

특집논문-08-13-5-07

## H.264/AVC 비트스트림으로부터 썸네일 추출 시 효율적인 오차 보상 방법

윤명근<sup>a)‡</sup>, 이여송<sup>a)</sup>, 손채봉<sup>a)</sup>, 박호종<sup>a)</sup>, 안창범<sup>a)</sup>, 오승준<sup>a),b)</sup>

### An Efficient Error Compensation Method for Thumbnail Extraction in H.264/AVC Bitstreams

Myung-Keun Yoon<sup>a)‡</sup>, Yeo-Song Lee<sup>a)</sup>, Chae-Bong Sohn<sup>a)</sup>, Ho-Chong Park<sup>a)</sup>,  
Chang-Beom Ahn<sup>a)</sup>, and Seoung-Jun Oh<sup>a),b)</sup>

#### 요약

최근에 H.264/AVC 표준을 이용하는 HDTV, IPTV와 같은 고화질 멀티미디어 서비스가 증가하고 있다. 이러한 고화질의 멀티미디어 서비스를 편리하게 이용하기 위해 축소 영상을 이용한 기술이 요구 된다. 그러나 H.264/AVC는 인트라 예측을 사용하기 때문에 종래 표준에 적용되던 주파수 공간에서 축소 영상을 추출하는 방법을 적용할 수 없다. 따라서 H.264/AVC 인트라 프레임으로부터 축소 영상을 주파수 공간에서 추출할 수 있는 방법이 제안되었다. 그러나 인트라 예측의 반올림 오차와 정수연산과 부동소수점 연산 사이에서의 불일치 문제로 인하여 썸네일의 화질을 저하시키는 문제점이 있다. 본 논문에서는 H.264/AVC 비트스트림으로부터의 썸네일 추출법에 대한 보상 방법을 제안한다. 제안하는 방법에서는 썸네일 추출 시 발생하는 오차에 대하여 분석하고, 통계적 모델을 통해 보상값을 제시한다. 실험결과를 통해 D1급 또는 HD급 영상에서 썸네일 영상의 화질을 효과적으로 보상함을 확인할 수 있으며, 속도 저하는 무시할 수 있다.

#### Abstract

Recently, high definition media services like HDTV and IPTV are growing. A fast reduced-size image extracting method is required to meet what those services require. Conventional DC image extracting methods, however, can't be applied to H.264/AVC streams since a spatial domain prediction scheme is adopted in H.264/AVC intra mode. To solve this problem, a thumbnail extraction method in H.264/AVC was proposed. However, the method has mismatch problem which was caused by round-off operation in intra prediction and mismatch between integer and floating point calculation. In this paper, we propose an error compensation method for extracting thumbnail directly in H.264/AVC bitstreams. The compensation method introduces the mismatch problem in thumbnail extraction and presents compensation values. Through the implementation and performance evaluation, proposed method compensated round-off error efficiently in D1 and HD sequences while the additional extraction time is negligible.

Keyword : Thumbnail extraction, H.264/AVC Intra frame, Thumbnail image

## I. 서론

최근의 디지털 동영상 처리 기술이 발전함에 따라 고화질 디지털 방송 (HDTV : High Definition Television), 인터넷 TV(IPTV : Internet Protocol Television)등과 같이 다양한 매체를 이용한 디지털 멀티미디어 서비스가 활성화되고 있다. 특히 고화질 디지털 방송이 일반화 되면서 다양한 서비스 어플리케이션이 개발되고 있으며, 고화질, 고해상도의 영상을 위한 고속 동영상 처리 기술들이 요구 되고 있다<sup>[1-3]</sup>. 비디오 데이터의 효율적인 보관 및 전송을 위해서는 MPEG(Moving Picture Experts Group)과 같은 동영상 압축을 사용해야 한다. 하지만 압축된 자료는 영상처리의 목적으로 다루기가 쉽지 않다. 압축된 자료의 효과적인 처리를 위하여 빠른 검색 및 복원을 위한 알고리즘에 대해 많은 연구가 진행되었다<sup>[4-6]</sup>. 이러한 연구 중의 하나가 축소 영상에 대한 연구이다. 축소 영상은 블록의 DC계수 값만으로 이루어진 동영상 시퀀스를 대표하는 DC 영상을 의미하며, 흔히 썸네일(Thumbnail)이라고 부른다. DC영상 혹은 썸네일은 영상의 특성을 대략적으로 표현하는데 필요한 전반적인 요소들을 지니고 있으므로 이를 이용하여 영상의 고속 검색이나 색인에 많이 이용된다. 또한 썸네일 영상의 추출은 완전한 크기로의 복원에 비해 보다 적은 메모리가 필요하며, 더 적은 연산 능력만으로 수행이 가능하므로 MPEG 기반으로 압축된 동영상의 효과적인 탐색에 사용되고 있다<sup>[7-10]</sup>. 블록단위의 DCT 부호화로 부호화된 동영상에서 축소 영상을 추출하는 대표적인 방법은 DCT 공간에서 DC 계수를 직접 추출하는 방법이다. 이 방법은 DCT 공간에서 DC 계수 값을 직접 추출함으로써 연산 량을 줄일 수 있다. 이러한 이유로 블록단위의 DCT 부호화 방법을 사용하는 종래의 비디오 표준에서 널리 사용되고 있다<sup>[11-15]</sup>.

H.264/AVC는 새로운 기술들을 채택함으로써 기존의 비

디오 압축 표준보다 월등히 높은 압축률을 제공한다<sup>[16]</sup>. 새롭게 채택된 기술중의 하나인 인트라 예측은 현재 블록을 예측함으로써 압축률을 높이는데 크게 기여한다. 하지만 인트라 예측으로 인해 기존의 국제 표준에 적용되는 DC영상 추출법을 적용할 수 없다. 따라서 이러한 필요에 부합하기 위해 DMB를 위한 요약영상 추출 알고리즘과 H.264/AVC의 비트스트림으로부터 썸네일을 빠르게 추출하는 방법이 제안되었다<sup>[17-18]</sup>. 제안된 H.264/AVC의 인트라 프레임으로부터의 썸네일 추출법은 DCT 공간에서 직접 썸네일을 추출하기 때문에 IDCT(Inverse DCT)과정과 화소 공간에서의 인트라 예측을 생략하여 연산을 줄일 수 있다. 뿐만 아니라 썸네일 추출을 위해 사용되는 연산은 수직/수평 여과기의 주파수의 변환특성을 이용하며 간단히 계산 가능하다. 그러나 H.264/AVC에서 사용하는 정수 연산과 제안된 방법의 부동소수점연산 사이에서 발생하는 불일치 문제와 화소 공간에서 수행하는 인트라 예측의 반올림 연산으로 인해 썸네일의 추출 과정에서 오차가 발생하여 영상에 누적되는 문제점이 있다.

본 논문에서는 H.264/AVC의 인트라 프레임에서 썸네일 추출시 발생하는 오차에 대한 보상 방법을 제안한다. H.264/AVC의 참조 복호화기로부터 썸네일을 얻는 방법과 H.264/AVC의 주파수 공간에서 썸네일을 추출하는 방법을 비교하여 오차가 발생하는 부분을 기술하고 오차들의 통계적 모델을 통해 보상값을 결정한다. 제안하는 보상 방법을 기존의 썸네일 추출 방법에 적용하면 큰 속도차이 없이 향상된 화질의 썸네일을 얻을 수 있다.

논문의 구성은 다음과 같다. 2장에서는 H.264/AVC의 비트스트림으로부터 썸네일을 추출하는 방법에 대하여 설명한다. 3장에서는 발생하는 오차에 대한 분석과 보상 방법을 제시한다. 4장에서는 제안한 보상 방법을 적용한 실험 결과를 보여주며, 5장에서는 결론을 맺는다.

