

울왜곡 최적기법을 이용한 JPEG-2000의 에러강인 엔트로피부호화

정회원 한성욱*, 정회원 최윤식**

JPEG-2000 Based Error Resilient Entropy Coding Using Rate-Distortion Optimization

Seong Wook Han*, Yoonsik Choe** *Regular Members*

요약

본 논문에서는 채널용량이 제한되고 에러에 민감한 무선환경을 고려하여 압축율이 기존의 방식보다 크고, 에러에 강인한 JPEG-2000 압축 표준을 기반으로 하여 새로운 EREC 기법을 적용한 에러강인 부호화를 제안하였다. JPEG-2000의 웨이브렛 변환 기반 내장형부호화기법(EBCOT)이 가지고 있는 슬롯 당 고정 비트율은, 울왜곡(R-D) 최적화에 의하여 가변 비트율로 변환하였고 여기에 EREC 기법을 적용하였다. 모의실험은 제안된 방법이 기존의 채널 코딩과 블록 경계 표시 헤드를 이용한 기존방법보다 특히 낮은 비트율에서 더욱 좋은 에러보정 능력을 보이는 것을 검증하였다.

Key Words : JPEG-2000; EREC; Error Resilient Coding

ABSTRACT

In this paper, a new improved EREC based error resilient coding in JPEG-2000 standard is proposed, considering the error sensitive wireless environment with limited channel capacity. In order to apply EREC, we use the variable bit-rate by using R-D optimization. Simulation results demonstrate that the proposed EREC based error resilient coding is more resilient than the error resilient schemes used in JPEG-2000.

1. 서론

이동 통신이 급속도로 발전하는 현 상황에서 휴대용 단말기에서의 멀티미디어 서비스가 많이 개발되고 있다. 그 중에서도 영상의 전송은 필수적인 기술이 되었다. 하지만, 채널의 상태가 안 좋은 경우 기존의 부호화 기법으로는 만족할 만한 성능을 기대할 수 없는 상황이었어서, 무선 통신 채널과 같이 에러가 많은 환경에서는 안정적으로 영상을 전송하는 것은 어려운 일이다. 이와 같이 부호화된 영상을 전송하는 경우, 채널의 상태가 좋지 않으면 높은 정확도의 소스 코딩이 더 부정확한 결과를 낳는다는

연구 결과는 알려진지 오래이다. 이들 연구는 DPCM[1]이나 블록 기반 DCT를 수행하고 부호화해야 할 차이 값이나 DCT 계수들을 양자화하는 경우, 채널의 상태와 양자화 간격이 복원된 영상에 미치는 영향에 대한 연구 결과를 제시하였다. 그 결과 채널이 열악할수록 낮은 비트율의 양자화의 성능이, 높은 비트율의 세밀한 양자화기보다 우수한 결과를 보인다는 결론을 얻었다.[3-6] 위 결과들은 모두 엔트로피 부호화 부분을 고려하지 못한 것이기는 하지만, 엔트로피 부호화를 추가하더라도 그 결과는 비슷하다.

이러한 엔트로피 부호는 비트열의 길이를 부호화

* * 온타임텍(주) 멀티미디어연구소(suphan@ontimetek.com),

** 연세대학교 전기전자공학부(yschoe@yonsei.ac.kr),

논문번호 : 040042-0202, 접수일자 : 2004년 2월 2일

다 다르게 정하기 때문에 채널의 오류에 의해서 동기를 잃고 심하게 정보가 손상될 수 있다. 그러나 에러강인 엔트로피 부호화 (EREC)알고리즘[11-13]을 이용해서 부·복호화를 하게 되면 에러가 있는 블록 내에서도 에러가 발생하기 이전의 정보에 대해서는 버리지 않고 이용할 수 있게 되며, 에러강인 엔트로피 부호화 알고리즘의 특성에 의해서 특별한 부가 정보나 채널 부호화를 하지 않더라도 매 블록마다 재동기가 가능하게 되어서 오류가 발생을 하더라도 채널 에러를 효율적으로 고립화 할 수 있다.

본 논문에서는 에러강인 엔트로피 알고리즘을 이용하여 멀티미디어 환경에 새롭게 대응하여 표준화된 JPEG2000을 기반으로 하는 내장형 부호기로 정지영상을 압축하여 에러에 강건하면서도 압축효율이 높은 영상압축을 수행하였다.

II. 에러 강인 엔트로피 부호화(EREC)

1948년 쉐논의 논문[2]이 발표된 이래, 현재까지 통신 시스템의 설계에서는 소스 부호화와 채널 부호화가 분리되어 이루어져 왔다. 그러나 시스템의 복잡도를 무한히 증가시킬 수 있는 이상적인 경우, 이와 같은 개별적인 최적화 작업은 전체 시스템의 최적화와 같은 의미를 가질 수 있지만, 시스템 복잡도에 제한이 있는 실제 시스템의 경우에는 그렇지 못하다. 따라서, 전체 시스템의 성능 개선을 위해 이 두 부호기를 함께 고려할 필요가 있다.

