

論文2002-39SP-6-4

MPEG-4 동영상 압축에서 비트율과 오류 내성을 고려한 인트라 업데이트

(Rate-Distortion Optimized Error-Resilient Intra Update in MPEG-4 Video Coding)

金祐湜*, 朴來弘*

(Woo-Shik Kim and Rae-Hong Park)

요 약

동영상 압축에서 움직임 보상은 뛰어난 압축 효율을 보이는 반면에 어느 한 시점에서 오류가 발생할 경우 그 시점 이후로 계속해서 오류가 확산된다는 단점이 있다. 인트라 업데이트 방법은 이러한 오류의 확산을 차단하는 방법이다. 그러나 인트라 업데이트 방법은 압축의 효율을 떨어뜨린다는 단점이 있다. 본 논문에서는 오류의 영향과 비트율 제어를 함께 고려하여 인트라 업데이트를 할 수 있는 방법을 제안한다. 제안된 방법은 비트율과 왜곡의 값을 추정하고 라그랑지안 최적화를 통해 부호화 모드와 양자화 파라미터를 선택하는 방법으로 MPEG-4 코덱에 적용하여 실험하였다. 실험 결과는 제안된 방법이 기존의 방법보다 패킷 손실과 같은 오류 환경에서 더 좋은 화질을 나타내는 것을 보여준다.

Abstract

Motion compensation is a powerful method to compress an image sequence. Its main drawback is that once an error is occurred, the error propagates through the frames. Recently, the intra update method was proposed to stop the error propagation at the expense of reduction in compression efficiency. This paper proposes an intra update method based on a rate-distortion optimization in error prone environments. The rate and the distortion are estimated using the Lagrangian optimization to select the coding mode and the quantization step size. The proposed method is applied to MPEG-4 codec, and the experimental results show that it is robust to the error such as packet losses comparing with the conventional ones.

Keywords : video coding, error resilience, intra update

I. 서 론

동영상은 정지 영상이 연속된 시간의 변화에 따라 연결된 것으로서 방대한 정보량을 지닌다. 이러한 동영

상을 저장하거나 전송하기 위해서는 메모리 또는 전송 대역폭의 제한이 따르게 되므로 방대한 양을 가진 동영상을 어떻게 압축하여 효과적으로 복원해 내는 것이 중요한 과제가 되어 왔다. 이에 따라 1988년부터 MPEG(Moving Picture Experts Group) 위원회가 결성되고 동영상 압축의 국제 표준안이 제정되어 왔다. 특히 최근 국제 표준안으로 제정된 MPEG-4에서는 기존의 MPEG-1, MPEG-2에서 저장 매체 또는 디지털 방송을 위해서 동영상을 압축했던 것을 포함함은 물론 객체 부호화, 오류 내성, 컴퓨터 그래픽스 등의 새로운 방법을 도입시켜 진보된 형태의 멀티미디어로의 발전

* 正會員, 西江大學校 電子工學科

(Dept. of Electronic Eng., Sogang Univ.)

※ 본 연구는 서강대학교 산업기술연구소에서 지원되었음

接受日字:2002年2月6日, 수정완료일:2002年10月23日

을 피하였다.^{11,12)}

이 중 오류 내성 방법은 압축된 동영상을 전송할 때에 발생하는 오류에 의해서 손상되는 정보를 효과적으로 복원할 수 있도록 고안된 것이다.¹³⁻¹⁵⁾ 동영상을 압축하기 위해서는 공간, 시간상에서의 중복성을 최대한 제거해야 하는데 이를 위해서 예측부호화 및 가변장 부호화를 사용한다. 이러한 부호화 방법을 통해 효율적으로 정보를 압축할 수 있지만 만약 정보에 오류가 발생할 경우 오류가 난 시점의 정보만 손상을 입는 것이 아니라 이후의 정보도 치명적인 손상을 입을 수 있다는 단점이 있다. MPEG-4에서는 이러한 문제점을 극복하기 위해 재동기화, 자료 분할, 역가변장 부호화, 그리고 인트라 업데이트의 4가지 오류 내성 방법을 사용하고 있다.¹⁶⁻¹⁸⁾

본 논문에서는 위의 오류 내성 방법 중 인트라 업데이트를 효과적으로 할 수 있는 방법을 제시한다. 인트라 업데이트는 움직임 예측을 사용한 부호화에서 어느 한 프레임에 오류가 발생한 경우 그 이후의 프레임으로 오류가 계속 확산되는 것을 막기 위해서 특정한 부분을 움직임 예측 없이 부호화하는 인트라 모드로 부호화하는 것이다.