## II. H.264/AVC 인트라 프레임에서 썸네일 추출 방법

H.264/AVC에서는 인트라 예측을 채택함으로써 인트라 프레임에서 기존의 비디오 압축 표준보다 높은 압축 효율

a) 광운대학교, VIA-멀티미디어 센터  
VIA-Multimedia Center at Kwangwoon University,  
b) 광운대학교, 방송통신융합플랫폼 센터  
BnC Convergence Platform Center at Kwangwoon University,  
‡ 교신저자: 윤명근(ymk81@media.kw.ac.kr)  
\* 본 연구는 2008년도 광운대학교 교내 학술연구비[60012008090]와 지식경제부와 한국산업기술재단의 전략기술인력양성사업과 정보통신연구진흥원의 차세대 DTV핵심기술개발사업의 일환으로 수행되었음

을 제공한다. 하지만 인트라 예측에서는 주변 블록과의 상관관계를 이용하여 현재 블록을 예측하기 때문에, 종래의 방법인 각 블록에서 독립적으로 썸네일을 추출하는 방법을 적용할 수 없다. 이번 장에서는 인트라 예측을 일반화 수식으로 정리하고, 일반화된 수식과 정의된 필터들을 통해 인트라 프레임에서 썸네일을 추출하는 방법을 기술한다.

1. 4x4 인트라 예측의 일반화

H.264/AVC의 인트라 예측은 현재 블록을 부호화 하기 위해 주변 블록의 정보를 이용한다. 인트라 16x16의 경우는 4가지 방향, 8x8, 4x4의 경우는 그림 1(b)와 같이 9가지 방향을 가지고 있다. 모드 2(DC) 경우에는 방향이 없기 때문에 표시 하지 않고 있다. 그림 1(a)에서 y는 현재 예측할 블록을 나타내며 x<sub>1</sub>, x<sub>2</sub>, x<sub>3</sub>, x<sub>4</sub>는 이전에 부호화 되어 복

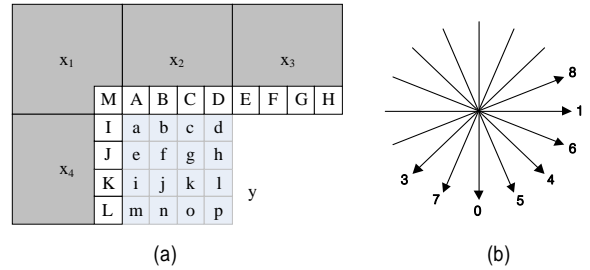


그림 1. 4x4 인트라 예측 모드, (a) 예측 블록과 참조 픽셀 (b) 8가지 예측 방향  
 Fig. 1. 4x4 intra prediction modes, (a) Prediction block and its reference pixels, (b) 8 direction of prediction

원된 주위 블록이다. M은 x<sub>1</sub> 블록내의 참조 화소를 나타내며, A, B, C, D는 x<sub>2</sub> 블록 내의 참조 화소, E, F, G, H는 x<sub>3</sub> 블록 내의 참조 화소, I, J, K, L은 x<sub>4</sub> 블록 내의 참조 화소를 나타낸다.

Chen은 인트라 예측을 소인수 분해를 통해서 행렬의 곱

$$\frac{1}{4} \begin{bmatrix} A+2B+C & B+2C+D & C+2D+E & D+2E+F \\ B+2C+D & C+2D+E & D+2E+F & E+2F+G \\ C+2D+E & D+2E+F & E+2F+G & F+2G+H \\ D+2E+F & E+2F+G & F+2G+H & G+3H \end{bmatrix} = \frac{1}{4} \left( \begin{bmatrix} A+2B+C & B+2C+D & C+2D & D \\ B+2C+D & C+2D & D & 0 \\ C+2D & D & 0 & 0 \\ D & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & E & 2E+F \\ 0 & E & 2E+F & E+2F+G \\ E & 2E+F & E+2F+G & F+2G+H \\ 2E+F & E+2F+G & F+2G+H & G+3H \end{bmatrix} \right) \quad (1)$$

$$\begin{bmatrix} A+2B+C & B+2C+D & C+2D & D \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ A & B & C & D \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ B+2C+D & C+2D & D & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ A & B & C & D \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ C+2D & D & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ A & B & C & D \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ D & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ A & B & C & D \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad (5)$$

으로 나타 낼 수 있음을 보여주었다<sup>[20]</sup>. Chen의 방법을 사용하여 모드 3을 소인수 분해하면 수식 (1)~(5)와 같으며, 수식 (1)의 첫 번째 부분도 각각의 행에 대하여 분해가 가능하다.

여기서 “x”는 “don't care”화소로 실제 계산에 영향을 미치지 않으며, 수식 (1)의 두 번째 부분도 위와 유사한 형태로 전개가 가능하다. 이와 같은 방법으로 각 모드에 대해 소인수 분해로 표현이 가능하며, 수식 (6)과 같이 일반화된 형태로 정리 할 수 있다.

$$\mathbf{y}_{pred}^m = \left( \sum_{n=1}^3 \sum_{i=1}^4 \mathbf{s}_i \mathbf{x}_n \mathbf{c}_{n,i}^m \right) + \left( \sum_{i=1}^4 \mathbf{c}_{4,i}^m \mathbf{x}_4 \mathbf{s}_i^T \right) \quad (6)$$

수식 (6)은 인트라 예측의 일반화된 수식을 나타낸다. 여기에서 m은 인트라 예측 모드를 나타내며,  $\mathbf{x}_n$ 은 그림 1(a)에서의 n번째 이웃하는 블록을 나타낸다.  $\mathbf{c}_{n,i}^m$ 는 참조 블록 오른쪽에 곱해지는 상수 행렬을 나타내며 각 모드와 열에 따라 다른 값을 가진다.  $\mathbf{s}_i$ 는 이동 행렬(Shift matrix)을 나타낸다. 인트라 예측에서 예측 블록은 이웃하는 주변 블록과 경계를 이루는 화소만을 참조한다. 따라서  $\mathbf{x}_n$ 의 하단 또는 우측의 값만을 추출하기 위한 여과기와 일반화 과정에 필요한 연산자를 다음과 같이 정의 한다.

a) 특수 행렬  $\tilde{\mathbf{v}}, \tilde{\mathbf{h}}, \mathbf{u}$ 를 다음과 같이 정의 한다.

$$\tilde{\mathbf{v}} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tilde{\mathbf{h}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \mathbf{u} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

b) 수직복사 연산자  $V(\cdot)$ : 수직 복사 행렬  $\tilde{\mathbf{v}}$ 와 임의의 행렬  $\mathbf{a}$ 의 행렬 곱으로 정의 한다.

$$V(\mathbf{a}) = \tilde{\mathbf{v}}\mathbf{a} = \mathbf{a}_v \quad (7)$$

c) 수평복사 연산자  $H(\cdot)$ : 임의의 행렬  $\mathbf{a}$ 와 수평 복사 행렬  $\tilde{\mathbf{h}}$ 의 행렬 곱으로 정의 한다.

$$H(\mathbf{a}) = \mathbf{a}\tilde{\mathbf{h}} = \mathbf{a}_h \quad (8)$$

d) UNI 연산자  $U(\cdot)$ : 임의의 행렬  $\mathbf{a}$ 의 평균값으로 모든 원소가 구성된 행렬로 정의 한다.

$$U(\mathbf{a}) = \frac{1}{N \times N} \mathbf{u}\mathbf{a}\mathbf{u} \quad (9)$$

이동 행렬과 수직/수평 복사 행렬의 특성을 이용하여, 수식 (6)을 정리 하면 다음과 같다.

$$\mathbf{y}_{pred}^m = \left( \sum_{n=1}^3 \sum_{i=1}^4 \mathbf{s}_i \mathbf{x}_n \mathbf{c}_{n,i}^m \right) + \left( \sum_{i=1}^4 \mathbf{c}_{4,i}^m \mathbf{x}_4 \mathbf{s}_i^T \right) \quad (10)$$

수식 (10)에 의하여 인트라 예측 블록  $\mathbf{y}_{pred}^m$ 을 구하기 위해서는 이웃하는 블록의 모든 값을 필요로 하지 않고,  $\mathbf{x}_n, \mathbf{v}, \mathbf{x}_n, \mathbf{h}$  값만이 필요함을 알 수 있다. 또한 현재 블록은 다음 예측 블록의 참조 블록으로 사용됨으로 현재 블록의 복원된 블록의 수직/수평 여과된 값을 저장하여 다음 블록의 연산에 사용될 수 있다.