블록 기반의 영상 부호화에서는 대부분 엔트로피 부호화를 사용하게 된다. 그러나, 이 방법은 효율적인 부호화는 가능하게 하지만, 채널 오류에 대해서는 오류 전파 현상 등에 의해서 매우 취약한 특성을 가진다. 이 문제를 해결하는 방법 중의 하나가 에러강인 엔트로피부호화 알고리즘이다. 즉 각각의 길이가 다른 블록 정보를 정해진 길이의 슬롯에 재배열하는 방법을 통해서 효율적으로 오류의 전파를 막을 수 있는 방법이다.

2.1 에러강인 엔트로피 부호화 알고리즘

에러강인 엔트로피 부호화 알고리즘은 일반적으로 널리 쓰이는 압축 방법으로 생성된 가변장 자료를 비트 오류에 강한 구조에 재배치한다. 각각의 블록을 복호화기와 부호화기가 모두 알고 있는 위치에서 시작하게 배치하는 방식을 사용하는데, 그로 인해서 매 블록마다 자동적으로 동기화가 되는 효과를 얻을 수 있다. 또한 특별히 추가되는 부가 정

보가 필요 없다는 장점이 있다.

에러강인 엔트로피 부호화 구조는 길이가 s_i 인 N개의 슬롯으로 구성된다. 따라서 전송은

$$T = \sum_{i=1}^N s_i \text{ 비트만큼 된다. 중요한 점은 부호화 단}$$

과 복호화 단에서 모두 T, N, s_i 의 값을 알고 있어야 부·복호화가 가능하다. 그래야 N개의 슬롯을 어떠한 재동기 신호를 추가하지 않고도 연속적으로 전송할 수 있다. 각각의 EREC 프레임에서는 N개의 가변장 블록을 전송하게 되는데, 매 블록의 길이가 b_i 라고 하면 그 길이의 합은 T보다는 작아야 한다.

$$R = T - \sum_{i=1}^N b_i \geq 0 \quad (1)$$

식(1)에서 R은 불필요하게 전송하게 되는 비트수이다. 따라서 T의 값을 R을 최소화 할 수 있는 적당한 값을 선택을 하는 것이 중요하다. 그리고 선택된 T는 오류에 의해서 손상되면 전 프레임의 복호화가 불가능하기 때문에 반드시 오류 정정 부호로 전송을 하여야 한다. 그리고 일반적인 동영상 부호화에서는 영상의 크기가 이미 정해져 있기 때문에 블록의 개수 N도 따로 전송할 필요가 없다. 또한 T, N의 값을 이용하면 s_i 를 계산할 수 있기 때문에 따로 전송할 필요가 없다. 또한 s_i 를 두 가지 값으로 선택을 하면 $R = 0$ 으로 맞출 수 있다. $s_{opt} = \lceil T/N \rceil$ 라고 할 때에 부호화에 필요한 비트 수는 다음의 식으로 구할 수 있다.

$$T = a \times s_{opt} + (N-a) \times (s_{opt} + 1) = -a + N(s_{opt} + 1) \quad (2)$$

식(2)는 a개의 슬롯은 길이가 s_{opt} 이고, 나머지 (N-a)개의 슬롯은 길이가 $(s_{opt}+1)$ 로 생각하고 구한 식이다. 식(2)를 풀면, 다음과 같이 경계가 되는 블록의 번호를 구할 수 있다.

$$a = N(s_{opt} + 1) - T \quad (3)$$

구한 결과를 적용하면 처음부터 a번째 슬롯은 $s_i = s_{opt}$ 이고, (a+1)번째 슬롯부터 마지막 슬롯까지는 $s_i = s_{opt} + 1$ 이 된다. 이렇게 하면 식(1)의 R값은 0으로 맞출 수 있다.

EREC는 가변장 블록들을 비트를 재 정렬함으로써 EREC 구조에 맞추어서 넣게 된다.

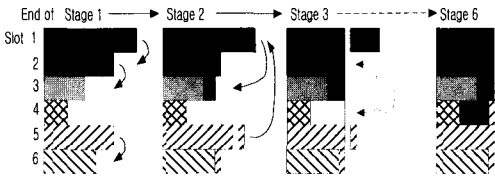


그림 1. 에러강인 엔트로피 부호화의 비트 재 정렬 알고리즘

그림 1에 알고리즘의 간단한 예가 나타나 있다. 첫 번째 단계에서는 매 블록의 자료를 해당하는 슬롯에 배치한다. 배치 할 때에는 최대한 많은 비트를 배치하게 된다. 따라서 $b_i = s_i$ 인 경우에는 그 블록과 슬롯은 완전히 부호화된 것이다. 그리고 $b_i < s_i$ 인 경우에는 블록은 부호화가 끝나고, 해당 슬롯은 $s_i - b_i$ 비트만큼 비어있게 된다. 마찬가지로, $b_i > s_i$ 인 경우는 해당 슬롯은 부호화가 끝난 것이고, 그 블록은 $b_i - s_i$ 비트만큼 더 부호화해야 할 내용이 남게 된다. 그러므로 첫 번째 단계가 끝나고 나면 1, 2, 5번 슬롯은 다 차게 된다. 다음 단계부터는 더 부호화할 비트가 남아 있는 블록들은