일반적으로 전송 대역폭, 또는 저장 공간의 한계 등에 의해 압축률이 제한되는데 이 경우 화질을 떨어뜨림으로써 압축률을 높이는 방법이 사용된다. 그러므로 주어진 압축률에서 보다 좋은 화질을 얻기 위한 방법이 요구되는데 이러한 방법을 비트율 제어라고 한다. 일반적으로 비트율 제어 방법은 영상의 고주파 영역을 제거함으로써 이루어지는데 이는 사람의 눈이 고주파 보다는 저주파 신호에 민감하기 때문이다.

영상 신호를 주파수 영역으로 변환하기 위해서 일반적으로 이산여현변환(DCT; discrete cosine transform)을 사용한다. DCT에 의해 영상 정보는 주파수 신호로 변환되고 대부분의 정보는 저주파 성분에 몰리게 되므로 정보량이 많지 않은 고주파 성분을 제거한 후 역변환함으로써 압축의 효율을 높일 수 있다. 고주파 성분을 제거하기 위해서는 양자화 방법이 사용되는데, 양자화 파라미터가 클수록 압축률이 높아지는 반면 복원된 영상의 질은 떨어지게 되고 양자화 파라미터가 작을수록 그 반대가 된다.¹⁹⁾ 비트율 제어는 양자화 파라미터를 효율적으로 선택함으로써 압축률과 화질간의 최적화를 이루도록 한다.¹⁰⁻¹²⁾

한편 인트라 모드로 부호화할 경우 압축률이 떨어지

는 반면 오류에 의한 영향이 줄어들어 복원된 화질의 향상이 기대된다. 그러므로 비트율과 오류의 영향을 적절하게 고려하여 최적화된 부호화 모드의 선택이 필요하게 된다. 본 논문에서는 부호화의 기본 단위가 되는 매크로블록(MB; macroblock) 단위로 비트 양과 양자화 파라미터, 오류에 의한 왜곡을 분석하여 최적화된 부호화 모드 선택 및 비트율 제어를 이루는 방법을 제안한다.

II. 동영상 부호화 모드 선택

MPEG-4에서 동영상은 동영상 내의 각각의 물체가 객체(video object)로 정의되어 부호화된다. 한 장의 영상(VOP; video object plane) 내에서 원하는 객체만을 부호화하기 위해 모양 부호화가 사용되고, DCT 및 양자화와 움직임 보상을 기반으로 하여 부호화가 이루어진다. MPEG-4에서는 여러 가지 부호화 모드를 제공하여 효율적으로 부호화가 이루어질 수 있도록 하는데 부호화 모드는 VOP 단위로 MB 단위로 이루어진다. VOP는 부호화 방법에 따라 I-, P-, B-, 그리고 S-VOP로 나누어진다. I-VOP는 움직임 보상을 하지 않고 하나의 VOP를 모두 인트라 모드로 코딩하는 것이고 P-VOP는 움직임 보상을 하되 MB에 따라 인트라 또는 인터 모드로 부호화하는 것이다. B-VOP와 S-VOP는 각각 양방향 움직임 예측과 스프라이트를 사용한 것이다.

표 1. VOP의 종류에 따른 MB 부호화 모드
Table 1.

VOP 종류	MB 형태 인덱스	MB 형태 이름
P	not coded	-
P	0	INTER
P	1	INTER+Q
P	2	INTER4V
P	3	INTRA
P	4	INTRA+Q
P	stuffing	-
I	3	INTRA
I	4	INTRA+Q
I	stuffing	-

MPEG-4에서 동영상 부호화의 기본 단위는 MB로 16×16 화소 크기의 어느 한 지역의 밝기 정보와 칼라

정보를 나타내기 위해 사용된다. <표 1>은 VOP의 종류에 따른 MB의 여러 가지 형태를 나타낸 것인데 B-VOP와 S-VOP는 생략하였다.

<표 1>에서 MB 형태가 INTRA인 경우 움직임 보상을 하지 않고 인트라 모드로 부호화한다. 그리고 INTER인 경우에는 하나의 MB에 하나의 움직임 벡터를 사용하여 움직임 보상을 한다. INTER4V는 하나의 MB에서 각각의 8×8 블록이 하나의 움직임 벡터를 사용한다.

그리고 +Q가 붙을 경우에는 양자화 파라미터 변경 방법을 사용하는데, 이 방법은 하나의 VOP당 양자화 파라미터의 값 QP가 고정되어 있는 것을 각 MB 별로 QP 값을 변화시킬 수 있도록 고안된 것으로 DQUANT라는 값이 추가적으로 비트열에 포함된다. DQUANT 값은 (-2, -1, 1, 2)의 4가지 값을 가지며 각 MB의 양자화 파라미터는 QP에 DQUANT 값을 더한 값을 사용한다. 이 방법을 통해 각 MB 별로 비트율 제어를 하는 것이 가능하다.