수식 (11)은 현재 블록  $\mathbf{y}$ 의 예측 블록  $\mathbf{y}_{pred}^m$ 이 다음 블록  $\mathbf{y}'$ 의 인트라 예측을 위해 수직/수평 여과되는 과정을 나타낸다.

$$\mathbf{y}'_{pred}^m = \left( \sum_{n=1}^3 \sum_{i=1}^4 \mathbf{s}_i V(\mathbf{y}_{pred}^m) \mathbf{c}_{n,i}^m \right) + \left( \sum_{i=1}^4 \mathbf{c}_{4,i}^m H(\mathbf{y}_{pred}^m) \mathbf{s}_i^T \right) \quad (11)$$

여기서  $\mathbf{y}'_{pred}^m$ 는  $\mathbf{y}'$ 블록의 인트라 예측 블록이다.  $V(\mathbf{y}_{pred}^m), H(\mathbf{y}_{pred}^m)$ 는 각각 다음과 같다.

$$\mathbf{y}_{pred,v}^m = V(\mathbf{y}_{pred}^m) = \left( \sum_{n=1}^3 \sum_{i=1}^4 V(\mathbf{s}_i \mathbf{x}_n \mathbf{c}_{n,i}^m) \right) + \left( \sum_{i=1}^4 V(\mathbf{c}_{4,i}^m \mathbf{x}_4 \mathbf{s}_i^T) \right) \quad (12)$$

$$\mathbf{y}_{pred,h}^m = H(\mathbf{y}_{pred}^m) = \left( \sum_{n=1}^3 \sum_{i=1}^4 H(\mathbf{s}_i \mathbf{x}_n \mathbf{c}_{n,i}^m) \right) + \left( \sum_{i=1}^4 H(\mathbf{c}_{4,i}^m \mathbf{x}_4 \mathbf{s}_i^T) \right) \quad (13)$$

수식 (12), (13)는 수식 이동 행렬과 수직/수평 복사 행렬의 성질을 이용하여 다음과 같이 간략화 시킬 수 있다.

$$\mathbf{y}_{pred,v}^m = \mathbf{x}_{1,v} \mathbf{c}_{1,4}^m + \mathbf{x}_{1,v} \mathbf{c}_{2,4}^m + \mathbf{x}_{1,v} \mathbf{c}_{3,4}^m + (\mathbf{p}_{4,v}^m \mathbf{x}_4 \mathbf{h})^T, \quad (14)$$

여기에서  $\mathbf{p}_{4,v}^m = \sum_{i=1}^4 (\mathbf{s}_i V(\mathbf{c}_{4,i}^m))$

$$\mathbf{y}_{\text{pred},h}^m = \mathbf{x}_{1,v} \mathbf{q}_{1,h}^m + \mathbf{x}_{2,v} \mathbf{q}_{2,h}^m + \mathbf{x}_{3,v} \mathbf{q}_{3,h}^m + \mathbf{c}_{4,4}^m \mathbf{x}_{4,h} \quad (15)$$

여기에서  $\mathbf{q}_{4,h}^m = \sum_{i=1}^4 (\mathbf{H}(\mathbf{c}_{n,i}^m) \mathbf{s}_i^T)$

본 논문에서는 16x16 인트라 예측의 일반화에 대해서는 생략한다.

## 2. 주파수 공간에서 썸네일 추출

앞서서 인트라 예측에 대한 일반화 과정과 수직/수평 여과기를 통해 수식을 정리 하였다.

또한 변형된 인트라 예측 식을 통하여 인트라 예측을 수행할 때 주변 블록의 전체가 필요하지 않고 이웃하는 블록의  $\mathbf{x}_{n,v}$ ,  $\mathbf{x}_{n,h}$  의 값만을 이용하여 다음 블록을 쉽게 예측 할 수 있음을 증명하였다. 이번 장에서는 변형된 수식을 이용하여 주파수 공간에서 썸네일을 생성하는 방법을 설명한다.

현재 블록  $\mathbf{y}$ 는 인트라 예측 블록  $\mathbf{y}_{\text{pred}}^m$ 과 예측오차  $\mathbf{y}_e$ 의 합으로 나타낼 수 있다.

$$\mathbf{y} = \mathbf{y}_{\text{pred}}^m + \mathbf{y}_e = \left( \sum_{n=1}^3 \sum_{i=1}^4 \mathbf{s}_i \mathbf{x}_{n,v} \mathbf{c}_{n,i}^m \right) + \left( \sum_{i=1}^4 \mathbf{c}_{4,i}^m \mathbf{x}_{4,h} \mathbf{s}_i^T \right) + \mathbf{y}_e \quad (16)$$

화소 공간에서의 가로, 세로 각각 1/4의 축소된 영상  $\mathbf{y}_{\text{dc}}$ 는 현재 블록  $\mathbf{y}$ 의 화소 값의 평균으로 구할 수 있다. 축소 영상은 인트라 예측 블록의 DC 값과 예측 오차에 대한 주파수 공간에서의 DC 값의 합을 이용하여 계산이 가능하다. 화소 공간에서의 블록의 평균은 주파수 공간에서 DC 값과 같기 때문에 수식 (16)에  $U(\cdot)$  연산을 적용하여 다음과 같이 나타낼 수 있다.

$$\begin{aligned} \mathbf{y}_{\text{dc}} &= (U(\mathbf{y}))_{(0,0)} \\ &= (U(\mathbf{y}_{\text{pred}}^m) + U(\mathbf{y}_e))_{(0,0)} \\ &= \frac{1}{4} (\text{HT}(\mathbf{y}))_{(0,0)} \\ &= \frac{1}{4} (\text{HT}(\mathbf{y}_{\text{pred}}^m))_{(0,0)} + \frac{1}{4} (\text{HT}(\mathbf{y}_e))_{(0,0)} \end{aligned} \quad (17)$$

썸네일 영상  $\mathbf{y}_{\text{dc}}$ 은 HT(H.264/AVC modified DCT) 수식에 의하여 변환된 블록의 DC 값은 화소 공간상의 블록의 평균보다 4배가 크기 때문에 1/4를 곱해서 계산된다.

수식 (17)에서  $U(\mathbf{y}_{\text{pred}}^m)$ 을  $U(\cdot)$  연산자와  $\mathbf{x}_{n,v}$ ,  $\mathbf{x}_{n,h}$  값을 이용하여 정리 하면 다음과 같다.

$$\begin{aligned} \mathbf{y}_{\text{pred},uni}^m &= U(\mathbf{y}_{\text{pred}}^m) \\ &= \frac{1}{16} \sum_{n=1}^3 \mathbf{x}_{n,v} \sum_{i=1}^4 \mathbf{c}_{n,i}^m \mathbf{u} + \frac{1}{16} \sum_{i=1}^4 \mathbf{u} \mathbf{c}_{4,i}^m \mathbf{x}_{4,h} \\ &= \sum_{n=1}^3 \mathbf{x}_{n,v} \mathbf{c}'_n + (\mathbf{c}'_n)^T \mathbf{x}_{4,h} \end{aligned} \quad (18)$$

$$\text{여기에서 } \mathbf{c}'_n = \frac{1}{16} \sum_{i=1}^4 \mathbf{c}_{n,i}^m \mathbf{u}$$

수식 (18)에서  $\mathbf{y}_{\text{pred},uni}^m$ 는 현재 블록에 대한 예측값을 나타내며 예측 오차와 함께 썸네일 데이터로 사용된다. 16x16 인트라 예측 블록에서도 4x4 경우와 유사한 방법으로 썸네일을 구할 수 있다.