$$i + \phi_n(\text{modulo } N) \quad (4)$$

로 정해지는 순서대로 남은 공간이 있는 슬롯을 찾게 된다. 식(4)에서 i 는 현재 부호화하는 블록의 번호이고, n 은 진행 중인 단계의 수이다. ϕ_n 은 미리 정해진 찾는 순서이다.

오류가 없는 채널 환경이라면, 복호화 단계에서는 이와 같은 알고리즘을 그대로 사용하여서 복호화할 수 있다. 모든 허프만 부호들은 접두 부호 성질에 의해서 그 부호의 끝을 알 수 있고, 모든 매크로 블록은 마지막 부호가 있기 때문에 그 블록의 끝의 위치를 알 수 있다. 이렇게 하면 복호화가 끝난 블록에 해당하는 슬롯에서는 비어있는 공간의 위치를 알 수 있으므로, 다음 블록의 복호화가 가능하게 된다.

III. JPEG-2000의 내장형 부호기

3.1 JPEG-2000 부호화 방식

그림 2는 JPEG-2000의 블록도이다. 입력 영상은 먼저 웨이블릿 변환을 통해 먼저 변환된다. 정수, 고정소수점 또는 부동 소수점 웨이블릿 변환[14]이 사용되고 다른 웨이블릿과 웨이블릿-패킷 변환도 지원된다.

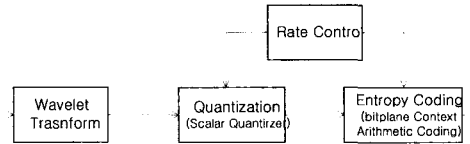


그림 2. JPEG-2000의 부호기 구조도

변환 계수들은 영점 스칼라 양자화기 또는 트렐리스 양자화기를 사용하여 양자화 한다. 비트 평면에 기반을 둔 산술 부호화는 양자화된 웨이블릿 계수들을 부호화하는 데 사용된다. 이런 계수들은 해상도, 질적인 비트 평면 또는 비트 평면 순서에 따라서 점진적으로 부호할 수 있다.

JPEG-2000에서는 EBCOT 부호 알고리즘을 사용하여 전체를 블록 기반으로 나누어 엔트로피 부호 작업을 한다. 블록 기반 엔트로피 부호기는 부대역 내에 인공적으로 정의된 블록들의 경계의 산술적 코딩 모델과 통계들을 초기화한다. 부대역 블록에 포함된 모든 계수들을 독립적으로 처리한다. 즉 해당하는 비트 스트림을 생성한다. 이런 방법은 블록에 포함된 모든 웨이블릿 계수들이 동시에 사용 가능하다는 것을 가정한 것이다. EBCOT 엔트로피 코더는 코딩 시스템의 복잡성을 줄이는 효과가 있다. 내장형 질적 확장을 지원하기 위해 블록 기반 엔트로피 부호기는 나중에 압축하고 다시 정렬하는 단계를 필요로 한다. 이런 추가적인 단계는 비율을 제어기와 결합할 때 특히 계산적으로 요구된다. [7-10],

3.2 내장형 부호기의 특징

MPEG-2, MPEG-4, H.263, JPEG과 같은 기존의 DCT 기반의 압축 표준들은 하나의 영상을 16×16 크기의 블록으로 나누고, 각각의 블록들을 서로 독립적으로 부호화하였다. 따라서, 첫 번째 블록을 완전히 복호한 후에, 두 번째 블록을 복호할 수 밖에 없었다. 이에 반하여 내장형 부호기는 전체 영상을 화질 또는 파일의 크기를 기준으로 나누어, 중요한 것부터 부호화한 후에 그보다 덜 중요한 부분은 나중에 부호화하는 방식을 사용하였다. 이런 방법은 사용자가 영상을 통신망으로 수신하고 있을 때, 원

하는 크기나 화질의 영상을 받았다고 생각하면 언제든지 통신망을 수신을 중단할 수 있다. 이런 부호화 방식을 내장형 부호화의 “점진적 부호화” 또는 “점진적인 전송”이라고 한다. 그림 3은 점진적 부호화의 예로써 그림 3(a)는 MPEG-2, JPEG과 같은 일반적인 블록기반의 부호화의 경우이고, 그림 3.(b)는 JPEG-2000에서 사용하는 내장형 부호화이다. W_0, W_1, \dots, W_N 는 부호화하고자 하는 심볼을 의미하고, b_1, b_2, \dots 는 각 심볼을 부호화하여 나온 비트들이다.

