

JPEG2000 응용에서 적절한 ROI 코딩 방법을 적용하기 위한 매개변수의 성능 평가

강 기 준[†] · 이 부 권^{††} · 서 영 건^{†††}

요 약

현재 여러 응용 프로그램에서 이미지 전체를 전송 및 복원하기보다는 이미지의 일부 특정 영역이나 사용자 위주의 관심영역(ROI: Region-of-Interest)에 대한 우선적 처리가 요구되고 있다. 이러한 우선적 처리는 JPEG2000 표준 ROI 코딩 방법들을 비롯하여, 이를 보완한 새로운 ROI 코딩 방법들에 대한 연구가 최근까지 활발하게 이루어지고 있다. 하지만 모든 응용에 적합한 ROI 코딩 방법은 존재하지 않는다. 본 연구는 JPEG2000 응용에서 적절한 ROI 코딩 방법을 적용하기 위하여 요구사항에 따른 ROI 코딩 방법의 선택 기준을 제시하고, 선택된 방법에 대하여 효율적인 매개변수 값을 결정할 수 있도록 실험 결과를 제시한다.

키워드 : JPEG2000, ROI, EBCOT

Performance Evaluation of Parameters for applying an Adequate ROI coding method in JPEG2000 Applications

Ki-Jun Kang[†] · Bu-Kwon Lee^{††} · Yeong-Geon Seo^{†††}

ABSTRACT

Currently, the preferred processing of a user-centered ROI(Region-of-Interest) or a specific region of image than the transmission and decompression of a full image is needed in different applications. This preferred processing has been actively studied about from the ROI coding methods in JPEG2000 standards to the new methods complementing them. But, there does not exist an ROI coding method suitable for all applications. Therefore, this study shows a criterion of selection according to the application requirements for applying an adequate ROI coding method in JPEG2000 applications, and shows the experimental results deciding efficient parameters in the selected methods.

Key Words : JPEG2000, ROI, EBCOT

1. 서 론

인간이 받아들이는 정보 중에 이미지와 같은 시각적인 정보는 전체 정보 중 상당한 부분을 차지할 뿐만 아니라 단순한 텍스트나 그래픽보다는 전달매체로서의 효과도 뛰어나기 때문에 의료진단, 웹 브라우징, 이미지 데이터베이스 그리고 컴퓨터 통신 등 다양한 응용 분야 전반에 걸쳐 활용되고 있다[1]. 하지만 이러한 이미지를 저장 및 전송하기 위해서 많은 메모리와 넓은 대역폭을 요구하기 때문에 사용자의 다양한 요구를 만족하는 이미지 활용은 어려운 실정이다.

이러한 문제를 해결하기 위해 이미지의 중복성을 이용하

여 원 이미지 품질을 시각적으로 크게 저하시키지 않는 범위 내에서 이미지를 압축하는 JPEG2000(Joint Photographic Experts Group 2000)이라는 새로운 표준을 개발하고 있다 [2].

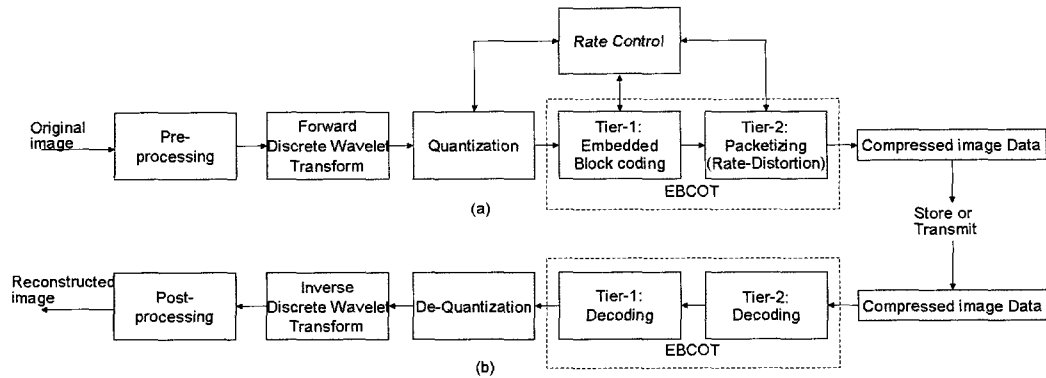
특히 JPEG2000은 기존의 표준에서는 제공하지 않는 ROI 코딩이라는 기능을 제공한다. 이 기능은 이미지 전체를 복원하기 이전에 이미지의 특정 영역이나 사용자 위주의 ROI에 대해서 우수한 화질을 보장하면서 먼저 전송하여야 하는 응용에서 활용된다. 표준 ROI 코딩 방법에는 Maxshift[3], Implicit[4], 일반적인 Scaling[3]이 있고, Maxshift와 일반적인 Scaling 방법의 단점을 보완한 Maxshift-Like[3], (G)BbBShift[5, 6], PSBShift[7], HBSHift[8] 등과 Implicit 방법의 단점을 보완한 Modified Implicit[4], Flexible and dynamic[9], Fast ROI Transcoding[10], Prioritized[11] 등이 있다.

[†] 준 회원 : 경상대학교 컴퓨터과학과 박사과정

^{††} 정 회원 : 경상대학교 컴퓨터과학과 교수

^{†††} 종신회원 : 경상대학교 컴퓨터교육과 부교수

논문접수 : 2006년 3월 14일, 심사완료 : 2006년 5월 3일



(그림 1) JPEG2000 (a)인코더와 (b)디코더 블록 다이어그램

이처럼 새로운 ROI 코딩 방법에 대한 연구는 활발하게 이루어지고 있다. 하지만, ROI 코딩 방법을 응용에 적용시키는 연구는 단순히 표준 ROI 코딩 방법에 디폴트 매개변수 값을 사용하여 적용시키는 수준에 그친다. 이렇게 하여도 서비스는 가능하지만, 최적의 ROI 서비스는 못된다[12, 13].