지금까지 화소 공간에서 인트라 예측의 일반화식을 변형하여 썸네일을 추출하는 방법에 대해서 설명 하였다. 위에서 전개한 썸네일 추출 방법에서 HT를 이용하면 부호화 스트림으로부터 완전 복호화 하지 않고 주파수 공간에서 직접 썸네일을 생성할 수 있다. 화소 공간에서 유도된 수식들을 주파수 공간으로 수식을 바꾸어 정리하면 다음과 같다.

$$\begin{aligned} \mathbf{Y}_{\text{pred},v}^m &= \text{HT}(\mathbf{x}_{1,v} \mathbf{c}_{1,4}^m + \mathbf{x}_{2,v} \mathbf{c}_{2,4}^m + \mathbf{x}_{3,v} \mathbf{c}_{3,4}^m + (\mathbf{P}_{1,4}^m \mathbf{x}_{4,h})^T) \\ &= \mathbf{X}_{1,v} \mathbf{C}_{1,4}^m + \mathbf{X}_{2,v} \mathbf{C}_{2,4}^m + \mathbf{X}_{3,v} \mathbf{C}_{3,4}^m + (\mathbf{P}_{1,4}^m \mathbf{X}_{4,h})^T \end{aligned} \quad (19)$$

$$\mathbf{Y}_{\text{pred},h}^m = \left( \sum_{n=1}^3 \mathbf{X}_{n,v} \mathbf{Q}_{n,h}^m \right)^T + \mathbf{C}_{4,4}^m \mathbf{X}_{4,h} \quad (20)$$

$$\mathbf{Y}_{\text{pred},uni}^m = \sum_{n=1}^3 \mathbf{X}_{n,v} \mathbf{C}'_n + (\mathbf{C}'_n)^T \mathbf{X}_{4,h} \quad (21)$$

여기에서  $\mathbf{C}_{n,i}^m$ ,  $\mathbf{X}_{n,v}$ ,  $\mathbf{X}_{n,h}$ 는  $\mathbf{c}_{n,i}^m$ ,  $\mathbf{x}_{n,v}$ ,  $\mathbf{x}_{n,h}$ 의 HT 변환된 행렬이다.  $\mathbf{P}_{1,4}^m$ ,  $\mathbf{Q}_{n,h}^m$ 는 각각 수식 (14), (15)의  $\mathbf{p}_{1,4}^m$ ,  $\mathbf{q}_{n,h}^m$ 에 대한 HT변환된 형태이다. 주파수 공간에서의 썸네일은 수

식 (21)에 1/4를 곱하여 계산 된다.

그림 2는 썸네일 추출기 구성도를 보여준다.

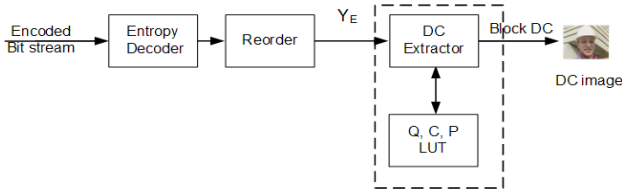


그림 2. H.264/AVC부터 썸네일을 추출하는 방법  
Fig. 2. Thumbnail extraction method in H.264/AVC bitstreams

DC 추출기(DC Extractor)는 수식 (19) ~ (21)를 수행하여 주파수 공간상에서 빠르게 DC값을 뽑아낸다. 이때 사용되는 Q, C, P 테이블은 모드에 따른 상수 테이블 이므로 미리 계산하여 추가적인 연산을 피한다. DC 추출기에는 다음 블록의 인트라 예측 블록을 복원하기 위해서  $y_{pred,v}^m$ 와  $y_{pred,h}^m$ 를 계산하는 블록이 포함되어 있다. 그림 3은 DC 추출기의 내부 구조를 나타낸다.

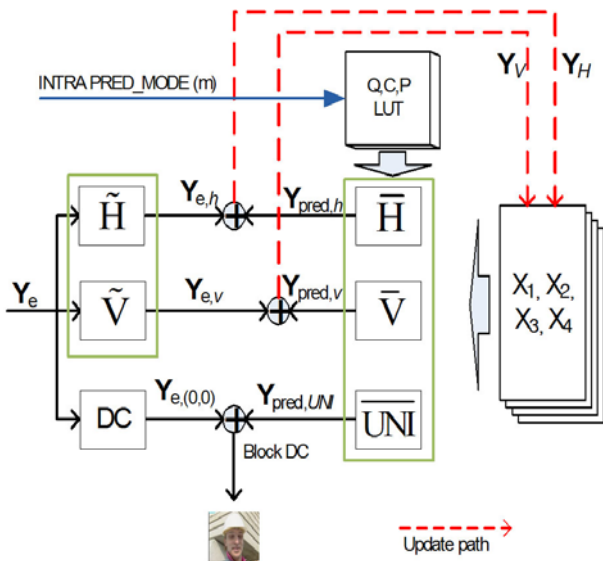


그림 3. DC 추출기의 구조  
Fig. 3. The structure of DC extractor

$Y_e$ 는 주파수 상의 예측 오차를 나타낸다.  $\tilde{H}, \tilde{V}$ 는 예측 오차의 주파수 값에 대한 여과 과정을 수행하며,  $\bar{H}, \bar{V}$ 는

이웃하는 블록  $X_1, X_2, X_3, X_4$ 을 이용하여  $Y_{pred}$  값을 구하는 연산을 수행한다. 마찬가지로  $\bar{UNI}$ 는  $X_1, X_2, X_3, X_4$ 을 이용하여 썸네일을 구하기 위해  $U(\cdot)$  연산과정을 수행한다. 최종적으로 주파수 공간에서 구하고자 하는 썸네일은  $U(\cdot)$  연산을 수행한 후 얻은  $Y_{pred,uni}$ 의 (0,0)값에  $Y_e$ 의 (0,0) 값을 더하여 얻는다.

DC 추출기의 동작 순서는 다음과 같다.

- Step 1) 입력된 비트스트림으로부터 역 양자화 과정까지 거쳐 생성된  $Y_e$ 에 대하여 각각 수직과 수평 여과 과정을 수행하여  $Y_{e,h}$ 와  $Y_{e,v}$ 를 생성한다.
- Step 2) LUT와 이웃한 블록인  $X_{n,v}, X_{n,h}$ 와의 연산을 통해  $Y_{pred,uni}, Y_{pred,v}, Y_{pred,h}$ 을 생성한다.
- Step 3)  $(Y_{pred,uni})_{(0,0)}$ 과  $(Y_e)_{(0,0)}$ 을 이용하여 썸네일 값을 생성한다.
- Step 4)  $Y_{pred,v}$ 와  $Y_{e,v}$  그리고  $Y_{pred,h}$ 와  $Y_{e,h}$ 의 합을 저장하고 다음 블록의 인트라 예측 블록 생성을 위한  $X_{n,v}, X_{n,h}$ 을 생성하여 저장한다.

### III. 제안하는 오차 보상 방법

H.264/AVC는 인트라 예측과 정수 변환, 그리고 양자화 전 과정에서 정수 연산을 사용하도록 되어있다. 그리고 각 과정마다 부호화기와 복호화기 간의 오차를 제거하기 위해 반올림 연산을 사용한다. 반올림 연산은 멀티미디어 시스템에서 흔하게 사용되는 연산이다. 하지만 화소 공간에서 수행되는 반올림 연산은 주파수 공간상에서 정의되지 않는다. 따라서 H.264/AVC의 공간 주파수 기반의 썸네일 추출 방법에서는 부동소수점 연산을 사용하기 때문에 정수 연산과 부동소수점 사이의 불일치 문제와 반올림 오차 문제를 포함하게 된다.

