을 갖고 있다. EBCOT의 가장 큰 목적은 그 이름에서도 알 수 있듯이 Tier 2에서 실제 전송될 데이터 패킷들을 구성할 때 네트워크 등의 조건에 따라 임의 또는 비트평면 단위로 데이터를 잘라내고(truncation) 구성할 수 있도록 하는 것이다. 이와 더불어 영상 데이

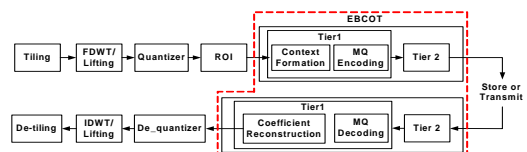


그림 1. JPEG2000 영상 압축 시스템의 블록도
Fig. 1. Block diagram of JPEG2000 image compression system

터가 갑자기 변화하는 것보다 점진적으로 변화하거나 동일 데이터가 연속되는 경우가 많다는 특성을 결합하여 주위의 데이터들과 연계하여 코딩하는 방법(컨텍스트 추출)이 엔트로피 코딩방법으로 채택되었다. 또한 이 결과를 효과적인 데이터 스트림으로 만드는 방법으로는 가변길이부호화 방식보다는 산술 부호화(MQ-코딩) 방식이 표준으로 채택되었다. 이 중 본 논문에서 연관된 컨텍스트 추출 부분을 다음 절에서 좀 더 상세히 설명한다.

2.2 컨텍스트 추출 알고리즘

웨이블릿 변환과 양자화 과정 및 ROI 처리를 거친 데이터는 EBCOT의 Tier 1 과정 중 컨텍스트 추출과정으로 입력된다. Tier 1 과정은 그림 2에 나타난 것과 같이 미리 정해진 코드블록단위(최대 64 X64)로 계수 값들을 분할하고 각 코드블록은 부호-크기 비트평면별로 처리된다. 이 중 가장 먼저 컨텍스트가 추출되고 그에 따라 미리 정해진 결정값(decision value)이 부가된다. 컨텍스트 추출의 데이터 단위는 종방향의 4비트이며(그림 2와 3의 음영 부분), 주위의 14비트를 참조데이터로 하여 총 18비트를 대상으로 한다(그림 3 참조). 데이터의 처리순서는 4비트 단위로 열방향(그림 2의 stripe)으로 진행하며 한 stripe의 처리가 끝나면 다음 stripe의 맨 좌측부터 진행한다. 경계면에 대해서는 필요한 데이터가 '0'이라 간주(zero padding)한다.

컨텍스트 추출은 현재 비트의 값, 현재위치의 상위 비트의 값, 그리고 주위 비트들의 값들을 참조하여 세 가지 종류로 처리되는데, 그림 4에 그 과정을 나타내었다. 이 과정에서 필요한 데이터들을 참조하기 위해 부가적인 비트평면들이 생성되는데, 이들은

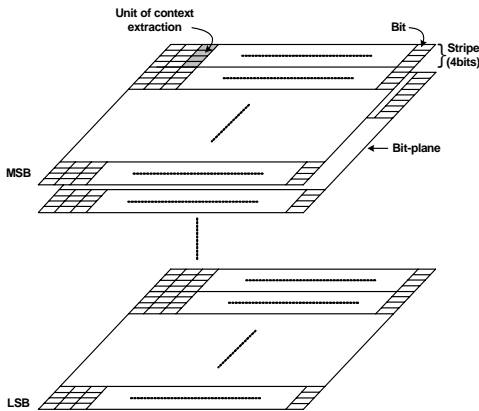


그림 2. 코드블록의 구조
Fig. 2. Structure of a code block

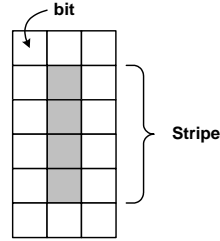


그림 3. 컨텍스트 추출을 위한 참조 데이터 범위
Fig. 3. Reference data range for context extraction

표 1. 코딩을 위한 부가 비트평면들
Table 1. Additional bit planes for coding

Category	Plane	Description
Bit_plane data	X	sign
	v	N-th magnitude
Coding state	σ	significance state
	π	delayed significance state
	τ	visited

표 1에 나타나있다. 부가 비트평면은 현재 비트평면의 데이터를 보유하고 있는 두 평면(현재 비트평면의 각 비트들의 부호를 보유하고 있는 sign bit-plane)과 그 때까지의 코딩상태를 보유하고 있는 세 평면(각 비트위치에서 현재까지 MSB가 나왔는지 여부를 알려주는 significant state bit-plane, magnitude refinement pass를 지났는지 여부를 알려주는 delayed significant state bit-plane, 현재 4비트의 단위 중 코딩처리된 비트들의 위치를 알려주는 visited bit-plane)으로 구성된다.