따라서 본 연구에서는 요구사항에 따른 ROI 코딩 방법 분류표와 실험에 의한 ROI 매개변수의 효과와 몇 가지 요점을 고려하여 특정 응용에 적합한 ROI 코딩 방법을 적용할 수 있는 방법을 제안한다. ROI 코딩 방법 분류표는 각 ROI 코딩 방법에 대한 알고리즘 이해를 바탕으로 특징들을 비교 분석하여 몇 가지 주요 요구사항에 따라 적합한 순으로 정리한다. 실험은 표준 ROI 코딩 방법들을 가지고 ROI 매개변수 값의 변화에 따른 ROI 코딩 성능을 조사하고, 이를 통해 ROI 코딩 방법에 몇 가지 요점들을 함께 제시한다.

2. 관련 연구

2.1 JPEG2000 이미지 압축 표준

JPEG2000은 ISO/IEC 산하의 JTC1/S29/WG1 그룹에서 새롭게 개발된 이미지 압축 표준이다. 이 표준은 기존의 표준기술보다 월 왜곡, 이미지성능 면에서 우수성을 제공할 뿐만 아니라 현재의 표준들이 언급할 수 없는 특징이나 기능들을 제공한다.

JPEG2000의 성능은 기존의 JPEG에 비해 같은 비트율에서 이미지의 품질이 대략 2dB 정도 향상된 것으로 나타난다. 그리고 같은 품질에 대해서 압축 향상이 20-30%로 뛰어난 것으로 나타난다. 특히 저 비트율에서 압축률은 더욱 우수하다[1]. JPEG2000의 대표적인 특성은 무손실/손실 압축, 무손실 코딩에서의 내포된 손실, 화소 정확성이나 해상도에 의한 진보적인 전달, 비트 오류 그리고 ROI 코딩 등이 있다. JPEG2000의 필요성이 강조되는 다양한 응용으로는 인터넷, 컬러 팩스, 인쇄, 스캐닝, 디지털 사진, 이동식 적용, 의학 이미지, 전자 도서관과 전자상거래 등이 있다[2].

(그림 1)의 (a)처럼 JPEG2000 인코딩 과정은 원 이미지 데이터를 가지고 전 처리 과정을 거쳐 이산 웨이블릿 변환

을 하고, 변환된 웨이블릿 계수(WC: Wavelet Coefficient)는 양자화한 후 엔트로피 과정을 통해 코드 열을 생성한다.

2.2 ROI 코딩 방법

2.2.1 ROI의 필요성

ROI 코딩은 JPEG2000의 주요 특징 중에 하나로서 이미지 내의 ROI를 BG(Background)보다 더 나은 품질로 빠르게 처리하는 기술로서 이미지 내의 특정 영역이 다른 영역보다 높은 중요성을 가지는 응용에 이용할 수 있다.

ROI 코딩을 지원하기 위해서 ROI는 BG보다 높은 화질에서 인코딩되고 이미지를 전달하는 동안 ROI는 높은 우선순위로 보내질 필요가 있다. 이 기술은 이미지의 다 해상도 해석을 이용한 점진적인 이미지 코딩 기반 위에 달성될 수 있다. ROI 이미지의 장점은 점진적 이미지 전송 방법보다 압축률과 전송시간을 단축시킬 수 있을 뿐만 아니라 효율적인 메모리 관리 등 사용자의 다양한 요구에 응할 수 있다.



(a) 점진적인 이미지와 (b) ROI 이미지

(그림 2) (a) 점진적인 이미지와 (b) ROI 이미지

(그림 2)는 0.125bpp에서 점진적인 이미지와 ROI 이미지로 재구성된 "Lena" 이미지를 나타낸다. ROI 코딩 방법은 타일, 계수 스케일링, EBCOT(Embedded Block Coding with Optimized Truncation algorithm)기반으로 나뉜다.

2.2.2 타일 기반 ROI 코딩 방법

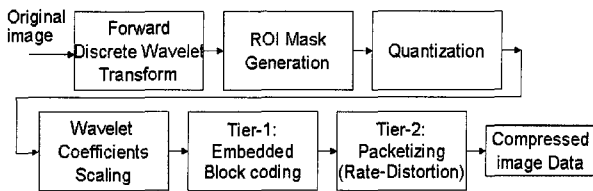
JPEG2000에서는 이미지가 큰 경우, 메모리를 줄이기 위해서 최초 이미지를 정사각형의 비중복된 블록들로 분할하여 압축하게 되는데, 이 과정을 타일링이라 하며, 분할된 블

록들을 타일이라고 한다.

JPEG2000에서는 모든 압축 과정을 이미지 타일 위에서 독립적으로 이루어진다. 따라서 전체 이미지 대신에 타일 단위로 ROI 코딩을 할 수 있다. 즉 BG 타일보다 ROI 타일을 높은 품질로 압축/복원 할 수 있다. 이 방법은 가장 단순한 ROI 코딩 방법으로서 메모리, 하드웨어 등의 제약이 있는 응용에 사용될 수 있다[14].

2.2.3 계수 스케일링기반 ROI 코딩 방법

계수 스케일링 기반 ROI 코딩은 양자화된 WC들 중에 ROI WC들과 BG WC들을 구분하여, ROI 중요도에 따라 WC들의 비트-평면(BP: Bit-Plane)을 이동시키는 방법이다. 이 방법은 인코딩 시에 ROI가 정의되는 정적인 ROI 코딩 방법에 속한다.



(그림 3) 계수 스케일링 기반 ROI 코딩 과정

일반적인 계수 스케일링 기반의 ROI 코딩 과정은 (그림 3)과 같고, 다음과 같은 순서로 수행된다.