이 장에서는 썸네일 추출 시 발생하는 오차에 대해 기술하고 인트라 예측에서 수행되는 반올림 오차 문제를 보상하기 위한 방법을 설명한다.

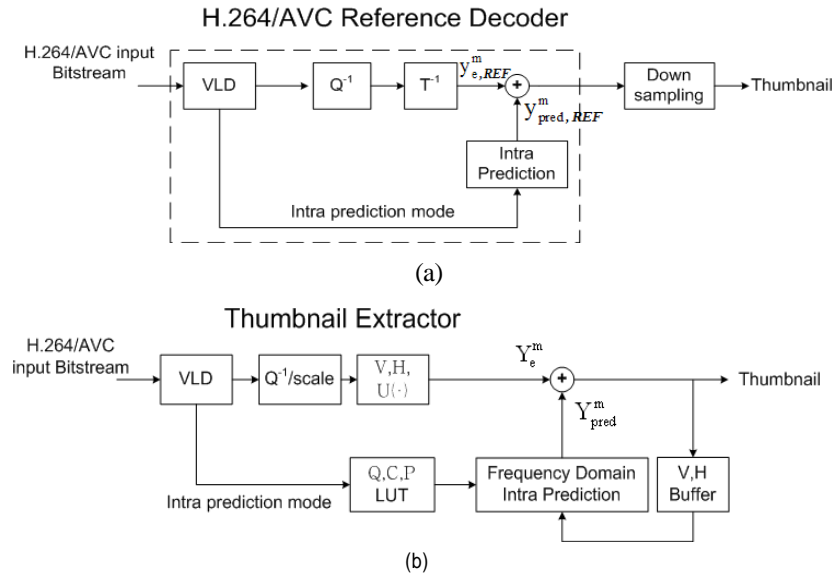


그림 4. 썸네일 추출기의 블록도 (a) H.264/AVC의 참조 복호화기에서 썸네일을 추출하는 방법 (b) 주파수 공간에서 썸네일을 추출하는 방법  
 Fig. 4. Block diagram of thumbnail extractor (a) Thumbnail extraction method from H.264/AVC reference decoder (b) Thumbnail extraction method in frequency domain

1. 썸네일 추출 시 발생하는 오차

$$D^m = D_{pred}^m + D_e^m \tag{24}$$

그림 4(a)는 H.264/AVC의 참조 복호화기에서 썸네일을 추출하는 블록도 이고, (b)는 주파수 공간에서 썸네일을 추출하는 블록도 이다. (b)의 scale은 역 양자화된 예측오차 값을 64와 PF로 나눈 값이다. 역 양자화된 예측오차 값을 64와 PF로 나눠줌으로써 HT공간으로 옮길수 있다. PF는 H.264/AVC의 Post scaling Factor이다.

그림 4의 (a)와 (b)의 블록도에서 예측 오차 블록과 예측 블록 그리고 현재 블록에서 발생하는 오차를 HT 공간에서 수식으로 정리하면 다음과 같다.

$$D_{pred}^m = HT(y_{pred,REF}^m) - Y_{pred}^m \tag{22}$$

$$D_e^m = HT(y_{e,REF}^m) - Y_e^m \tag{23}$$

수식 (22)에서  $D_{pred}^m$ 은 인트라 예측의 반올림 오차를 나타내며, 수식 (23)에서  $D_e^m$ 은 예측오차 성분 간의 오차 블록을 나타낸다. 썸네일 추출 시 발생하는 총 오차는 수식 (24)같이 정리할 수 있다.

그림 5는 인트라 예측에서 수행되는 반올림 오차의 영향을 그래프로 나타낸 것이다. 세로축은 PSNR을 나타낸다.

QP33 까지는 반올림 오차의 영향이 크며, QP45에서는 반대로 양자화 오차가 반올림 오차보다 월등히 많음을 확인할 수 있다. 그림 5를 통하여 멀티미디어 방송에서 주

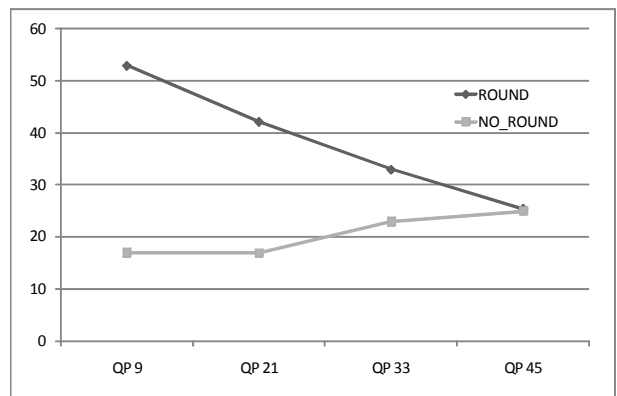


그림 5. 인트라 예측의 반올림 오차의 영향  
 Fig. 5. Influence of round-off effect in Intra prediction

로 사용하는 QP대역에서는  $D_{pred}^m \gg D_e^m$ 로 가정할 수 있다. 따라서 수식(24)은 (25)와 같이 근사화 할 수 있으며, 현재 블록에 존재하는 오차는 인트라 예측에서 발생하는 반올림 오차로 가정 할 수 있다.

$$D^m \cong D_{pred}^m \tag{25}$$

화소 공간에서 발생하는 인트라 예측의 반올림 오차는 주파수 공간에서 정의 되지 않는다. 따라서  $D^m$ 을 주파수 공간에서 값을 구하기 어려우므로,  $D^m$ 의 예측값인  $D'^m$ 을 구해야 한다.  $D'^m$ 은 각 블록내의 계수들의 위치에서 발생하는 반올림 오차들의 통계적 모델을 통해 근사화 할 수 있다.

## 2. 통계적 모델을 이용한 오차 보상 방법

일반적으로 반올림 연산의 마지막 과정에서는 정수자리 이하의 소수점 자리를 버리게 된다. 이때 버려지는 값들 즉, 절단오차들은 나누는 수에 따라 일정한 집합을 갖게 되며, 이 집합의 원소들은 확률적으로 균일분포를 가진다. 이 수들의 확률분포를 가지고 절단 오차에 대한 랜덤변수  $r$ 을 생성할 수 있다. 따라서  $r$ 에 대한 분산인  $E[(r-s)^2]$ 을 최소화 하는  $s$ 를 찾으면 반올림 오차를 보상할 수 있다. 여기서  $s$ 는 간단하게  $E[r]$ 로 계산될 수 있다<sup>[19]</sup>.  $s$ 는 인트라 예측의 반올림 오차에 대하여 수식 (26)와 같이 표현될 수 있다.

$$s^m = \begin{bmatrix} E[r_{0,0}] & \cdots & E[r_{0,3}] \\ \vdots & \ddots & \vdots \\ E[r_{3,0}] & \cdots & E[r_{3,3}] \end{bmatrix} \tag{26}$$

그림 6은 한 프레임에서 DC모드로 선택된 블록들의 (0,0)위치에서의 오차 분포를 나타낸다. 이 분포를 이루고 있는 오차들의 평균값은 주파수 공간의 블록내의 계수 위치에서 보상해야 할 대푯값으로 볼 수 있다.