그림 4의 Significant Propagation Pass(SPP)는 부호-크기 비트평면으로 나뉜 계수 중에서 해당 위치의 MSB를 찾는 패스이며, Magnitude Refinement

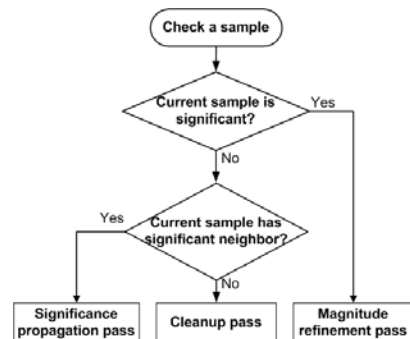


그림 4. 패스 결정 순서도
Fig. 4. Flowchart to determine pass

Pass(MRP)는 이미 중요 전달 패스에서 부호화 된 계수를 자세하게 부호화하는 패스이다. Clean-Up Pass(CUP)는 앞의 두 패스에 의해 부호화가 이루어지지 않은 계수값을 부호화 하는 패스이다.

각 패스에서는 해당 비트와 주위의 비트들, 그리고 이전 비트들에 따라(컨텍스트) 코드와 결정값이 부여되며, 각 경우의 값들은 이미 표준안으로 확정되어 있으므로 컨텍스트와 해당 비트 값에 해당하는 코드와 결정값을 선택한다. 이 때 표 2의 네 가지의 코딩방법이 적용되며, 각 패스에서 처리되는 코딩방법 및 그 순서는 표 3에 나타내었다. 예를 들어 CUP로 확정되면 먼저 RLC의 수행여부를 판별한다. 만약 RLC의 조건이 맞으면 RLC를 수행하고 SC의 조건을 만족하면 SC와 ZC를 수행한다.

표 2. 컨텍스트 형성을 위한 네 가지 부호화
Table 2. Four coding for context formation

Coding	Context choice
Zero coding (ZC)	0 ~ 8
Sign coding (SC)	9 ~ 13
Magnitude refinement coding (MRC)	14 ~ 16
Run length coding (RLC)	17 ~ 18

표 3. 각 패스코딩을 위한 처리순서
Table 3. Process order for each pass coding

Coding \ Pass	RLC	SC	ZC	MRC
SPP		①	②	
CUP	①	②	③	
MRP				①

2.3 EBCOT의 연산량 및 기존 연구

앞의 설명과 같이 EBCOT 알고리즘은 모든 비트들을 그 주위 및 이전의 값들에 따라 처리하기 때문에 계산량이 매우 많다는 단점을 갖고 있다. 표 4에는 JPEG2000 시스템의 각 과정에 대해 연산횟수의 비율을 나타내고 있다. 표에서는 흑백영상과 컬러영상을 구분하여 나타내었으며, 각 경우에서 양자화 과정의 포함여부에 따라 무손실 압축의 경우와 손실압축의 경우로 다시 분류하였다. 또한 각 처리 과정을 보다 상세히 나타내기 위해서 EBCOT 과정은 Tier 1과정과 Tier 2과정으로 나누었으며, Tier 1 과정은 다시 컨텍스트 추출과정과 MQ 코딩과정을 분리하였다. 표에서 보듯이 EBCOT는 전체 연산의 50~70% 정도를 차지하고 있으며 그 중 컨텍스트 추출과정이 30~50%의 가장 높은 비율을 차지하고

표 4. JPEG2000의 각 과정에 대한 연산 사이클 비율
Table 4. Rate of computation cycles for each step in JPEG2000

Operation		Computation cycle percentage(%)			
		Gray scale image		Color image	
		Lossless	Lossy	Lossless	Lossy
Color transform		N.A.	N.A.	0.91	14.12
DWT		10.81	26.38	11.90	23.97
Quantization		N.A.	6.42	N.A.	5.04
EBCOT Tier 1	Context Extraction	51.88	37.91	49.96	31.76
	MQ Coder	19.75	14.35	19.33	12.06
	Total	71.63	52.26	69.29	43.82
EBCOT Tier 2		17.75	14.95	17.90	13.01

있다. EBCOT 전체 비율 중 컨텍스트 추출과정의 비율이 감소할 경우 단순히 이 비율만 감소하는 것이 아니라 그것에 따르는 MQ-코더와 Tier 2의 비율이 부가적으로 감소하는 효과를 기대할 수 있다. 따라서 연산량이 과다하다는 JPEG2000 고유의 문제점을 해소하기 위해서는 컨텍스트 추출과정의 연산량을 감소시키는 것이 절실히 필요하다.