- (1) DWT를 수행한다.
- (2) 사용자에게 의해 ROI가 정의되면 ROI WC들을 가리키는 ROI 마스크를 생성한다.
- (3) 양자화된 WC들은 부호 절대값 표현으로 저장한다.
- (4) WC들의 BP들을 중요도에 따라 이동시킨다.
- (5) WC들은 점진적인 엔트로피 코딩한다.

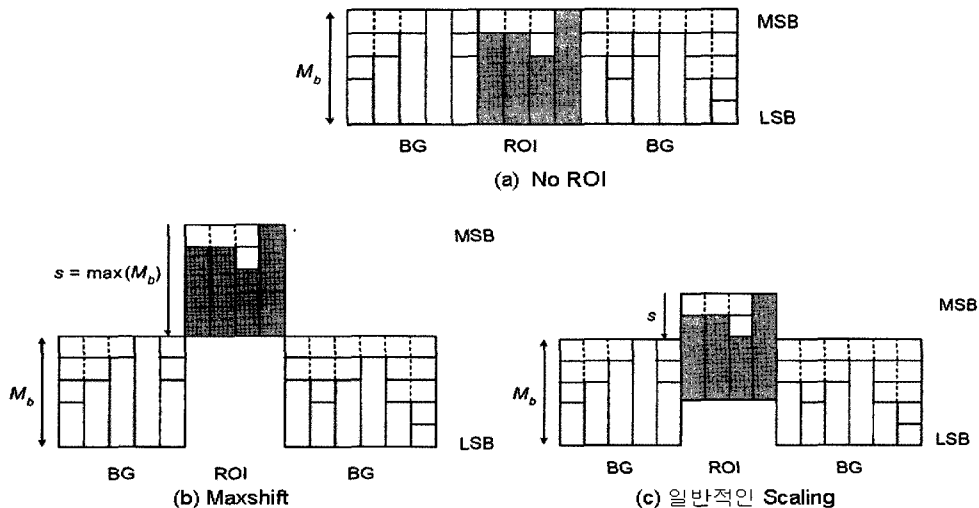
디코더는 이미지를 복구하기 위해 이 단계를 역으로 한다. 이 방법에는 Maxshift[3], Scaling based[3], Maxshift-like [3], (G)BbBShift[5, 6], PSBShift[7] 등이 있다. Maxshift 방법은 Part1 표준으로서 BG WC들 중에서 가장 큰 계수 값, s 를 구한 다음 ROI WC들을 s 보다 높은 BP에 이동시킨다. 반면에 일반적인 Scaling 방법은 Part2 표준으로서 ROI 중요도에 따라 s 를 적절하게 조절할 수 있다. 그러나 인코더 부분에서 ROI 모양을 인코딩하여 전송해야 한다(그림 4).

Maxshift-like[3] 방법은 우선 일반적인 Scaling 방법처럼 ROI 중요도에 따라 ROI WC의 BP를 스케일업한 후, ROI의 BP와 BG의 BP이 겹치는 BP의 수만큼 ROI의 LSB를 삭제하고, 양자화 스텝 사이즈도 적절하게 조절함으로써 구현된다. 그리고 BG가 ROI보다 높은 품질을 가지지 않도록 하기 위하여, ROI로부터 삭제된 BP 수만큼 BG의 LSB도 삭제한다. BbBShift 방법[5]는 BP 마스크를 사용하여 BP-by-BP 기초로 ROI와 BG의 BP들을 interleaving overlapping하게 이동시켜 구현된다. PSBShift 방법[7]은 높은 비트율에서는 ROI와 BG 모두 높은 품질을 가진다는 사실을 기반으로 ROI의 일부 MSB만을 이동한다.

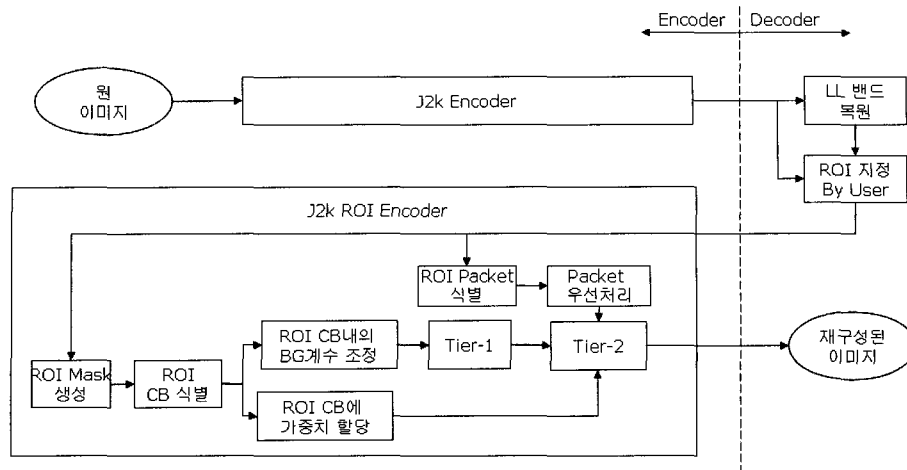
HBSHift 방법[8]은 중요도에 따라 BP를 MSR(Most significant ROI), GSRB(General Significant ROI and BG), LSRB(Least Significant ROI and BG)으로 분류하여 이동한다.

2.2.4 EBCOT 기반 ROI 코딩 방법

디코더에서 ROI를 정의하여 ROI 코딩하는 방법을 동적 ROI 코딩이라고 하며, EBCOT 기반의 ROI 코딩 방법은 동적 ROI 코딩 방법에 속한다. EBCOT에서 각 품질 레이어는 CB(혹은 Packets, Precincts)들의 임베디드 비트 스트림으로부터 임의의 공헌도를 포함한다. 따라서 ROI 강조는 ROI로 재구성되어야 하는 CBs(혹은 Packets, Precincts)의 품질 레이어에 대해 비교적 높은 공헌도를 포함시켜 달성된다. 일반적인 EBCOT 기반 ROI 코딩 과정은 다음과 같다(그림 5).