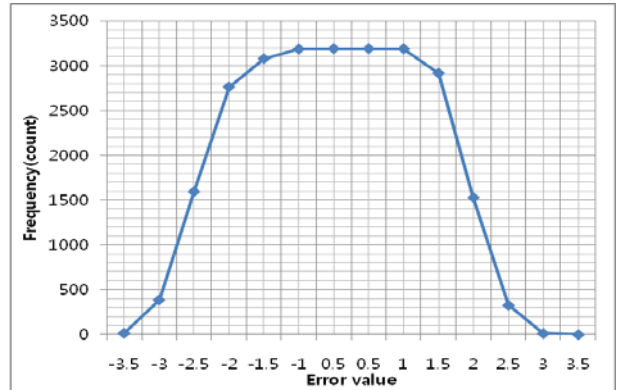


그림 6. DC모드로 선택된 블록의 (0,0)위치에서 오차값들의 분포  
Fig. 6. Error distribution of (0,0) coded with DC mode block

이러한 오차 분포를 통해 구한 대푯값들을 주파수 공간의 블록으로 각 모드와 계수 위치로 표현하면 수식 (27)과 같이 표현할 수 있다.  $D_{(0,0)}$ 은 (0,0)위치에서의 오차값을 나타낸다.

$$D'^m = \begin{bmatrix} E[D_{0,0}] & \cdots & E[D_{0,3}] \\ \vdots & \ddots & \vdots \\ E[D_{3,0}] & \cdots & E[D_{3,3}] \end{bmatrix} \tag{27}$$

수식 (28)은 실험을 통해 실험 영상들의 인트라 프레임에서 모드2의 오차값의 대푯값들을 구한 것이다. 주파수 공간에서의 썸네일 추출 방법에서는  $V(\cdot), H(\cdot)$  형태의 블록을 사용하므로 보상값 역시  $V(D'^m), H(D'^m)$ 의 형태가 되어야 한다.

$$V(D'^m) = \begin{bmatrix} 0.2585 & 0.0001 & -0.0015 & -0.0083 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{28}$$

$$V(HT(s^m)) = \begin{bmatrix} 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

수식 (28)을 통해서 통계적 보상 모델인  $V(HT(s^m))$ 와 실험으로 구한  $V(D'^m)$ 이 상당히 유사함을 확인할 수 있



다. 따라서 블록 내 화소 공간에서 절단오차에 대한 기대값은 주파수 공간에서의 오차값들에 대한 평균과 같다고 가정할 수 있다. 이러한 가정을 통하여 수식 (29)에서 인트라 예측의 반올림 오차들에 대한 보상값을 단순히 주파수 공간내의 오차들의 평균값으로 구할 수 있음을 보여준다. 보상값에 대한  $H(\cdot)$  형태도  $V(\cdot)$ 와 유사하게 구할 수 있다.

$$V(HT(s^m)) \cong V(D_{pred}^m) = \begin{bmatrix} E[D_{0,0}] & E[D_{0,1}] & E[D_{0,2}] & E[D_{0,3}] \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (29)$$

인트라 예측에서 수행되는 반올림 연산은 블록 내 각 위치마다 동일한 반올림 연산을 수행한다. 따라서  $D'^m$ 은 영상의 특성이 아닌 예측 모드의 특성에 기반 한다. 이러한 특성에 의해서  $D'^m$ 을 서로 다른 영상의 동일한 모드에 적용할 수 있는 장점이 있다. 만들어진 보상값 들은 LUT형태로 저장되어 각 모드에 따라 현재 블록에 더하는 방식으로 간단하게 보상할 수 있다. 휘도 성분 뿐만 아니라 색차 성분도 동일한 방법으로 실험을 통해 보상값을 결정하고 LUT에 저장하여 적용이 가능하다.

#### IV. 실험결과

본 논문에서는 주파수 공간상에서의 썸네일 추출법에 대한 보상 방법을 제안하였다. 썸네일은 원본영상에 대한 대략적인 정보를 제공하며 빠른 검색과 색인에 사용된다. 따

라서 썸네일에서는 추출 수행시간이 중요하다. 이번 장에서는 제안하는 방법의 성능을 주관적 화질 비교와 수행 시간을 비교하여 보여 준다.

그림 7은 제안하는 방법의 성능을 평가하기 위한 블록도로, H.264/AVC 참조 복호화에 의해 얻어진 영상을 1/16 크기로 다운샘플링을 수행한 영상군을 A라 한다. 다운샘플링은 평균 연산자를 이용 하였다. 보상 방법을 적용하지 않고 H.264/AVC의 주파수 공간에서 썸네일을 추출하는 방법을 B라 하며, B의 추출 방법에 제안하는 보상 방법을 적용하여 얻은 영상군을 C라고 한다. 각각의 영상군을 추출하는 수행시간과 각 영상군의 주관적 화질 비교를 통해 제안하는 방법의 성능을 검증 하고자 한다. 표 1은 실험환경을 나타낸다.

표 1. 실험환경  
Table 1. Experimental environment

구분	실험 환경	
참조 소프트웨어	JM 12.2 BP, Level 4.0	
프레임 부호화 모드	인트라	
적응형 프레임/필드 모드	프레임 모드	
양자화 계수	9, 21, 33, 45	
실험 영상	704x576	City, Crew, Harbour, Soccer
	1920x1080	Clip, Table
썸네일 추출 크기	176x144 / 480x270	

그림 8은 세 가지 썸네일 추출 방법에 대한 속도를 비교한 것이다. SD(Spatial Domain)는 H.264/AVC 참조 복호화기에서 얻어진 영상을 화소 공간에서 다운샘플링을 하여 썸네일을 얻는 방법이다. TD(Transform Domain)는 주파수

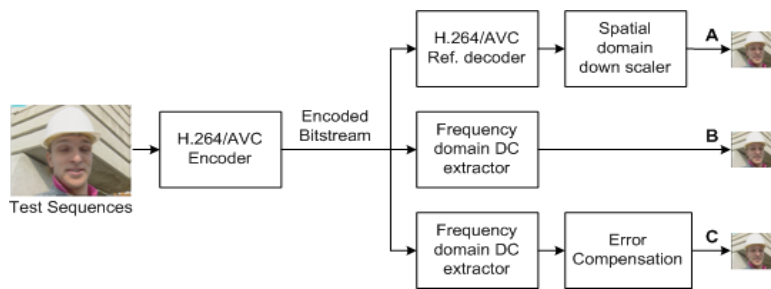


그림 7. 성능평가를 위한 블록도  
Fig. 7. A block diagram for performance evaluation

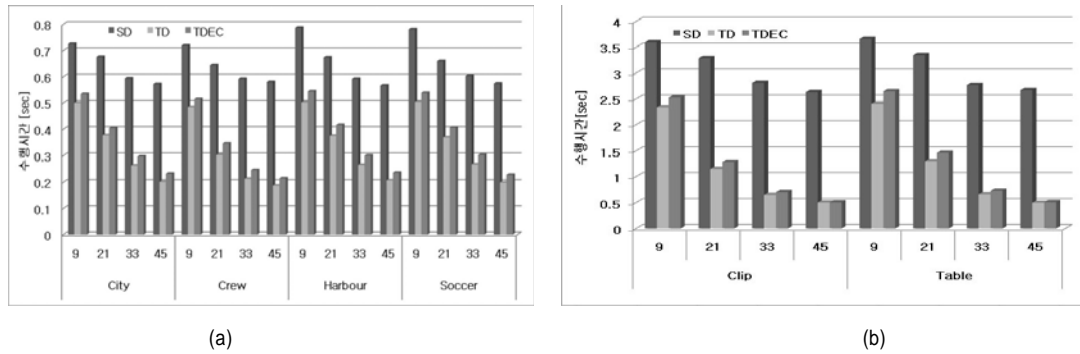


그림 8. QP에 따른 축소 영상 추출 수행 속도 비교 (a)176x144 축소 영상 (b)480x270 축소 영상  
 Fig. 8. Extracting time of SD, TD and TDEC corresponding each QP (a)176x144 thumbnail (b)480x270 thumbnail

공간에서 썸네일을 추출하는 방법이며, TDEC(Transform Domain with Error Compensation)은 TD방법에 제안하는 보상 방법을 적용한 것이다.  
 그림 8에서는 화소 공간에서 썸네일을 추출하는 방법에 비해서 주파수 공간에서 추출하는 방법은 모든 영상에서