지금까지 진행되어 온 EBCOT 관련 연구들 대부분은 EBCOT 알고리즘의 연산횟수를 줄이는데 그 목적을 두고 있다. Chen은 컨텍스트 추출 알고리즘을 수행하는 과정에서 생기는 잉여연산을 줄이기 위해 Sample Skipping(SS), Group-of-column Skipping(GOCS), Multi Column Skipping(MCOLS) 등의 방법들을 제안하였다[8]. 컨텍스트 추출을 위한 연산 사이클 수는 SS 방법을 사용하였을 경우 약 22.2% 감소하였고, GOCS를 사용하였을 경우는 약 55%가 감소하였으며, 두 가지 방법을 모두 사용하였을 경우(MCOLS) 60%정도 감소하였다. 또한 그의 다른 연구 보고서에서는 비트평면의 값이 모두 '1'일 경우 세 가지 패스 중에서 두 가지 패스를 수행하지 않는 Pass Skipping(PxS) 방법을 제안하였으나 위의 방법들에 비해 큰 이득을 얻지는 못하였고 sample들을 그룹화하는 복잡한 연산이 추가되는 단점이 있다. Acharya는 컨텍스트 추출을 위해 각 부대역의 계수 값을 메모리에 효율적으로 저장하는 방법과 세 개의 컨텍스트 추출블록 및 MQ 부호화기를 사용하는 하드웨어 구조를 제안하였다^[9]. Hsieh는 컨텍스트 추출에서 생기는 잉여연산을 줄이기 위해 Pass-Parallel의 하드웨어 구조를 제안하였는데^[10], 이 방법은 연산시간을 약 25% 감소시켰다. Ohshima의 연구에서는 양자화 계수들을 비트평면으로 만들고 이를 다시 상위 비 평면과 하위 비

트평면으로 나누어 각각 4개씩의 MQ 부호화기를 사용하는 병렬처리 구조를 제안하였다^[11]. 이와 같은 하드웨어 구현방법들은 속도를 증가시키는 장점을 갖는 반면 높은 하드웨어 자원을 사용하는 단점이 있다.

III. 트리구조를 이용한 적응적 컨텍스트 추출 방법

앞 절에서 언급한 기존연구들 중 알고리즘 연구들은 모두 컨텍스트 추출과정의 특성 자체를 고려한 연구들이었다. 본 논문에서는 이 과정 자체의 특성이 아닌 웨이블릿 변환의 특성을 이용하여 네트워크의 상황 등에 적응적으로 적용할 수 있는 컨텍스트 추출 방법을 제안한다.

3.1 DWT 결과 부대역 계수들의 트리구조

DWT는 영상전체를 대상으로 해당 주파수 대역의 정보를 분리함으로써 각 주파수 대역에서도 원영상의 위치정보를 그대로 보유한다(그림 5 참조). DWT는 또한 저주파 성분으로 영상에너지를 집중시키는데, 이것은 저주파 성분에 더 민감한 눈의 특성을 잘 이용하기에 유리하다. 그림 5에서 저주파 성분에 해당하는 부분을 “coarse” 성분, 고주파 성분에 해당하는 부분을 “fine” 성분이라고 정의하였다. 또한 본 논문에서는 각 부대역에 XYi의 명칭을 부여하는데, 여기서 ‘i’는 변환레벨을 뜻하고, X와 Y는 각각 ‘H’(고대역통과) 또는 ‘L’(저대역통과)을 가지며 수평 및 수직방향의 필터링 종류를 나타낸다.

DWT 결과의 부대역들은 주파수대역에 따라 계층적인 구조를 갖는다. 즉, 최저주파수 대역을 기준

으로 세 방향의 차순위 저주파수 부대역들이 존재하고, 이 차순위 부대역부터는 각 방향성이 따라 그 다음 차순위 저주파 부대역들이 한개씩 존재하며 이것은 최고주파수 대역까지 이어진다. 앞에서 언급한 각 부대역이 원영상의 위치정보를 보유하고 있다(레벨이 증가할수록 크기가 1/4로 축소됨)는 사실은 위의 부대역간 계층성과 함께 부대역 계수들의 계층적 구조의 근거가 된다. 즉, 최저주파수 대역의 특정 위치의 계수는 그 레벨의 나머지 각 부대역의 해당위치 계수가 동일 화소의 정보를 포함한다. 최저주파수 대역을 제외한 부대역의 특정 계수는 그 방향의 다음 저주파 부대역에서 그 위치에 해당하는 4개의 계수에 해당하며, 그 방향으로 한 계층 고주파 부대역으로 갈수록 그 전 주파수 대역의 한 계수당 그 위치의 4개의 계수에 해당한다.

이러한 계수들의 구조를 ‘계수트리(coefficient tree)’라 한다. 본 논문에서는 이 계수트리 중 최저주파수 대역을 제외한 나머지 부대역들 간 계수트리만을 고려한다. 또한 저주파 부대역의 계수를 부모(parent)라 명명하고 그 다음 고주파 부대역의 4개의 계수들을 그 전 저주파 부대역 계수의 자식(children), 그 다음 고주파 부대역의 16개의 계수들은 손자(grand-children)라 명명한다. 그림 5의 예를 들면, LH3의 특정위치 계수는 LH2의 그 위치에 해당하는 4개의 계수를 자식으로 갖고 LH1의 그 위치에 16개의 계수들을 손자로 갖는다.