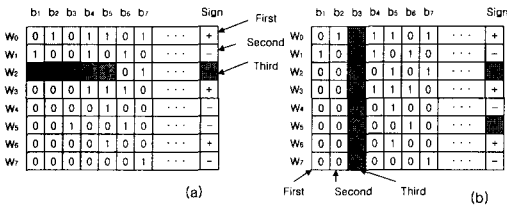


그림 3. 점진적 부호화의 예 (a) 일반적 부호기 (b) 내장형 부호기

그림 3.(a)를 통하여 블록기반 부호화는 W_0 심볼을 부호화한후, W_1 심볼을 부호화한다. 물론 복호기단으로 전송할 때도 W_0 의 비트들을 모두 전송하고 나서 W_1 심볼을 비트들을 전송한다. 이에 반하여 그림 3.(b)는 영상을 구성하고 있는 모든 심볼들의 첫 번째 비트들을 모두 전송하고, 두 번째 비트들을 전송한다. 두 기법의 궁극적인 차이는 복호단에서 일어난다. 복호단에서 블록기반 부호화는 영상의 모든 비트들을 수신해야만, 복호된 영상을 구성할 수 있다. 이에 반하여 내장형 부호기는 영상을 구성하고 있는 첫 번째 위치에 있는 영상의 모든 심볼들의 비트를 수신한 뒤, 두 번째 위치의 비트를 수신한다. 따라서 첫 번째 위치의 비트들을 받아 먼저 복호 영상을 구성한 뒤, 화질이 원하는 수준이 아니면 두 번째 위치의 비트들을 받아 보다 향상된 화질의 영상을 받을 수 있다. 따라서 신호를 보내는 공급자는 완전한 신호를 보내어도, 사용자 측에서 각자 원하는 수준의 영상을 판단하여 수신할 수 있는 기능이 가능하다.

위에서 언급한 점진적 전송은 기존의 블록기반 압축 표준과 내장형 부호화 사이의 가장 큰 차이점이다. 내장형 부호화는 지금까지 EZW[19], SPIHT[20], EBCOT[15][16] 등의 많은 기법들이 제안되었고, JPEG-2000만이 국제 표준으로 확정되

었다. JPEG-2000은 JPEG의 문제점들을 해결하기 위해 제안된 표준으로 본 연구에서 사용하는 부호기이다.

3.2 JPEG-2000의 에러강인 부호화 기법

JPEG-2000은 양자화된 웨이블릿 계수를 압축하기 위하여 가변 장 부호화를 사용한다. 가변장 부호화는 채널과 전송 오류에 민감한 것으로 잘 알려져 있다. 하나의 비트 오류도 엔트로피 부호기에서 동기를 잃게 만들어서 복원 영상에 심각한 손실을 줄 수 있다. 에러가 있는 채널을 통해 압축된 영상들을 전송할 때 효율을 높이기 위하여 에러에 강인한 비트스트림 구조와 도구들을 JPEG-2000 표준에서는 포함하고 있다. 에러강인 기법들은 다음의 방법들을 사용하여 채널 에러에 대처한다.

- 1) 중요 데이터 분할 (Data Partitioning)
- 2) 재동기 표시 (resynchronization)
- 3) 에러 검출 및 복원
- 4) 우선순위에 따른 서비스 품질 보장 전송

양자화된 계수들의 엔트로피 부호화는 코드 블록 내에서 수행된다. 각각의 코드 블록의 부호화와 복호화는 서로 독립적이기 때문에 코드 블록의 비트 스트림에 생기는 비트 오류들은 해당 코드 블록내에 한정된다. 에러에 강인함을 증가시키기 위하여 산술 부호화의 종료는 모든 부호화 과정 후에 허용되고, 산술 부호화의 문맥은 각각의 부호화 과정이 후에 리셋된다. 이것은 에러가 발생되더라도 산술 복호기가 복호 과정을 계속 할 수 있도록 만든다.

“느린 부호화” 모드는 또한 에러 강인화에 유용하다. 산술 부호화 없이 원래의 비트를 비트스트림으로 사용하는 선택적 산술 부호화 과정과 관련 있는데, 이것은 가변장 부호화가 민감한 오류 전파 속성을 방지한다.

패킷 수준에서는 재동기 표시를 가진 패킷에 공간적 분할과 재동기를 할 수 있게 한다. 패킷 헤더는 재동기 표시와 시퀀스 번호를 포함하고 타일안에 있는 모든 패킷들 앞에 위치한다. 따라서, 에러 강인화는 엔트로피 부호화 단계와 패킷 단계에서 수행될 수 있다.