속도 향상을 가져온다. TDEC방법은 TD방법에 비해 수행 시간이 다소 높아지는 것은 오차를 보상하는 과정에서 추가적인 연산이 들어가기 때문이다. 그러나 기존의 TD방법과 속도면에서는 큰 차이를 보이지 않는다.  
 그림 9 - 11은 주관적인 화질 비교를 보여 준다. 그림 9은

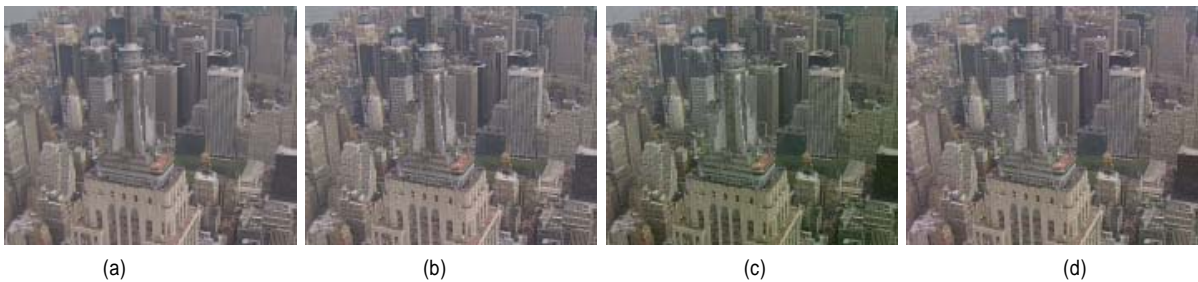


그림 9. City영상에서의 화질 비교, (a) 원 영상을 1/16크기로 축소한 영상, (b) QP 21에서 SD방법으로 추출한 영상 (c) QP 21에서의 TD방법으로 추출한 영상 (d) QP 21에서 TDEC방법으로 추출한 영상  
 Figure 9. Subjective quality performance of City sequence. (a) 1/16 size image (176x144) (b) Thumbnail extracted from SD method in QP 21 (c) Thumbnail extracted from TD method in QP 21 (d) Thumbnail extracted from TDEC method in QP 21



그림 10. Crew영상에서의 화질 비교, (a) 원 영상을 1/16크기로 축소한 영상, (b) QP 21에서 SD방법으로 추출한 영상 (c) QP 21에서의 TD방법으로 추출한 영상 (d) QP 21에서 TDEC방법으로 추출한 영상  
 Figure 10. Subjective quality performance of Crew sequence. (a) 1/16 size image (176x144) (b) Thumbnail extracted from SD method in QP 21 (c) Thumbnail extracted from TD method in QP 21 (d) Thumbnail extracted from TDEC method in QP 21



(a)



(b)



(c)

그림 11. Clip영상에서의 화질 비교, (a) QP 21에서 SD방법으로 추출한 영상 (b) QP 21에서의 TD방법으로 추출한 영상 (c) QP 21에서 TDEC방법으로 추출한 영상

Figure 11. Subjective quality performance of Clip sequence. (a) Thumbnail extracted from SD method in QP 21 (b) Thumbnail extracted from TD method in QP 21 (c) Thumbnail extracted from TDEC method in QP 21

City영상에 대한 결과이며, 704×576의 크기의 영상에 QP 21을 적용한 결과이다. 여기서 SD, TD, TDEC방법을 통해 얻은 영상들은 모두 가로 세로 각각 1/4 크기로 감소한 크기에서 원 영상의 특징을 가지고 있음을 알 수 있다. TD방법을 통해 얻은 썸네일은 영상의 우측 하단으로 갈수록 영상의 밝기가 어두워지는 현상이 발생한다. 이것은 TD방법에서 인트라 예측의 반올림 오차를 보상하지 못해 발생하는 현상이다. TDEC방법을 통해 얻은 썸네일은 TD방법보다 향상된 화질을 보여주며, SD방법을 통해 얻은 썸네일과 유사한 화질을 보여준다. 결과 영상에서, 썸네일 추출 시 발생하는 오차가 제안하는 방법을 통해 보상이 되는 것을 확인할 수 있다. 하지만 1920×1080 크기의 HD영상에서는 보상 후에도 여전히 오차가 남아 있음을 확인할 수 있다. 실제로 구한 보상값은 반올림 오차에 대한 예측치이기 때문에 완벽하게 보상을 할 수는 없는 단점이 존재한다.

그림 10-11은 각각 Crew, Clip영상에 대해 QP 21을 적용하였을 때 결과이며, City영상 실험에서와 유사한 결과를 보여준다.

표 2. 각 영상의 첫 번째 인트라 프레임에서 QP에 따른 PSNR 비교

Table 2. PSNR for the first intra frame in different sequences at different QPs

Sequence	Method	PSNR(dB)			
		QP 9	QP 21	QP 33	QP 45
City	TD	27.94	22.36	24.85	29.79
	TDEC	39.75	34.14	33.1	30.15
Crew	TD	27.26	24.49	31.61	33.74
	TDEC	37.41	37.37	38.54	32.99
Harbour	TD	30.33	23.83	27.17	27.88
	TDEC	41.27	36.99	31.83	28.92
Soccer	TD	26.20	24.45	24.06	30.18
	TDEC	38.65	40.66	31.46	30.17
Clip	TD	18.76	19.33	24.04	31.22
	TDEC	32.05	33.88	27.65	28.69
Table	TD	17.04	15.58	21.09	30.07
	TDEC	28.64	36.47	27.84	31.17

표 2는 결과영상의 PSNR을 나타낸다. 원본 영상에 대한 1/16 크기의 영상이 존재하지 않기 때문에, 원본영상을 화소 공간에서 1/16 크기로 다운샘플링을 하여 TD와 TDEC에 대한 PSNR을 측정하였다. QP45를 제외한 전 QP대역에서 객관적으로 화질이 개선됨을 확인할 수 있으며, QP45에서는 양자화 오차가 커짐으로 인해 TD와 TDEC의 PSNR

이 유사 하다. 모든 실험 영상에 대하여 TDEC을 적용함으로써 오차 보상이 없는 TD 보다 평균적으로 약 8.1dB의 화질개선을 얻을 수 있다.

실험 결과를 통하여 제안하는 보상 방법을 적용하여 썸네일을 추출할 경우 적용하지 않았을 때의 방법과 비교하여 유사한 추출속도를 보이며, 보다 향상된 화질의 썸네일을 얻을 수 있다. 보상 방법으로 인한 속도저하는 최소 4%, 최대 6%정도이며 비트스트림을 완전히 복호화 하여 썸네일을 추출하는 방법보다 D1급 영상에서는 평균 50%, HD급 영상에서는 평균 64% 빠르게 썸네일을 추출할 수 있다.

### V. 결 론

본 논문에서는 H.264/AVC의 인트라 프레임에서 썸네일 추출 시 발생하는 오차에 대한 보상 방법을 제안하였다. 제안하는 보상 방법에서는 H.264/AVC의 참조 복호화기로부터 썸네일을 추출하는 방법과 H.264/AVC 주파수 공간에서 썸네일을 추출하는 방법을 비교하여 썸네일 추출 시 발생하는 오차를 기술하였다. 또한 인트라 예측에서 발생하는 반올림 오차에 대한 실험적 모델과 실제 썸네일 추출 시 발생하는 오차에 대한 통계적 모델을 통해 보상값을 결정하였다. 결정된 보상값은 LUT에 저장되어 쉽게 보상할 수 있다. 실험결과를 통해 제안하는 방법이 썸네일 추출 시 발생하는 오차를 효과적으로 보상함을 확인할 수 있었으며, 추출속도 면에서 큰 저하 없이 D1급과 HD급 영상에서 TD보다 평균적으로 약 8.1dB의 화질개선을 얻을 수 있었다.

### 참 고 문 헌

[1] B. Yeo and B. Liu, "Rapid scene analysis on compressed video", IEEE Trans. Circuits and Systems for video Technology, vol. 5, no. 6, pp. 533-540, 1995.