이 트리구조를 형성하는 계수집단은 다음과 같은 성질을 갖는다. 부모계수가 작은(큰) 값을 가지면 그 자식들 또한 작은(큰) 값을 가질 확률이 매우 높다. [12]에서는 이 확률이 98%에 이른다고 실험적으로 밝히고 있다. 이 성질을 좀더 확대하면 주어진 트리구조의 계수들 중 특정 위치의 계수가 작은(큰) 값을 가지면 자손들 모두가 작은(큰) 값을 가질 확률이 매우 높다(0.98j, j는 레벨차이).

3.2 트리구조-기반의 선행처리 과정

본 논문에서는 위 절에서 설명한 트리구조 계수들의 성질을 이용한다. 더구나 양자화 과정은 고주파성분일수록 양자화 폭을 크게 하는 것이 보통이므로 양자화를 거친 계수들(양자화 계수들)은 고주파쪽으로 갈수록 작은 값을 가질 확률이 더 높아진다. 따라서 엔트로피 코딩 바로 전의 계수들은 트리구조의 특정 위치의 계수가 특정 값보다 작으면 그 자손들이 매우 높은 확률로 그 값보다 작은 값을 갖는다.

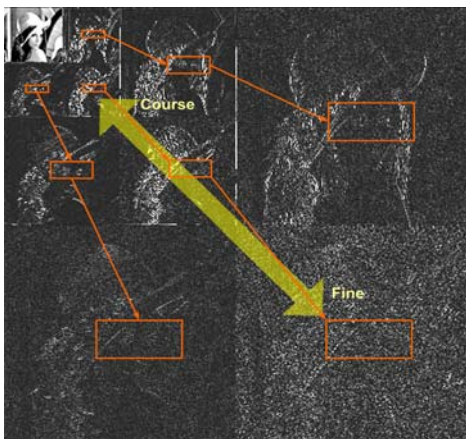


그림 5. 3-레벨 DWT 결과의 트리구조
Fig. 5. The tree structure of the result from 3-level DWT

본 논문의 근본적인 목적은 엔트로피 코딩에 소요되는 연산량을 줄여 연산시간을 감소시키는 것이다. 이를 위해 본 논문에서는 엔트로피 코딩 전의 선행처리과정을 제안한다. 기본 아이디어는 특정 문턱값(본 논문에서는 significance threshold(ST)라 부른다)보다 작은 계수와 그 자손계수들은 컨텍스트 추출과정을 거치지 않고 미리 정해진 코드를 부여함으로써 컨텍스트 추출대상 계수의 수를 줄이는 것이다. 이 ST값은 그보다 작은 계수들을 컨텍스트 추출과정 없이 처리하도록 하므로 이 값이 크면 컨텍스트 추출시간을 많이 줄일 수 있다. 그러나 ST 값이 커지면 데이터 손실이 커지므로 화질의 열화가 심해진다. 반면 문턱값이 커짐에 따라 컨텍스트 추출과정을 거친 데이터량이 적어져 결국 전송 또는 저장하여야 하는 데이터량이 감소하는 부가적인 효과가 얻어진다.

ST값에 따른 효과를 판별하기 위해 500개의 영상을 대상으로 실험을 수행하였다. 실험결과를 그림 6에 나타내었는데, ST 값에 따라 화질의 열화, 연산량(연산 사이클 수), 그리고 전송 데이터량을 각각 나타내었다. 예측한 바와 같이 ST값이 커짐에 따라 연산량이 감소하고 대신 화질의 열화가 심해지며 그 결과의 데이터량이 감소하였다. 따라서 ST 값에 따라 화질과 연산량 또는 전송데이터량 간에는 상보적인 관계를 나타냄을 알 수 있다. 이 관계는 네트워크의 상태 즉, 엔코딩/디코딩 단말기의 연산능력이나 전력량, 네트워크의 용량 등에 따라 선택적으로 ST값을 결정함으로써 네트워크 적응적인 엔코딩 기능을 부가할 수 있다.

그림 6 (a)에 의하면 평균 ST값이 5 이상이 되면 화질이 30dB 이하로 떨어지는 것을 볼 수 있다. 본 연구팀이 실험적으로 얻은 데이터에 의하면 30dB 전후에서 상대적으로 화질의 열화정도의 변화가 크다. 따라서 본 논문에서는 ST값을 0에서 4까지의 값만을 사용한다. 여기서 ST값이 0이라는 것은 본 논문에서 제안하는 연산량 감소효과가 전혀 없는 원래의 컨텍스트 추출과정을 그대로 거친다는 의미이다.