그 외에 stuffing으로 표시된 것은 부호화 양이 너무 작아서 인코더 버퍼에서 버퍼 모자람이 발생하게 되면 반대로 디코더 버퍼에서 버퍼 넘침이 발생하게 되므로 이를 방지하기 위해서 인코더에서 의미 없는 비트를 삽입하고 디코더에서 버리도록 한 것이다.

MPEG-4에서는 이와 같이 MB 형태에 따른 다양한 부호화 모드를 지원함으로써 효과적으로 부호화할 수 있도록 하였는데, 어떤 경우에 어느 모드를 사용할 것인가는 최적화된 부호화에 매우 중요한 영향을 미친다. 다음에서는 MPEG-4 verification model (VM)^[13]에서 사용한 MB 형태 선택 방법에 대해 설명한다.

MPEG-4 VM에서 부호화 모드는 움직임 예측을 하는 과정에서 선택한다. 먼저 현재 블록과 가장 유사한 블록을 이전 영상에서 찾기 위해

$$SAD_N(x, y) = \sum_{i=1, j=1}^{N, N} |p_{i,j} - q_{i+x, j+y}| \quad (1)$$

로 정의되는 SAD를 계산한다. 여기서 N은 16 또는 8의 값을 가지며 16일 경우 INTER, 8일 경우 INTER4V의 MB 형태를 말한다. 그리고, x, y는 현재 블록과 이전 영상의 블록의 변위를 나타내며 범위는 [-64, 63]이고 p_{i,j}와 q_{i+x, j+y}는 각각 현재 블록과 이전 영상의 블록의 화소의 밝기 값을 나타낸다. 만약 움직임 벡터가 영 벡터이면, 즉, x=y=0이면 부호화의 효

율이 높아지므로

$$SAD_{16}(0, 0) = SAD_{16}(0, 0) - (N_B/2 + 1) \quad (2)$$

와 같이 편차를 둔다. 여기서 N_B는 해당 MB의 화소의 개수를 의미한다. 이 과정에서 SAD₁₆ 또는 한 MB에서 4개의 SAD₈의 합 중 가장 작은 값을 나타내는 벡터 (x, y)가 움직임 벡터가 되며 이 때의 SAD를 SAD_{inter}라고 한다.

인트라와 인터 모드를 결정하기 위해서 현재 MB의 편차 A를

$$A = \sum_{i=1, j=1}^{16, 16} |p_{i,j} - \frac{1}{N_c} \sum_{x=1, y=1}^{16, 16} p_{x,y}| \quad (2)$$

와 같이 구한다. 여기서 N_c는 화소의 개수이다. 이 때 만약

$$A < SAD_{inter} - 2 \times NB \quad (3)$$

이면 INTRA 형태의 MB로 부호화한다. 만약 INTRA 형태의 MB로 결정되면 INTER4V를 결정하기 위한 과정은 생략된다.

만약 INTER 형태의 MB이면 SAD₁₆과 SAD₈의 합 ΣSAD₈을 비교하여 만약

$$\Sigma SAD_8 < SAD_{16} - (N_B/2 + 1) \quad (5)$$

이면 INTER4V 형태의 MB로 부호화한다.

이상과 같이 MPEG-4에서는 여러 가지 모드로 부호화할 수 있도록 하고 MPEG-4 VM을 통해 효율적으로 부호화할 수 있는 방법을 제시하였다.

III. 인트라 업데이트 및 비트율 제어 알고리즘

DCT와 움직임 보상을 기반으로 한 동영상 압축 알고리즘은 압축 효율에 있어서는 뛰어난 성능을 보이지만 오류에는 약하다는 단점이 있다. 특히 움직임 보상을 사용하는 인터 모드로 부호화할 경우 오류가 발생하면 오류가 발생한 지역뿐 아니라 그 이후에도 계속적으로 오류가 확산되는 단점을 가지고 있다.

이것을 효과적으로 막을 수 있는 방법이 인트라 업데이트인데 이 방법을 사용할 경우 움직임 보상 방법

을 사용하지 않기 때문에 압축 효율이 떨어지게 된다. 그러므로 오류 환경에서 효과적으로 부호화하기 위해서는 비트율 제어를 고려한 인트라 업데이트가 필요하다.

다음에서는 기존의 인트라 업데이트 방법과 비트율 제어 알고리즘에 대해 살펴본다.