IV. 에러에 강인한 영상정보 전송 시스템 설계

그림 4은 본 논문에서 제안하는 에러에 강인한 영상 정보 전송 시스템에 대한 구성도이다. 압축 표준은 기존의 정지 영상 부호화 기법 중에서 가장

압축 효율이 좋은 JPEG-2000을 사용하고 압축한 비트 스트림에 에러 강인성 기능을 추가하기 위하여 에러강인 엔트로피 복/부호기를 적용하였다. 즉, 그림의 소스 부호기 블록과 에러강인 엔트로피 블록은 본 논문에서 궁극적으로 구현하고 자 하는 에러에 강인한 정지영상 부호기이다.

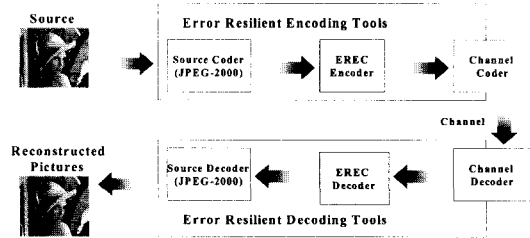


그림 4. 에러에 강인한 정지영상 부호기 설계 구성도

4.2 에러강인 엔트로피 구조를 가지는 JPEG-2000

JPEG-2000은 MPEG-2, H.263, JPEG과는 다르게 에러에 강인한 부호화를 기본적으로 지원하고 있다. 부호화 스트림에서 중요한 정보와 덜 중요한 정보들을 분리하여 데이터를 정렬하는 “데이터 분할” 기법과 에러의 전파를 막기 위해 사용하는 “재동기 표시” 기법들이 그 예이다. 그 중에서도 재동기 표시는 비트 스트림 데이터에 일정 간격으로 헤더를 덧붙여서 비트 에러 전파를 막는 것이다. 즉, 헤더를 많이 붙이면 붙일수록 에러에 강인한 특성을 가지게 되지만, 실제 영상 정보와는 관련 없는 부가 정보가 많이 생기기 때문에 압축 효율이 안 좋아지는 단점을 가진다. 따라서 헤더와 같은 부가정보를 덧붙이지 않고서도 에러의 전파를 막을 수 있는 기법이 바로 에러강인 엔트로피 부호화기법이다.

4.2.1 에러강인 엔트로피 부호화 구현 시 발생하는 문제점

위에서 언급한 것과 같이 에러강인 엔트로피 부호화를 적용하는 기준은 서로 다른 길이를 가진 슬롯이다. 슬롯마다 길이가 다르기 때문에, 슬롯 중간에 에러가 발생했을 때 다음 슬롯에 그 영향이 전파되어 에러 전파 현상이 발생하는 것이다. 이를 방지하기 위해 에러강인 엔트로피 부호화 알고리즘은 서로 다른 길이들로 구성된 슬롯을 적절히 재 정렬하여, 동일한 길이로 만들어 에러의 전파를 방지한다. 이런 이유로 에러강인 엔트로피 부호화를 적용하고자 할 때는 슬롯마다 길이가 모두 달라야 한다는 대전제를 만족해야한다.

일반적인 그림 5.(a)의 블록기반 부호기(JPEG이

나 MPEG등의 압축 표준)에서는 블록마다 길이가 모두 다르다. W_0, W_1, \dots, W_7 을 순서대로 부호화하기 때문에, 각 심볼들은 서로 길이가 다르고 또한 슬롯으로도 구성할 수 있다. 하지만 그림 5.(b)의 내장형 부호기는 비트 스트림을 구성할 때, 각 심볼들의 첫 번째 비트를 먼저 내보내고, 두 번째 심볼을 내보낸다. 즉, 그림 5.(b)에서와 같이 부호화를 b_1, b_2, b_3 의 순서로 수행한다. b_1, b_2, \dots, b_7 의 길이는 영상의 심볼의 개수만큼 모두 동일하다. 따라서 슬롯을 구성할 때 모든 슬롯의 길이가 동일하기 때문에, 에러강인 엔트로피부호화를 적용할 수 없다는 치명적인 약점을 가지게 된다.

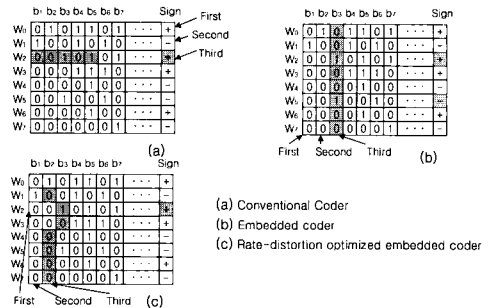


그림 5. 올웨곡 최적화 과정을 가지는 에러강인 내장형 부호화

4.2.2 올웨곡 최적화 과정을 가지는 에러강인 엔트로피 부호화

본 논문에서는 일반적인 내장형 부호기에서는 슬롯을 구성할 때 슬롯의 길이가 동일하기 때문에 에러강인 엔트로피 부호화를 적용할 수 없다는 문제를 R-D 최적 방법을 적용하여 해결하였다. 즉,

$$D = \sum_{i=1}^N D_{i, n(i)} \text{ s.t. } R_{Max} \geq R = \sum_{i=1}^N R_{i, n(i)}$$

$$\Leftrightarrow D(\lambda) + \lambda R(\lambda) = \sum_{i=1}^N [D_{i, n(i)} + \lambda R_{i, n(i)}]$$