본 논문에서는 특정 부대역에 대하여 특정 계수가 ST값보다 크지(SC, significant coefficient) 작을지(IC, insignificant coefficient)를 판별하는 과정을 선행처리 과정이라고 하며, 그 내용을 그림 7에 요약하였다. 이 그림에서 XYi는 LHi, HLi, HHi 중 하나이며, j-레벨 DWT가 수행된 것을 가정하였다($1 \leq i \leq j$). 그림에서 보는 바와 같이 이 선행처리과정

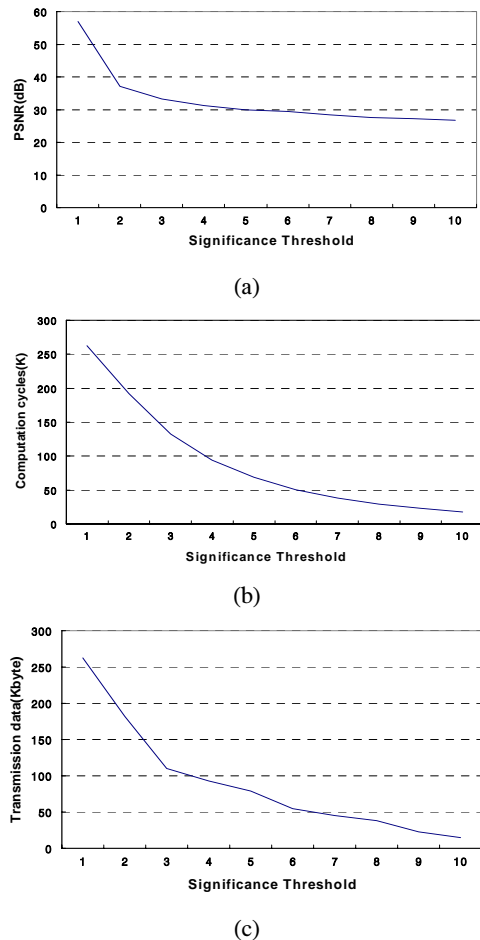


그림 6. ST값의 변화에 따른, (a) 화질의 열화, (b) 연산량의 감소, (c) 데이터량의 감소
 Fig. 6. According to the ST values, (a) image quality degradation, (b) reduction of the amount of calculation, (c) reduction of the amount of data

은 각 레벨의 각 부대역별로 이루어지며, 각 계수 중 부모 부대역의 IC 계수들에 해당하는 계수들은 무조건 IC로 판별하고 부모 부대역의 SC 계수들에 해당하는 계수들에 대해서만 주어진 ST값과 비교하여 그 계수를 SC 또는 IC로 판별한다. 이 판별 결과는 중요계수맵(SCMap_XYi)에 저장되어 컨텍스트 추출과정에서 사용된다.

선행처리과정 결과의 중요계수맵을 이용한 컨텍스트 추출과정은 그림 8에 나타내었다. 컨텍스트 추출은 표준에서와 같이 코드블록의 비트평균별로 4비트의 코딩단위별로 이루어진다. 4비트 중 중요계수맵의 값이 'IC'인 값은 컨텍스트 추출과정 없이 컨텍스트와 결정값 모두 '0'값을 할당한다. 해당 중요계수맵의 값이 'SC'인 계수에 대해서는 정상적인

```

PreProcessing_SC_Selection{
input: XYi, ST
output: SCMap_XYi
begin {
  for each subband of XYi {
    initialize SCMap_XYi as nulls;
    for each coefficient Cp,q {
      if parent(Cp,q) was IC then
        SCMap_XYi(p, q)='IC';
      else if Cp,q<ST then
        SCMap_XYi(p, q)='IC';
        Cp,q='0';
      else
        SCMap_XYi(p, q)='SC'; }
    send SCMap_XYi to ContextExtraction{;} }
  if i=1 then
    end;
  else
    i=i-1;
  PreProcessing_SC_Selection{;} }
    
```

그림 7. 제안한 선행처리 과정
Fig 7. The proposed pre-process procedure

```

ContextExtraction_XYi {
input: XYi, SCMap_XYi
output: context and decision values
begin {
  for each code_block {
    for each bit_plane {
      for each coding-unit {
        for each coefficient Cp,q {
          if SCMap_XYi(p, q)=IC then
            context='0';
            decision='0';
          else
            proceed normal context extraction process;
        }
      }
    }
  }
}
    
```

그림 8. 제안한 컨텍스트 추출 과정
Fig 8. The proposed context extraction procedure

컨텍스트 추출과정을 거치며, 이 때 주위의 'IC' 계수값은 '0'으로 간주한다. 정상적인 컨텍스트 추출 과정은 앞 장의 설명으로 대신하며, 더 상세한 내용은 참고문헌 [4]를 참고하도록 하고 본 논문에서는 설명을 생략한다.