(그림 4) (b) Maxshift 방법과 (c) 일반적인 Scaling 방법



(그림 5) EBCOT 기반 ROI 코딩 과정

- 1) 원 이미지를 ROI 코딩 없이 압축을 한다.
- 2) 디코더에서 기본적인 정보 혹은 LL 밴드 내용을 전송 받아 이를 통해 사용자가 ROI를 정의하여 그 정보를 인코더로 전송한다.
- 3) ROI 마스크를 생성한다.
- 4) ROI CB(혹은 Packets, Precincts)들을 식별한다.
- 5) ROI CB들에 우선적 처리를 한다.
- 6) 재구성된 이미지 출력한다.

ROI 코딩 방법들이 활발히 연구되고 있다.

하지만 모든 응용에 적합한 최적의 ROI 코딩 방법은 존재하지 않는다. 왜냐하면 응용마다 요구사항이 다를 뿐만 아니라, ROI 코딩 방법들이 나름대로 장단점을 가지며, 또한 ROI 코딩 성능을 좌우하는 많은 ROI 매개변수들이 존재하기 때문이다. 따라서 최적의 ROI 서비스를 위해서는 많은 ROI 코딩 방법들 중에 특정 응용에 적합한 ROI 코딩 방법과 ROI 매개변수 값을 선택하여 적용하는 문제가 필요하다.

대표적인 EBCOT 기반 ROI 코딩 방법에는 Implicit[4], Modified Implicit[4], Flexible and dynamic[9], Fast ROI Transcoding[10] 그리고 Prioritized[11] 등이 있다.

Implicit[4] 방법은 Part1 표준으로서 CB 내에 ROI WC가 포함하면 ROI CB으로 처리를 한다. 따라서 ROI CB에 포함된 BG WC도 우선적 처리를 하는 문제가 발생한다. 이를 보완하기 위해 Modified Implicit[4] 방법은 ROI CB에 포함된 BG WC들의 우선권을 줄이기 위해 ROI CB에 포함된 BG WC들의 BP들 중에서 k 만큼의 LSB(Least Significant Bit-planes)를 삭제한다.

Flexible and Dynamic[9] 방법은 ROI 정의에 따라 처리될 Precincts를 계산하여 데이터 스트림에서 해당 패킷들을 식별하고, 식별된 패킷들 사이에 동적으로 빈 패킷들을 채우는 방식으로 우선적 처리를 한다. Fast ROI Transcoding[10] 방법은 패킷 헤더의 재 코딩을 하지 않고 단순히 패킷 삭제와 재정렬을 통해 우선적 처리를 한다.

Prioritized[11] 방법은 ROI 우선적 처리하는 것뿐만 아니라 ROI와 인접한 BG도 우선적 처리가 가능하다. 이 방법은 가우시안 분배를 사용하여 ROI로부터 BG의 거리에 의해 얼마나 많은 BG를 ROI에 포함시킬 것인지를 결정한다.

2.3 최적의 ROI 코딩 방법 선택의 필요성

ROI 코딩은 이미지 품질과 압축률 간에 뛰어난 tradeoff를 제공함으로써 다양한 응용에서 사용자의 요구를 충족시킬 수 있기 때문에 앞 장에서 소개된 것처럼 최근까지 많은

3. ROI 코딩 방법 비교 및 적용

본 장에서는 2장에서 소개된 ROI 코딩 방법들의 비교를 통해 몇 가지 요구사항에 따라 적합한 순으로 ROI 코딩 방법을 분류한다. 그리고 ROI 매개변수들을 소개한다.

3.1 ROI 코딩 방법 비교

<표 1>은 ROI 코딩 방법들 간의 비교를 나타낸다. 표준 항목에서 ①②⑦은 JPEG2000 표준 방법이고, ④⑤⑥은 표준과 호환이 안 되므로 별도의 모듈이 필요한 방법이고, 그 외는 표준과 호환되는 방법이다.

ROI 모양은 ROI 처리 단위와 밀접한 관련이 있다. 계수 스케일링 기반은 모두 WC 단위로 ROI 처리를 하기 때문에 임의의 모양을 지원한다. 단, ②는 ROI 모양 코딩으로 인한 오버헤드 때문에 직사각형과 타원만을 지원한다. 그리고 EBCOT 기반은 .CB(혹은 Precinct, Packet) 단위로 ROI 처리하기 때문에 다각형 모양을 지원한다. 단 ⑧인 경우는 임의의 모양을 지원한다.

ROI 중요도는 ①⑨를 제외하고는 모두 조절이 가능하다. 다양한 등급으로 다중 ROI는 ①③④⑨를 제외하고는 모두 지원한다. 단 ⑩은 ROI 중요도 조절이 안 되지만, ROI 주위의 일부 BG 중요도 조절이 가능하다. 계수 스케일링 기반은 ③②, ⑤⑥④, ①순으로 가변 범위의 변화가 적고, EBCOT 기반은 가변 범위의 변화가 전혀 일어나지 않는다. ③⑧⑩ 방법은 코딩 성능을 더욱 더 향상시키기 위해서 삭제하는

<표 1> ROI 코딩 방법들 간의 비교

구분	ROI 방법	표준	ROI모양	정적 /동적	ROI 중요도 조절	다수 ROIs	ROI 모양 코덱	단위	무손실	년도
계수 스케일링 기반 ROI 코딩 방법	① Maxshift	1	임의모양	정적	X	X	X	WC	○	2000
	② Scaling based	2	직사각형,타원	정적	○	○	○	WC	○	2000
	③ Maxshift-Like	1C	임의모양	정적	○	X	X	WC	X	2001
	④ (G)BbBShift	No	임의모양	정적	○	X	X	WC	○	2002
	⑤ PSBShift	No	임의모양	정적	○	○	X	WC	○	2003
	⑥ HBSHift	No	임의모양	정적	○	○	X	WC	○	2004
EBCOT 기반 ROI 코딩 방법	⑦ Implicit	1	다각형	동적	○	○	X	CB	○	2000
	⑧vProposed Implicit	1C	임의모양	동적	○	○	X	CB/WC	X	2005
	⑨ Flexible and dynamic	1C	다각형	동적	X	X	X	Precinct	○	2002
	⑩ Fast ROI Transcoding	1C	다각형	동적	○	○	X	Packet	X	2005
	⑪ Prioritized ROI	1C	다각형	동적	X	○	X	Packet	○	2004

전략을 사용하였다. 따라서 시각적으로 무 손실 응용에 적합하다.