(5, 6)

식 (5)는 부호화하고자 하는 영상의 목표 비트량이 R_{Max} 일 때, 프레임을 구성하고 있는 N 개의 심볼을 나타내는 비트 스트림을 각각 $n(i)$ 개의 비트만으로 부호화했을 때 생기는 왜곡, $\sum_{i=1}^N D_{i, n(i)}$ 를 최소로 만드는 $n(i)$ 의 집합을 찾는 것이다. 식(6)

은 식(5)의 제한조건의 최적해를 찾는 문제를 라그랑제 방법을 이용하여 변환시킨 식이다. 비트스트림을 심볼마다 비트 평면을 구성시킬 때 동일한 양만큼 할당하지 않고, 심볼마다 비트 평면을 구성하는 비트량을 다르게 함으로써, 화질개선 효과와 각각의 비트 평면을 구성하는 비트 수도 다르게 하는 역할을 한다. 따라서, 각각의 비트 평면을 슬롯으로 구성한다면 에러강인 엔트로피 부호화 알고리즘을 적용할 수 있으면서, 화질 개선 효과도 이룰 수 있다. 그림 6은 JPEG-2000 압축 표준에 율웨곡 최적 기법을 적용했을 때 비트평면을 구성하는 비트량을 표현한 것이다.

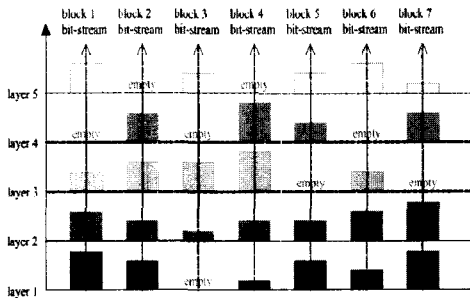


그림 6. JPEG-2000에서 비트 평면을 구성하는 각 심볼의 비트량 (율웨이 기법 적용시)

V. 실험 결과

실험 영상은 DCT 부호기의 경우에는 일반적으로 영상실험에 많이 쓰이는 lena 영상을 사용하고 가변장 길이 부호기에 대한 실험은 JPEG-2000 실험용 영상으로 ITU-T에서 제공한 hotel 영상과 goldhill 영상을 사용하였고 실험 채널은 비트당 에러 발생 비율을 변화시켜 다양한 환경 하에서의 부호기의 성능을 평가하였다.

5.1 EREC를 적용한 JPEG-2000 실험 결과

일반적으로 JPEG-2000 압축 표준에서 제공하는 에러 강인 기술 중 가장 널리 사용되고 쉬우면서도 가장 효과적인 방법은 재동기 표시 기술이다. 패킷마다의 경계를 나타내기 위해 오버헤드를 붙이고 패킷의 크기를 전송함으로써 에러 전파를 막는 방법이다. 하지만 재동기 표시 기술은 패킷마다의 동기를 유지하기 위해 오버헤드를 많이 붙이기 때문에 에러가 적게 생기거나 거의 없는 일반적인 환경

에서는 압축 효율이 많이 떨어진다. 또한 재동기 표시와 같은 오버헤드를 덧붙이지 않으면서 동기를 맞출 수 있는 기술이 있다면 에러에 강인하면서도 압축 효율이 높은 부호기가 될 것이다. 이와 같은 재동기 표시를 대체하는 기술이 에러강인 엔트로피 부호화 기법이다. 본 논문에서는 JPEG-2000의 원본 파일에 에러를 첨가한 경우와 재동기 표시를 덧붙인 경우와 EREC를 사용한 경우 3가지 경우에 대해 비교 분석하였다.(표 1)

표 1. 고정 비트율에서의 재동기 표시 방법과 에러강인 엔트로피부호화 방법의 비교

영상 이름	bpp	0.125		0.25		0.5	
		BER	Resync	EREC	Resync	EREC	Resync
Goldhill	0.0001	26.4175	26.9460	26.8443	27.2164	27.9133	28.1146
	0.00005	26.7414	27.3892	29.1804	30.3819	30.9476	31.1997
	0.00001	29.2249	29.4596	31.4439	31.6293	33.7444	33.7800
Hotel	0.0001	22.7017	24.5244	23.7007	24.6650	23.9980	24.8316
	0.00005	24.6289	24.7195	25.2237	26.0758	26.9928	28.6429
	0.00001	27.0581	27.2483	29.8826	30.1485	33.7705	33.9154

표 1은 0.125bpp와 0.25bpp, 0.5bpp로 압축 했을 경우 재동기 표시를 이용하여 헤더정보를 덧붙이는 경우와 EREC로 부호화한 경우의 성능비교 이다. 재동기 표시에는 EREC 부호기에는 없는 헤더를 많이 추가시키기 때문에 압축 효율이 떨어지는 것을 확인 할 수 있다.