IV. 실험 및 결과

본 논문에서 제안한 방법의 검증을 위해 선행처리 과정과 컨텍스트 추출과정을 C-언어로 구현하였고, 본 연구실에서 미리 구현한 JPEG2000 시스템의 해당 위치에 삽입하였다. 실험은 Pentium IV 2.8GHz 환경에서 수행하였으며, 실험에 사용된 영상은 512×512 크기의 500개 영상들이었다. 이 중 본 장에서는 ISO 표준영상인 Lena, Goldhill, Woman의 세 영상을 예로 들어 설명하도록 한다.

표 5에서는 이 세 영상을 4-레벨 웨이블릿 변환한 후 제안한 선행처리과정에 따라 SC와 IC로 결정된 계수들을 부대역 별로 정리한 것으로 이 때의 ST는 '1'이었다. 표에서 볼 수 있듯이 LL4에서의 IC의 비율은 7~8%에 불과하나 LH4, HL4, HH4의

표 5. 선행처리 과정으로 판별된 SC와 IC 개수(ST=1)
Table 5. Number of SCs and ICs determined by the preprocess (ST=1)

Sub-band	SC			IC		
	Lena	Gold-hill	Wo-man	Lena	Gold-hill	Wo-man
LL4	950	943	951	74	81	73
HL4	636	887	684	388	137	340
LH4	858	747	782	166	277	242
HH4	796	785	785	228	239	239
HL3	2133	3298	2285	1963	798	1811
LH3	2961	2285	2631	1135	1225	1465
HH3	2926	2871	2800	1170	1023	1296
HL2	8800	12523	8740	7584	3861	7644
LH2	11344	11114	9774	5040	5270	6610
HH2	12747	12169	11484	3637	4215	4900
HL1	41794	48082	31439	23787	17454	34097
LH1	51183	45353	35898	14353	20183	29638
HH1	55166	49812	35868	10370	15724	29668
Total	192249	191657	144121	69895	70487	118023

다음 고주파 부대역에서는 20~40%로 급증하여 최저주파수대역인 LL4에 에너지가 집중되어 있음을 알 수 있다.

또한 특정 트리구조를 이루는 부대역들에서 IC의 개수는 고주파 부대역으로 갈수록 높아지나 그 비율은 반드시 증가하지는 않는다. 또한 각 트리구조의 IC 비율은 그 영상의 방향성에 따라 값이 서로 다를 수 있다. 예를 들어 Lena 영상의 경우 HL4, HL3, HL2, HL1의 IC 개수는 각각 388, 1963, 7584, 23787로 증가하나, 그 비율은 37.89%, 47.92%, 46.29%, 36.30%를 각각 나타낸다. 또한 LH4, LH3, LH2, LH1의 IC 개수는 각각 166, 1135, 5040, 14353을 나타내었으며, 그 비율은 16.21%, 27.71%, 30.76%, 21.90%이다.

그림 9는 ST값을 2로 하였을 때 세 영상에 대해 원영상과 복원된 영상을 비교한 그림이다. ST=2일 때의 평균 화질은 약 38.7dB이었으며, 그림의 세 영상에서 Lena의 어깨나 Goldhill의 지붕 등에서 약간의 열화를 보일 뿐 화질에 큰 영향을 주지 않음을 알 수 있다.

그림 10에서는 ST값을 1에서 4까지 변화하였을 때의 복원된 Lena 영상을 보이고 있다. ST값이 증가할수록 Lena의 어깨, 모자, 코의 일부분에서 점차 심한 열화현상이 나타남을 알 수 있으며, 이로써 표 5와 비교할 때 ST값이 증가함에 따라 연산량이 감소함과 동시에 화질이 감소하는 상보적인 관계를 확인할 수 있다.



그림 9. ST=2일 때의 화질 테스트 결과; (a)원 Lena 영상, (b) 복원된 Lena 영상, (c) 원 Goldhill 영상, (d) 복원된 Goldhill 영상
 Fig. 9. The results from quality tests when ST=2; (a) original Lena image, (b) reconstructed Lena image, (c) original Goldhill image, (d) reconstructed Goldhill image



그림 10. ST값에 따라 복원된 Lena 영상들; (a) ST=1, (b) ST=2, (c) ST=3, (d) ST=4
 Fig. 10. Reconstructed Lena images according to ST values; (a) ST=1, (b) ST=2, (c) ST=3, (d) ST=4

표 6. Taubman 방법과의 연산시간 비교
 Table 6. Comparison of calculation time with Taubman method