3.2 ROI 코딩 방법 선택 기준

최적의 ROI 코딩 방법의 선택 문제는 적용할 응용의 요구사항을 선택된 ROI 코딩 방법이 얼마나 잘 충족시키는지 하는 문제와 동일하다. 본 연구에서는 ROI 응용에 대한 대표적인 요구사항인 대화형 유무, ROI 중요도, 실시간성, 다수 ROI를 고려하여 요구사항에 따른 ROI 코딩 방법 분류표를 만들었다<표 2>.

대부분의 응용에서는 정적 ROI 코딩으로도 충분하지만, 인코딩 시에 ROI를 알지 못하는 경우에는 동적 ROI 코딩이 필요하다. 이러한 동적 ROI 코딩은 상호 대화형 응용에 유용하다. ROI 중요도는 ①⑨가 가장 높지만, 조절이 불가능하며, 그 외는 조절이 가능하다. 단 ⑩은 일부 BG 중요도 조절도 가능하다. 정적 ROI에서 실시간성은 ③이 BPs를 삭제함으로써 가장 우수하고, 다음으로 ①④⑤⑥이 우수하고, ②는 ROI 모양 코딩으로 인한 오버헤드 때문에 성능이 가장 떨어진다. 그리고 동적 ROI에서는 실시간성은 ⑨⑩이 재구성을 기반으로 Packet이나 Precinct 단위로 처리하기 때문에 가장 우수하고, 다음으로 CB단위로 처리하는 ⑦⑧이 우수하고, ⑪은 일부 BG 중요도 조절을 위한 처리로 인한 오버헤드 때문에 성능이 가장 떨어진다.

정적 ROI에서 다수 ROI는 ②⑤⑥은 다양한 등급으로 지원하며, ①③④는 동일한 등급으로 지원한다. 그리고 동적 ROI에서 다수 ROI는 ⑨만 동일한 등급으로 지원하고 나머지는 모두 다양한 등급으로 지원한다.

<표 2> 요구사항에 따른 ROI 코딩 방법 분류

요구사항	정적 ROI	동적 ROI
ROI 중요도	①, ②③④⑤⑥	⑨, ⑦⑧⑩⑪
실시간성	③, ①④⑤⑥, ②	⑨⑩, ⑦⑧, ⑪
다수 ROI	②⑤⑥, ①③④	⑦⑧⑩⑪, ⑨

3.3 ROI 매개변수

ROI 코딩 성능에 영향을 미치는 매개변수로는 타일 크기, 웨이브렛 필터 타입, 품질 레이어 수, CB 크기, ROI 수, ROI 크기, ROI 모양과 위치, ROI 중요도가 있다. 타일 크기는 작을수록, DWT 분해레벨의 수와 CB 크기가 작아지면서 블로킹 현상(Blocking artifacts)이 발생하기 때문에 ROI 코딩 성능을 감소시킬 수도 있다.

ROI 코딩을 위해서는 반드시 다수의 품질 레이어가 사용되어야 한다. 만약 하나의 레이어로 이미지를 인코딩하면, ROI 코딩의 이점은 전혀 없게 된다. 그리고 무손실 코딩에서 여러 레이어(50 레이어까지)를 사용하여 발생하는 오버헤드는 무시해도 좋다. 그러나 이 오버헤드는 낮은 비트율에서 증가한다[1].

웨이브렛 필터 타입은 DWT시 사용되는 WC 수와 관계가 있기 때문에 효율적인 ROI 크기에 영향을 미칠 수 있다. 그리고 분해레벨 수가 더욱 많이 사용될 때 그 영향력은 더욱 커진다. 또한 웨이브렛 기저함수의 중첩성으로 인하여, ROI WC는 전체 해상도에서 다수의 샘플들에 영향을 미칠 수 있다. 그래서 ROI 계수들에 대한 재구성은 ROI에 인접한 BG에도 포함될 것이다. CB 크기가 작으면 CB 수가 증가하여 많은 CBs를 코딩해야 하는 단점이 있지만, 반면에 CB의 공간적인 지역성이 뛰어나다는 장점도 있다. 이러한 상황은 CB 단위로 ROI 처리를 하는 EBCOT 기반의 방법에서 더욱 뚜렷하게 나타날 것이다.

ROI 크기의 변화에 따라 완전한 ROI를 처리하는 비트율은 낮아지고, ROI 정제속도는 빨라진다. 계수 스케일링 기반은 WC 레벨에서 ROI 코딩이 이루어지기 때문에 ROI 모양과 위치는 덜 중요하다. 그러므로 이 방법들은 ROI 크기가 작거나, 복잡한 모양의 ROI를 인코딩할 때 유용하다. 그리고 EBCOT 기반은 CB 단위로 ROI 코딩이 이루어지기 때문에 다각형 모양의 ROI를 인코딩할 때 유용하다.

중요한 포커스가 2개 이상이면, 같은 중요도나 서로 다른 중요도를 가지는 다수의 ROI가 요구된다. 이처럼 ROI 개수 선택은 ROI 크기와 위치 매개변수에 영향을 준다.