그림 7부터 그림 10까지는 JPEG-2000에 에러 강인 기능을 첨가하지 않은 부호화와 재동기 표시를 첨가한 경우, EREC를 적용한 경우의 실험 결과이다. 에러 강인 기능을 전혀 적용하지 않은 경우, BER이 10^{-3} 보다 큰 경우에는 영상이 복원조차 되지 못했다. 그림 7과 그림 8은 PSNR을 통해 세 가지 경우를 객관적으로 비교한 결과로서 EREC의 성능이 다른 경우보다 좋은 것을 확인 할 수 있다. 이런 결과는 그림 9와 그림 10의 주관적 비교를 통해서도 확인 할 수 있다. 그림 9에서는 재동기 표시와 EREC 경우만을 비교하였는데 재동기 표시를 사용한 경우 이러한 표시 헤더조차 에러에 영향을 받은 경우에는 블록 전체가 깨지는 현상을 확인 할 수 있다. 그림 10에서는 4가지 경우를 주관적으로 비교한 경우인데 에러 강인 기능을 사용하지 않은 경우 에러 전파 현상이 발생한 것을 눈으로 확인 할 수 있다. 재동기 표시 경우에는 오버헤드로 인해

서 EREC와 비교했을 때 화질의 열화가 심함을 알 수 있다.

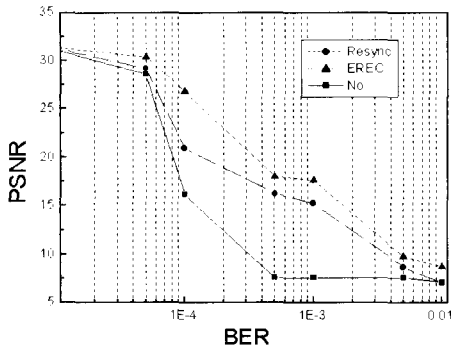


그림 7. goldhill 영상의 BER 변화에 따른 PSNR 비교

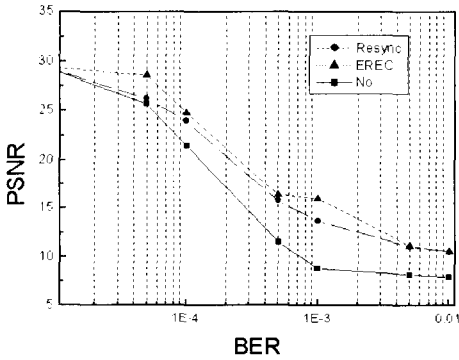
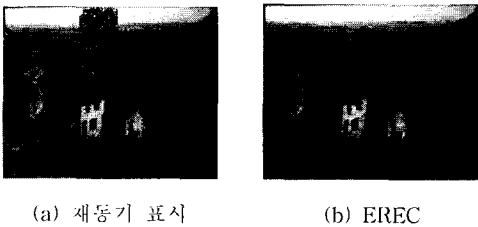


그림 8. Hotel 영상의 BER 변화에 따른 PSNR 비교



(a) 재동기 표시

(b) EREC

그림 9. BER = 10⁻⁴일때의 주관적 비교 (goldhill 영상)



(a) 에러 없는 hotel 영상

(b) 단순부호화 (에러첨가)



(c) Resync marker



(d) EREC

그림 10. BER = 10⁻⁴일때의 주관적 비교 (hotel 영상)

VI. 결론

본 논문에서는 무선 통신 채널의 영상 전송을 목적으로, 높은 압축 율을 가지면서도 신호에 오버헤드를 덧붙이지 않고 효과적으로 동기를 보장하는 기술인 EREC 기술을 에러강인 부호화를 위하여 적용하였다. 하지만 기존의 EREC 기술은 블록 기반의 DCT 가변부호화를 기반으로 연구되었기 때문에 JPEG-2000과 같이 SPIHT, EZW과 같은 웨이블릿 기반의 고정 비트율 내장형 부호기에는 적용할 수 없었다. 이에 본 논문에서는 율-제어 기법을 이용하여 내장형 부호기의 슬롯 당 비트율을 다르게 최적화하고 따라서 EREC를 JPEG-2000 부호기에 적용할 수 있는 방법을 제안하였다. 결과적으로 JPEG-2000의 경우 재동기 표시를 덧붙이지 않고도 에러에 강인한 부호기를 설계할 수 있게 되어 보다 나은 압축 효율을 얻을 수 있었다. 하지만 EREC의 가장 큰 단점은, 데이터를 에러에 강인하게 재 정렬하는 복부호에 재동기 표시에 비해 시간이 많이 걸린다는 점이다. 따라서 향후에는 복부호 시간을 단축하면서 보다 에러에 강인한 기법의 연구가 이루어져야 한다.