Image	Taubman		This paper					
	CPU time (sec)	Data rate (byte)	ST	# of SCs	CPU time		Data rate	
					Value (sec)	Reduct. rate(%)	Value (byte)	Reduct. rate(%)
Lena	0.246	157,020	0	262144	0.246	0	157,020	0
			1	192249	0.236	4.07	115,455	26.47
			2	132212	0.164	33.30	79,303	49.49
			3	94001	0.115	53.25	56,279	64.16
			4	68581	0.084	65.85	40,997	73.90
Goldhill	0.241	155,551	0	262414	0.241	0	155,551	0
			1	191657	0.234	2.35	113,540	27.00
			2	138090	0.171	29.05	81,868	47.37
			3	104186	0.126	47.72	61,726	60.32
			4	78937	0.099	58.92	46,852	69.88
Woman	0.183	125,216	0	262144	0.183	0	125,216	0
			1	144121	0.179	2.19	68,800	45.05
			2	91680	0.113	38.25	43,781	34.97
			3	64625	0.080	56.28	30,841	75.37
			4	48059	0.061	66.67	22,602	80.95

표 6에서는 앞의 세 영상을 대상으로 컨텍스트를 추출하기 위한 연산량에 대해 제안한 방법과 Taubman의 방법[5-7]을 비교하였다. Taubman 방법은 모든 샘플값을 고려해야 하므로 계수의 개수만큼의 연산 횟수가 필요하나, 본 논문의 방법은 트리 구조를 이용함으로써 ST값 이하의 계수들 및 그 자

손들은 컨텍스트 추출과정을 거치지 않으므로 계산량이 확연히 감소함을 알 수 있다. 이 표의 연산량은 CPU 시간으로 측정하였으며, 여기에는 본 논문에서 제안한 선행처리과정을 수행하기 위한 시간도 포함되어 있다. 따라서 ST=1의 경우는 세 경우 모두 4% 이하의 낮은 시간감소를 보이나, ST=2 이상

의 경우는 약 30% 이상의 시간감소를 얻을 수 있으며, ST=4의 경우 최대 약 67%의 시간이 감소를 보이고 있다.

이상의 실험결과들에서 보는 바와 같이 본 논문에서 제안하는 방법은 네트워크의 조건이나 환경에 따라 ST값을 조절함으로써 화질의 열화를 보상으로 연산량을 줄이고 출력되는 데이터량을 감소시키는 네트워크 적응적 엔트로피 코딩 방법이다.

V. 결론

본 논문에서는 정지영상 압축 기술의 국제표준인 JPEG2000 기술 중 엔트로피 코딩방법에서 연산량이 가장 많은 컨텍스트 추출 알고리즘의 연산량을 줄이는 방법을 제안하였다. 이 방법은 웨이블릿 변환 결과 계수들의 트리구조와 이 구조의 특징, 즉 특정 계수의 값이 작으면 그 자손들의 값 또한 작을 확률이 매우 높다는 성질을 이용하였다. 이것은 선행처리과정에서 특정 계수의 값이 특정 임계값보다 작은 계수들을 선별하며, 컨텍스트 추출과정에서 그 계수와 그 자손 모두에 컨텍스트 추출과정을 거치지 않고 정해진 일정한 컨텍스트 값과 결정값을 할당하도록 하였다. 임계값은 실험결과 화질의 열화를 수용할 수 있는 0에서 4까지의 값으로 한정하였으며, 그 결과는 평균 30dB 이상의 화질을 유지하였다.

제안한 방법을 구현하여 실험한 결과 본 논문과 가장 유사한 기존의 방법과 비교할 때 임계값을 1로 하였을 때 평균 3%, 임계값이 2일 때는 약 34%, 임계값이 3일 때는 약 52%, 그리고 임계값이 4일 때는 약 64%의 연산시간 감소를 얻었다. 그러나 화질에서는 임계값이 1에서 4까지 변화할 때 평균 약 56dB, 39dB, 34dB, 31dB의 열화를 보였으며, 평균 데이터는 31%, 42%, 66%, 73%의 감소율을 각각 보였다.

따라서 본 논문은 임계값이 증가함에 따라 연산시간과 화질 또는 데이터량 간에 상보적인 관계를 보인다. 임계값을 네트워크의 환경 또는 조건(네트워크의 용량, 단말기의 연산능력 및 전력량 등)에 따라 달리하면 네트워크 적응적인 엔트로피 코딩 방법으로 유용하리라 판단된다. 특히 최근의 경향은 무선 네트워크의 사용빈도가 급증하고 있으며, Motion JPEG2000과 같이 정지영상 뿐 아니라 동영상에도 JPEG2000을 사용하는 비율이 증가하고 있다. 따라서 휴대용 단말기를 사용하여 정지영상

또는 동영상을 포획하여 전송하거나 쌍방향성 동영상 통신 등에 본 논문의 방법이 매우 유용하게 사용될 수 있을 것으로 기대된다^{[13],[14]}.