ROI 중요도는 ROI에서 BG로 자연스럽게 전이할 수 있도록 한다. ROI 중요도의 변화에 따라, ROI와 BG 비트율 간에 tradeoff가 존재한다. 그리고 낮은 비트율에서 높은 ROI 중요도를 가질 경우, BG의 품질이 너무 떨어져서 전체 이미지 품질을 떨어뜨리는 결과를 초래할 수 있다. 이 문제를 보완하기 위해서 대부분의 ROI 코딩 방법에서는 DWT 분해 레벨의 낮은 서브밴드들에 대해서 ROI 중요도를 적용하고 있다.

4. 실험 및 평가

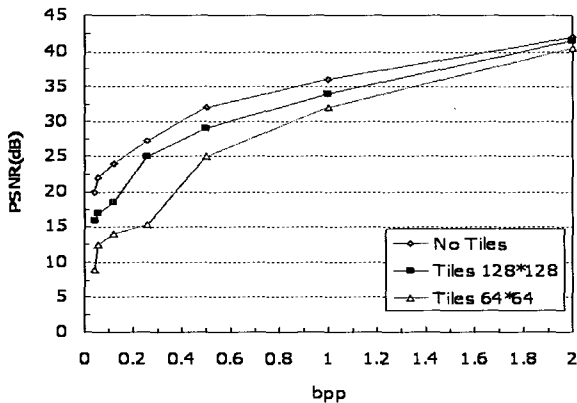
본 장에서는 표준 ROI 코딩 방법을 이용하여 ROI 매개 변수에 대한 효과를 정의하기 위하여 경험적인 실험을 한다. 실험 이미지는 <표 3>과 같다.

<표 3> 실험 이미지

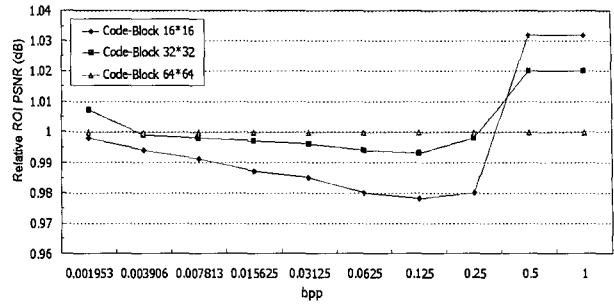
이미지	해상도	타입
barb	512*512	자연이미지
mountain	640*480	자연이미지
library	464*352	복합이미지
boat	512*512	자연이미지
satellite	1417 * 793	항공이미지
france	672*496	복합이미지

모든 이미지는 그레이 이미지(8 bpp)이며, 만약 Layered progressive 상태라면, 비트율이 0.03125에서 2 bpp로 생성 및 복원된다. 이것은 압축률이 256:1에서 4:1사이가 된다. 이 이미지 손실은 PSNR로 측정하였다. 그 외 매개변수들은 5 레벨 DWT, 가장 낮은 레벨을 ROI에 포함, 20 레이어, 그리고 나머지 매개변수들은 디폴트로 설정하였다.

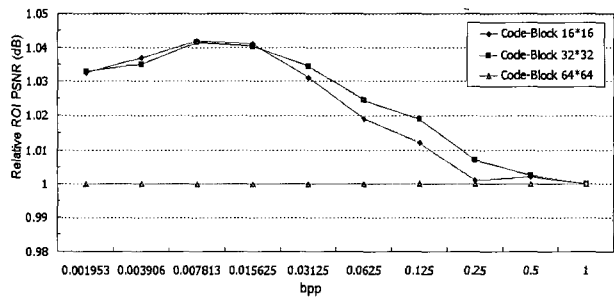
(그림 6)은 타일 크기의 변화에 따른 성능 평가를 나타낸다. (그림 6)처럼 타일 크기가 작을수록 압축 효율이 떨어졌다. 특히 저비트율(<1bpp)에서 하나의 타일로 이미지를 압축한 것에 비해 128*128 타일인 경우에는 약 5dB 감소하였고, 64*64 타일인 경우에는 약 10dB 감소하였다.



(그림 6) 타일 크기의 변화에 따른 성능 평가



(a) Maxshift Method



(b) Implicit Method

(그림 7) CB 크기의 변화에 따른 성능 평가

(그림 7)의 (a)와 같이 Maxshift 방법에서는 저 비트율일 때 작은 CB를 사용하는 것 보다 큰 64*64 CB를 사용할 때 ROI 성능이 우수했다. 그리고 작은 CB 크기를 사용할 경우, 비교적 높은 비트율에서부터 성능이 좋아지기 시작한다. 왜냐하면 ROI의 공간적인 지역성을 이용한 코딩 오버헤드보다 증가하는 CB 수를 코딩하는 오버헤드가 더 높기 때문이다. 그러나 높은 비트율에서 ROI는 CB 크기에 상관없이 시각적으로 무손실이다.

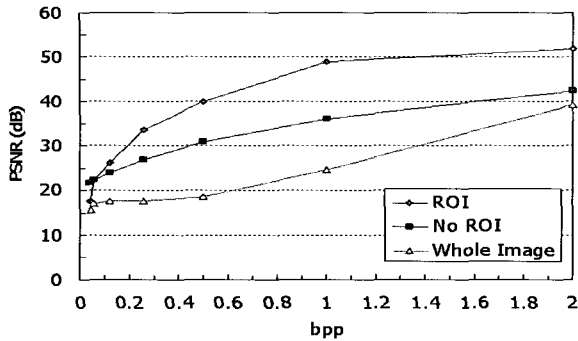
(그림 7)의 (b)와 같이 Implicit 방법은 CB 크기가 32*32 일 때 성능이 우수하다. 왜냐하면 Implicit 방법은 CB 단위의 ROI 처리를 함으로서 증가하는 CB 수를 코딩하는 오버헤드가 비교적 낮기 때문이다.