참고 문헌

- [1] A. Netravali, "Onquantizers for DPCM coding of picture signal,," IEEE Trans. Inform. Theory, vol IT-23, pp 360 - 370, 1977
- [2] C. E. Shannon, "A mathematical theory of communication", Bell Syst. Tech. J., vol.27, pp.379-423, 623-656, 1948
- [3] J. W. Modestino and D. G. Daut, "Combined Source-Channel Coding of Images,," IEEE Trans. Commun., vol. com-27, pp. 1644-1659, Nov. 1979.

- [4] J. W. Modestino, D. G. Daut and A. L. Vickers, "Combined Source-Channel Coding of Images Using the Block Cosine Transform," IEEE Trans. Commun., vol. com-29, pp. 1261-1274, Sep. 1981.
- [5] N. Farvardin and V. Vaishampayan, "Optimal Quantizer Design for Noisy Channels : An Approach to Combined Source-Channel Coding," IEEE Trans. Inform. Theory, vol. 33, pp. 827-838, Nov. 1987.
- [6] R. Chandramouli, N. Ranganathan, and S. J. Ramadoss, "Adaptive Quantization and Fast Error-Resilient Entropy Coding for Image Transmission," IEEE Trans. Circ. Syst. Video Technol., vol. 8, pp. 411-421, Aug. 1998.
- [7] W.B. Pennebaker and J.L. Mitchell, JPEG Still Image Data Compression Standard. New York: Van Nostrand Reinhold 1993.
- [8] ISO/IEC, "Information Technology - JPEG 2000 Image Coding System - Part 1", ISO/IEC FDIS15444-1, Aug. 2000
- [9] ISO/IEC, "JPEG 2000 Verification Model 8.5", ISO/IEC JTC 1/SC 29/WG1 WG1 N1878, Sept. 2000
- [10] G.Langdon, Jun, "An Introduction to Arithmetic Coding", IBM J. Res. Develop., vol. 28, pp 135-149, 1984
- [11] N. T. Cheng and N. G. Kingsbury, "The ERPC : An efficient error-resilient technique for encoding positional information on sparse data," IEEE Trans. Commun., vol. 40, pp. 140-148, Jan. 1992
- [12] D. W. Redmill and N. G. Kingsbury, "The EREC:An Error-Resilient Technique for Coding Variable-Length Blocks of Data", IEEE Trans. IP. vol. 5, no. 4, pp 565-574, April 1996
- [13] 최윤식, 김용구, 송진규, "EREC 기반 비트열을 위한 Source-Channel 결합 최적 양자화기 설계 및 이를 통한 안정적 영상 통신에 관한 연구", 한국통신학회 논문지, Vol.25, No.12B, pp.2094-2108, 2000. 12
- [14] M. Antonini, M. Barlaud, P. Mathieu and I. Daubechis, "Image coding using wavelet transform," IEEE Trans IP, vol.1, pp 205-229, April 1992
- [15] D. Taubman and A Zakhor, "Multirate 3-D subband coding of Video," IEEE Trans. IP, vol.3, pp 572-588, Sept. 1994
- [16] D. Taubman, "High Performance Scalable Image Compression with EBCOT", IEEE Trans. IP. vol.9 no.7, pp 1158-1170, July 2000
- [17] J. Li and Shawmin, "An Embedded Still Image Coder with Rate-Distortion Optimization", IEEE Trans. IP. vol.8 no.7, July 1999
- [18] E. Ordentlich, M. Weinberger, and G. Seroussi, "A low-complexity modeling approach for embedded coding of wavelet coefficients," in Proc. Data Compression Conf. 1998, J. A. Storer and M. Cohn, Eds., Mar 1998
- [19] J. M. Shapiro, "Embedded Image Coding Using Zerotress of Wavelet Coefficients", IEEE Trans. Signal Processing, Vol. 41 pp 3445-3462, Dec. 1993
- [20] A. Said and W. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchial trees," IEEE. Trans. CSVT, vol 6, pp 243-250, June 1996
- [21] T. Ebrahimi, "The JPEG2000 still image coding system:An overview," IEEE Trans. Cons. Electronics, vol 46, pp 103-1127, Nov. 2000
- [22] J. Ling and R. Talluri, "Tools for Robust Image and Video Coding in JPEG2000 and MPEG4 Standards," IS&T/SPIE Conf. on visual comm. San jose, vol 3653, Jan 1999
- [23] I. Moccagatta, S. Soudagar, J. Liang and H. Chen, "Error resilient coding in JPEG-2000 and MPGE4," IEEE journ. comm. vol 18, pp 899-914, June 2000