본 논문에서 제안한 방법은 컨텍스트 추출 알고리즘 자체의 특성은 고려하지 않았다. 기존에 컨텍스트 추출 알고리즘 자체의 특성을 고려한 방법들^[8-10]이 제안되었는데, 본 논문의 방법과 이 방법들을 결합한다면 더욱 높은 연산시간 감소율을 보여 더욱 효과적인 방법이 될 것으로 판단된다.

참고 문헌

- [1] "Information technology-JPEG-Digital compression and coding of continuous-tone still image-Part 1: Requirement and guidelines", ISO/IEC 10918-1 and ITU-T Recommendation T.81, 1994.
- [2] ISO/IEC 14496-1 MPEG-4 "Coding of Audio-Visual Objects-Part 2 : Visual", Aug. 2002.
- [3] Joint Video Team of ITU-T and ISO/IEC JTC 1. "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification(ITU-T Rec. H.264 ISO/IEC 14496-10 AVC)", Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, March 2003.
- [4] JPEG2000 Final Part I: Final Draft International Standard. ISO/IEC FDIS 15444-1, ISO/IEC JTC1/SC29/WG1 N1855, Aug. 2000.
- [5] D. Taubman and HP Labs, "Report on core experiment CodEff22, EBCOT: Embedded block coding with optimized truncation", Tech. Rep.N1020R, ISO/IEC JTC1/SC29/WG1, October 1998.
- [6] D. Taubman, "High performance scalable image compression with EBCOT", IEEE Trans. Image Processing, Vol. 9, pp. 1158-1170, July 2000.
- [7] D. Taubman, E. Ordentlich, M. Weinberger, and G. Seroussi, "Embedded block coding in JPEG2000", IEEE Trans. Signal Processing: Image Commun., Vol. 17, No. 1, pp. 49-72, Jan. 2002.
- [8] C. Lian, K. Chen, and L. Chen, "Analysis and Architecture Design of Block-Coding

Engine of EBCOT in JPEG2000” IEEE Trans. on Circuit and Systems for Video Technology, Vol.13, No. 3, pp. 219-229, March 2003.

- [9] K. Andra, C. Chakrabarti, and T. Acharya, “A High-Performance JPEG2000 Architecture”, IEEE Trans. on Circuit and Systems for Video Technology, Vol. 13, No. 3, March 2003.
- [10] J. S. Chiang, Y. S. Lin, and C. Y. Hsieh, “Efficient Pass-Parallel Architecture for EBCOT in JPEG2000”, IEEE Int. Circuits and Systems, Vol. I, pp. 773-776, 2002
- [11] H. Yamauchi, S. Okada, K. Taketa, and T. Ohyama, “A Single-Chip JPEG2000 Encoder Processor Capable of Compressing D1-Image at 30frames/s without Tile Division”, IEICE Trans. ELECTRON., Vol.,E87-C, No. 4, April 2004.
- [12] J. M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficient”, IEEE Trans. Signal Processign, Vol. 41, No. 12, pp. 3445-3463, Dec. 1993.
- [13] J. R. Ohm, Multimedia Communication Technology, Springer, New York, 2003.
- [14] J. R. Ohm, “Registered Responses to the Call for Proposals on Scalable Video Coding”, MPEG Report, M10569, Munich, 2004.

최 현 준 (Hyun-Jun Choi) 정회원



2003년 2월 광운대학교 전자재료공학과 졸업(공학사).
2005년 2월 광운대학교 대학원 졸업(공학석사).
2005년 3월~현재 광운대학교 전자재료공학과 박사과정.
<관심분야> Image Processing, 암호학, FPGA/ASIC 설계

서 영 호 (Young-Ho Seo) 정회원



1999년 2월 광운대학교 전자재료공학과 졸업(공학사).
2001년 2월 광운대학교 대학원 졸업(공학석사).
2000년 3월~2001년 12월 인티스닷컴(주) 연구원.
2003년 6월~2004년 6월 한국전

기연구원 연구원

2004년 8월 광운대학교 대학원졸업(공학박사)
2004년 9월~2004년 11월 유한대학 겸임교수
2004년 12월~2005년 8월 유한대학 연구교수
2005년 9월~현재 : 한성대학교 정보통신공학과 교수
<관심분야> Image Processing/Compression, 워터마킹, 암호학, FPGA/ASIC 설계

김 동 욱 (Dong-Wook Kim) 종신회원



1983년 2월 한양대학교 전자공학과 졸업(공학사).
1985년 2월 한양대학교 대학원 졸업(공학석사).
1991년 9월 Georgia공과대학 전기공학과 졸업(공학박사).
1992년 3월~현재 광운대학교 전자재료공학과 정교수. 광운대학교 신기술 연구소 연구원.

2000년 3월~2001년 12월 인티스닷컴(주) 연구원.
<관심분야> 디지털 VLSI Testability, VLSI CAD, DSP 설계, Wireless Communication