[2] Wing-San Chau, Oscar C. Au, Tak-Song Chong, Tai-Wai Chan, Chi-Shun Cheung, "Efficient Scene Change Detection in MPEG Compressed Video Using Composite Block Averaged Luminance Image Sequence," IEEE ICICS, pp.688-691, 2005.

[3] 윤자찬, 설상훈, "압축영역에서 빠른 축소 영상 추출을 위한 다중부호 록업테이블 설계," 한국방송공학회, vol. 10, no.3, pp.413-421, May 2005.

[4] 김명준, 송병철, 장성규, 나종범, "변형된 IDCT 기저함수를 이용한 동영상 축소 전환 기법," 한국통신학회 논문지, 제24권, 제11B호, pp.2147-2153, Nov. 1999.

[5] 김성득, 나종범, "압축된 동영상에서의 AC 예측 기법을 이용한 DC 영상 추출기법," 대한전자공학회 학술대회 논문집, 제21권 2호, pp.867-870, 2005.

[6] N. Merhav and V. Bhaskaran, "Fast algorithms for DCT-domain image down-sampling and for inverse motion compensation," IEEE Trans. Circuits Syst. Video Technol., vol. 7, pp. 468-476, June 1997.

[7] Taskiran, C. Jau-Yuen Chen, Bouman, C.A. Delp, E.J. "A compressed video database structured for active browsing and search," IEEE image Processing, vol.3, pp.133-137, Oct. 1998.

[8] Jung-Rim Kim, Sungjoo Suh, Sanghoon Sull, "Fast scene change detection for personal video recorder," IEEE Trans. Consumer Electronics, vol. 49, no. 3, pp. 683-688, Aug. 2003.

[9] Ja-Cheon Yoon, Hyeokman Kim, Seong Soo Chun, Jung-Rim Kim, Sanghoon Sull, "Real-Time Video Indexing System for Live Digital Broadcast TV Programs," Lecture Notes in Computer Science, CVIR2004, vol. 3115, pp. 261-269, Jul. 2004.

[10] G.C. Feng and J. Jiang, "Image extraction in DCT domain," IEE Proc.-Vis. Image Signal Process., Vol. 150, No. 1, pp.20-27, Feb. 2003

[11] ITU-T Recommendation H.263. Video coding for low bitrate communication, Nov. 1995.

[12] ISO/IEC, JTC1/SC29/WG11, Coding of moving pictures and associated audio for digital storage media at up to about 1.5Mbits/s -- Part 2: Video, FDIS 11172-2, June 1996.

[13] ISO/IEC, JTC1/SC29/WG11, Information Technology-coding of audio-visual objects, Part 2: Video, FDIS 14496-2 Oct. 1998.

[14] ISO/IEC, JTC1/SC29/WG11 (MPEG), Information technology - Generic coding of moving pictures and associated audio information: Video, Inter-national Standard 13818-2, ISO/IEC, 2000.

[15] J. Song and B.L. Yeo, "Fast Extraction of Spatially Reduced Image Sequences from MPEG-2 Compressed Video," IEEE Trans. Circuits and Systems for Video Technology, vol. 9, no.7, pp. 1100-1114. Oct. 1999.

[16] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC), 2003.

[17] 권용광, 원치선 "DMB를 위한 요약영상 추출 알고리즘" 한국방송공학회, 제12권, 제5호, pp.525-534, Sep, 2007

[18] 유상준, 윤명근, 김은석, 손채봉, 심동규, 오승준, "H.264/AVC 비트스트림에서 효율적으로 축소 영상을 추출하는 방법" 한국방송공학회, 제13권, 제2호, pp.222-235, Mar, 2008

[19] Ping-Hao Wu, Chen Chen, and Horner H. Chen "Rounding Mismatch Between Spatial - Domain and Transform-Domain Video Codecs", IEEE Trans. Circuits and Systems for Video Technology, vol. 16, no. 10, pp.1286-1293, Oct2006.

[20] Chen Chen, Ping-Hao Wu and Chen H, "Transform-Domain Intra

Prediction for H.264," IEEE ISCAS, pp.1497-1500, May 2005.  
 [21] C. Chen, P.-H. Wu, and H. Chen, "MPEG-2 to H.264 trans-

coding," Picture Coding Symposium, Dec. 2004

---

## 저 자 소 개

---



### 윤 명 근

- 2007년 2월 : 광운대학교 전자공학과 졸업 공학사
- 2007년 3월 ~ 현재 : 광운대학교 대학원 전자공학과 석사과정
- 주관심분야 : H.264/AVC, Thumbnail, 영상압축



### 이 여 송

- 1996년 2월 : 광운대학교 전자공학과 졸업 공학사
- 1999년 2월 : 광운대학교 전자공학과 공학석사
- 2007년 8월 ~ 현재: 광운대학교 전자공학과 박사 과정
- 주관심분야 : 방송 시스템, 비디오 압축



### 손 채 봉

- 1993년 2월 : 광운대학교 전자공학과 공학사
- 1995년 2월 : 광운대학교 대학원 전자공학과 공학석사
- 2006년 2월 : 광운대학교 대학원 전자공학과 공학박사
- 1991년 8월 ~ 1993년 2월 : 삼성전자 소프트웨어 멤버십 1기
- 2000년 3월 ~ 2006년 2월 : 한양여자대학 인터넷정보과 전임강사
- 2001년 3월 ~ 현재: 주식회사 손랩 대표이사
- 2006년 3월 ~ 현재 : 광운대학교 전자통신공학과 조교수
- 주관심분야 : HDTV, UDTV, IPTV, DMB, DVB-H System



### 박 호 종

- 1986년 2월 : 서울대학교 전자공학과 공학사
- 1987년12월: Univ. Wisconsin-Madison 전기 및 컴퓨터공학과 공학석사
- 1987년 5월 : Univ. Wisconsin-Madison 전기 및 컴퓨터공학과 공학박사
- 1993년 8월 ~ 1997년 8월 : 삼성전자 정보통신본부 선임연구원
- 1997년 9월 ~ 현재 : 광운대학교 전자공학과 교수
- 주관심분야 : 멀티미디어 시스템, 음성/오디오 신호처리



저 자 소 개



안 창 범

- 1981년 : 서울대학교 전기공학과 학사
- 1983년 : 한국과학기술원 전기및전자공학과 석사
- 1986년 : 한국과학기술원 전기및전자공학과 박사
- 1987년 ~ 1991년 : 미국 University of California, Irvine 조교수
- 1992년 ~ 현재 : 광운대학교 전기공학과 교수
- Sylvia Sorkin Greenfield Award, 장영실상, 메디슨 의공학상, 광운대 Best Teacher 상, 게르베 학술상 등 수상.
- 주관심분야 : 영상처리, 영상압축, 의학영상시스템, 다차원 생체신호처리 시스템.



오 승 준

- 1980년 2월 : 서울대학교 전자공학과 졸업(학사)
- 1982년 2월 : 서울대학교 전자공학과 대학원 졸업(석사)
- 1988년 5월 : 미국 Syracuse University 전기/컴퓨터공학과 졸업(박사)
- 1982년 3월~1992년 8월 : 한국전자통신연구원 근무(멀티미디어연구실 실장)
- 1986년 7월~1986년 8월 : NSF Supercomputer Center 초청 학생연구원
- 1987년 5월~1988년 5월 : Northeast Parallel Architecture Center 학생연구원
- 1992년 3월~1992년 8월 : 충남대학교 컴퓨터공학부 겸임교수
- 1992년 9월~현재 : 광운대학교 전자공학부 교수 (VIA-Multimedia Center 멀티미디어연구실)
- 2002년 3월~현재 : SC29-Korea 의장 및 MPEG Forum 부의장
- 주관심분야 : 비디오 데이터 처리, 비디오 코덱 시스템, 비디오 및 영상압축, 멀티미디어시스템