(그림 8)은 ROI 크기의 변화에 따른 성능 평가를 나타낸다. 실험 환경은 Maxshift 방법을 사용하여 이미지의 한 중앙에 ROI를 위치시키고, ROI의 크기는 각각 전체 이미지의 1/4, 1/8, 1/16로 하고, 사각형 ROI 모양에 32*32로 설정하여 실험하였다. ROI 크기가 작을수록 전체 ROI를 받는 비트율은 낮아지고, ROI 정제속도는 빠르게 나타났다. 특히 큰 ROI(>1/8)를 가지면서 저비트율(<0.125bpp)에서는 ROI의 PSNR이 BG보다 더 낮게 나타났다. 그 이유는 ROI 코딩에 대한 오버헤드 때문이다.

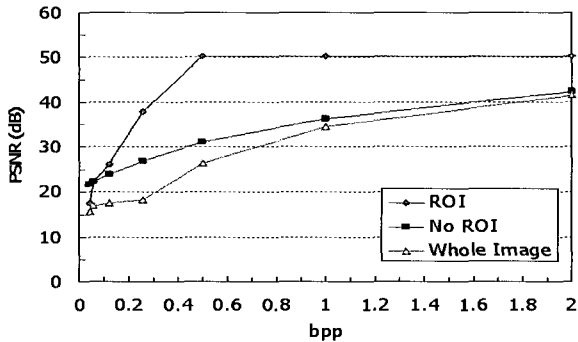
(그림 9)는 Implicit와 Maxshift 방법의 ROI 중요도 변화에 따른 ROI와 BG의 성능평가를 나타낸다. JPEG2000은 ROI 코딩을 하지 않았기 때문에 ROI와 BG간에 PSNR 성능 차이는 거의 없다. ROI 코딩에서 ROI 중요도가 높을수록 ROI와 BG간에 PSNR 성능 차이도 커진다.

Implicit 방법은 높은 ROI 중요도일 때가 높을수록 저 비트율에서는 ROI의 PSNR 간에 성능 차이는 적고, 반면에

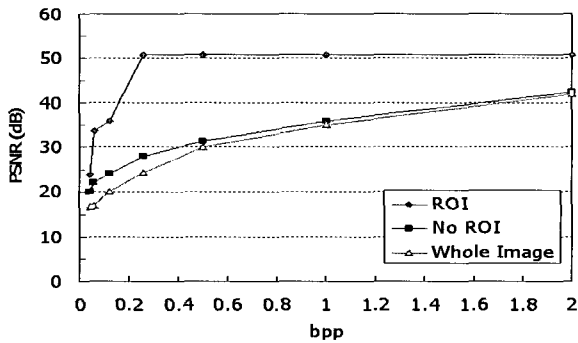
BG의 PSNR 간에 성능 차이는 크다.



(a) 1/4의 사각형 ROI, 32*32 CB

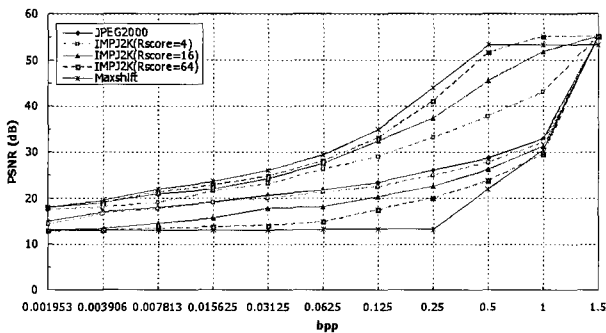


(b) 1/8의 사각형 ROI, 32*32 CB

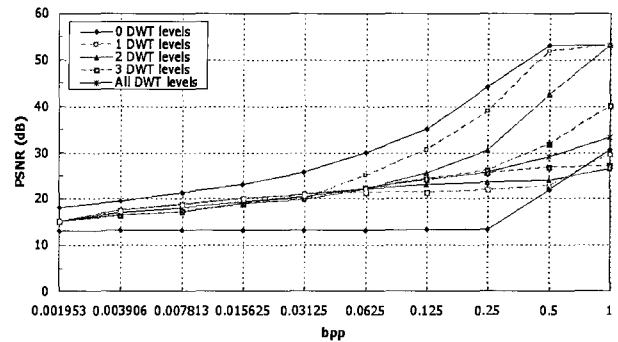


(c) 1/16의 사각형 ROI, 32*32 CB

(그림 8) ROI 크기의 변화에 따른 성능 평가



(그림 9) ROI 중요도의 변화에 따른 성능 평가



(그림 10) ROI 중요도로 적용되는 분해레벨의 수에 따른 성능 평가

(그림 10)은 ROI 중요도로 적용되는 분해레벨의 수에 따른 성능평가를 나타낸다. ROI 중요도로 코딩되는 낮은 분해레벨의 서브밴드들의 수가 증가할수록, 재구성될 BG 품질이 높아진다. 그러나 ROI 성능은 높은 비트율에서 영향을 받는다. 극단적으로 모든 해상도 레벨들이 ROI로 포함되면 ROI 코딩을 하지 않은 것과 같다.

지금까지 ROI 매개변수에 대한 실험을 통해 ROI 코딩에 대한 몇 가지 요점들을 정리하면 다음과 같다.

- ROI 중요도에 따라 ROI 품질과 BG 품질 간에 tradeoff가 존재한다.
- 만약 메모리 제약 문제가 없으면, 타일링은 효율적인 ROI코딩 방법은 아니다.
- 저 비트율이 아니면 여러 품질 레이어 사용으로 인한 오버헤드는 무시해도 좋다.
- ROI 코딩 효율성은 비트율, ROI 크기, 그리고 ROI 중요도에 의존한다. 특히 32*32 CBs는 ROI가 크고, 비트율을 높거나, BG가 시각적으로 중요한 경우에 사용되어야 한다. 반면에 16*16 CBs는 ROI가 작고 비트율이 낮거나 혹은 ROI가 시각적으로 중요한 경우에 사용되어야 한다.
- 일반적으로 Maxshift와 같은 계수 스케일링 기반의 방법에서는 CB 크기가 64*64일 때 성능이 우수하고 Implicit와 같은 EBCOT 기반의 방법에서는 CB 크기가 32*32일 때 우수하다.
- 큰 ROI(>1/8)를 저비트율(<0.125bpp)로 코딩하는 경우 ROI의 품질이 BG보다 오히려 떨어진다.
- 계수 스케일링기반은 얼굴 이미지와 같은 비트 스트림 내에 포함된 임의의 ROI 모양을 요구하는 응용에 적합하다. 반면에 EBCOT 기반은 서버/클라이언트 응용과 같은 압축된 비트 스트림으로부터 임의의 ROI를 추출하는 것을 요구하는 응용에 적합하다.

5. 결론 및 향후과제

모든 응용에 적합한 최적의 ROI 코딩 방법은 존재하지 않는다. 따라서 특정 응용의 요구사항을 만족시키는 적절한

코딩 방법과 매개변수 값의 선택이 필요하다. 본 연구에서는 요구사항에 따른 ROI 코딩 방법 분류표와 ROI 매개변수들에 대한 실험을 통하여 JPEG2000 응용에서 ROI 코딩 방법을 적용하기 위한 적절한 매개변수 선택 방법을 제시하였다. 또한 ROI 코딩에 대한 몇 가지 요점들도 제시하였다. 향후 과제로는 더 많은 ROI 코딩 방법의 실험과 두 개 이상의 조합된 매개변수를 고려하여 실험하는 것이다. 그리고 제안된 방법을 통해서 실제로 응용에 적용시키는 것이다.

참 고 문 헌

[1] M. Rabbani and R. Ioshi, "An Overview of the JPEG2000 Still Image Compression Standard", Signal Processing : Image Communication, Vol.17, pp.3-48, 2002,

[2] C. Christopoulos, A. Skodras and T. Ebrahimi, "The JPEG2000 Still Image Coding System: An overview", IEEE Transactions on Consumer Electronics, Vol.46, pp.1103-1127, Nov., 2000

[3] R. Grosbois, D. Santa-Crus and T. Ebrahimi, "New Approach to JPEG2000 Compliant ROI Coding", Proc. of the SPIE, Aug., 2001

[4] H. Yang, M. Long and H. M. Tai, "Region-of-Interest Image Coding Based on EBCOT", IEE Proceedings-Vision, Image, and Signal Processing, Vol.152, Issue.5, pp.590-596, Oct., 2005

[5] Z. Wang, A. C. Bovik and Fellow, "Bitplane- by-Bitplane Shift(BbBSHift) - A Suggestion for JPEG2000 Region of Interest Image Coding", IEEE Signal Processing Letters, Vol.9, No.5, May 2002

[6] Z. Wang, S. Banerjee, B. L. Evans and A. C. Bovik, "Generalized Bitplane-by-Bitplane Shift Method for JPEG2000 ROI Coding", IEEE ICIP, 2002

[7] L. Liu, and G. Fan, "A New JPEG2000 Region-Of-Interest Image Coding Method: Partial Significant Bitplanes Shift", IEEE Signal Processing Letters, Vol.10, No.2, Feb., 2003

[8] L. B. Zhang, and K. Wang., "New Approach for JPEG2000 Region Of Interest Image Coding : Hybrid Bitplane Shift", Proceedings of the 3rd Int'l Conference on Machine Learning and Cybernetics, Shanghai, Aug., 2004

[9] R. Rosenbaum, and H. Schumann, "Flexible, Dynamic and Compliant Region Of Interest Coding in JPEG2000", IEEE ICIP, Sep., 2002

[10] H. S. Kong, A. Vetro, T. Hata and N. Kuwahara, N., "Fast Region-of-Interest Transcoding for JPEG2000 Images", Mitsubishi Electric Research Laboratories, Inc., 2005

[11] V. Sanchez, A. Basu, and M. K. Mandal, "Prioritized Region Of Interest Coding in JPEG2000", IEEE Transactions on Circuits and Systems for Video Technology, Vol.14, No.9, Sep., 2004.

[12] G. K. Anastassopoulos and A. N. Skodras, "JPEG2000 ROI Coding in Medical Imaging Applications", <http://www.upatras.gr/ieee/kodraspubs/ans-c50.pdf>

[13] D. Santa Crus, T. Ebrahimi, M. Larsson, J. Askelof and C. Cristopoulos, "Region Of Interest Coding in JPEG2000 for Interactive Client/Server Application", IEEE MMSP, pp.389-394, Sep., 1999.



강 기 준

e-mail : kijuny@empal.com
 2000년 경상대학교 컴퓨터학과(학사)
 2002년 경상대학교 컴퓨터교육과(석사)
 2003년~현재 경상대학교 컴퓨터학과
 박사과정
 관심분야: MPEG, JPEG2000, 원격교육



이 부 권

e-mail : bklee@gsnu.ac.kr
 1972년 경상대학교 농경제학과(학사)
 1978년 미시건주립대학(석사)
 2001년 경남대학교 컴퓨터공학과(박사)
 1980년~현재 경상대학교 컴퓨터학과
 교수
 2001년~현재 경상대학교 컴퓨터정보통신연구소원
 관심분야: 시뮬레이션, 멀티미디어, 원격교육



서 영 건

e-mail : young@gsnu.ac.kr
 1987년 경상대학교 전산통계학과(학사)
 1989년 숭실대학교 전자계산학과(석사)
 1997년 숭실대학교 전자계산학과(박사)
 1989년~1992년 삼보컴퓨터
 1997년~현재 경상대학교 컴퓨터교육과
 부교수
 2001년~현재 경상대학교 컴퓨터정보통신연구소원
 관심분야 : 멀티미디어통신, 영상인식, JPEG2000