1. 기존의 인트라 업데이트 방법

MPEG-4에서 사용하는 인트라 업데이트 방법으로는 adaptive intra refresh (AIR) 방법이 있다.^[6] 이 방법에서는 하나의 VOP에서 인트라 모드로 부호화될 MB의 개수를 고정된 값으로 미리 정해놓고 움직임이 많은 지역을 인트라 모드로 부호화하는 방법이다. 움직임이 많은 지역은 오류 은닉에 의한 오차가 크기 때문에 이 지역을 인트라 모드로 부호화하는 것이다. 움직임이 많은 지역은 움직임 예측 과정 중에 계산된 한 MB의 SAD의 값이 정해진 문턱치를 넘어서면 움직임이 많은 MB가 되고 이러한 MB를 인트라로 부호화할 수 있도록 정해진 개수 안에서 차례대로 인트라 모드로 부호화한다. 이 방법은 Haskell과 Messerschmitt^[8]에 의해 제안된 방법과 유사한데 그들의 방법에서는 움직임이 많은 지역을 부호화하되 모든 지역이 인트라 모드로 부호화될 수 있는 주기를 오류 발생률의 역수보다 작도록 하였다.

Liao와 Villasenor^[14]는 H.263^[15]에서 group of blocks (GOB) 헤더에 의한 재동기화를 고려하여 오류가 발생할 수 있는 위치를 GOB 헤더와 현재 이전의 블록 사이, 현재 블록, 그리고 현재 이후의 블록으로 나누어 각각의 왜곡을 추정한 후 왜곡이 가장 심한 블록을 인트라 모드로 부호화하는 것이다. Stuhmüller 등^[16]은 디코더에서의 왜곡을 인코더에서의 왜곡과 전송시의 오류에 의한 왜곡으로 분류하고 각각의 왜곡을 인코더에서 부호화시에 인트라 모드로 부호화할 MB의 비율에 관한 식으로 나타내어 왜곡을 최소화시키는 비율을 결정하는 방법을 제안하였다.

Zhang 등^[17]은 전송시의 오류 발생률과 움직임 벡터를 사용한 오류 은닉에 의한 영향을 고려한 인트라 업데이트 방법을 제안하였다. 이 방법에서는 H.263에서 화소 단위로 왜곡을 고려하되 오류가 발생하였을 경우 현재 프레임의 바로 앞의 GOB의 움직임 벡터를 사용하여 이전 프레임에서 오류 은닉을 수행한다. 이 때 바로 앞의 GOB에서도 오류가 발생하였다면 움직임 벡터

를 0으로 하여 오류 은닉을 수행한다. 그러므로 현재 화소의 왜곡을 계산하기 위해 오류가 발생하지 않았을 경우, 현재 GOB에서 오류가 발생하였을 때 이전 GOB에서 오류가 발생한 경우와 그렇지 않은 경우, 총 3가지 경우로 나누어 계산한다.

계산된 왜곡의 값은 오류 환경을 고려하지 않은 비트율 제어 방법에 사용된 라그랑지안 최적화 방법에 적용되어 각 MB별로 라그랑지안 함수 J_{MB} 를 최소화시키는 모드를 선택한다. 이 과정을 식으로 나타내면

$$\min_{mode} (J_{MB}) = \min_{mode} (D_{MB} + \lambda R_{MB}) \quad (4)$$

와 같다. 이 때 D_{MB} 는 현재 MB의 각 mode별 왜곡 값, R_{MB} 는 현재 MB의 각 mode별 부호화 양의 미하고 mode로는 인트라 모드와 인터 모드가 사용된다. λ 는 라그랑지안 승수이다. 이 방법에서는 비록 정밀하게 왜곡을 계산할 수 있지만 왜곡을 계산하기 위해서 부호화 과정에서 왜곡을 추정하지 않고 디코딩된 화소의 값을 사용하기 때문에 많은 계산량을 필요로 한다.

Côté와 Kossentini,^[18] 그리고 Wenger와 Côté^[19]의 방법에서는 왜곡을 계산하기 위해서 오류가 일어났을 경우와 일어나지 않았을 경우로 나누어 생각하여 real time protocol (RTP) 상에서 H.263를 사용한 경우에 적용한 것이다. 왜곡을 계산할 때에 오류가 발생하지 않았을 경우에는 양자화에 의한 왜곡의 값을 사용하고 오류가 발생하였을 경우에는 오류 은닉에 의한 왜곡을 사용한다. 계산된 왜곡의 값은 Zhang 등의 방법에서와 같이 라그랑지안 함수

$$J = (1 - p)D_q + pD_c + \lambda R \quad (5)$$

를 최소화시키는 과정을 거쳐 최적화된 모드를 결정한다. 이 때 p 는 한 MB에서 오류가 발생할 확률을 나타내며 D_q , D_c 는 각각 양자화와 오류은닉에 의한 왜곡을 나타낸다. 그리고 R 은 MB의 부호화 양이다.

Côté 등^[20]의 방법에서는 위에서 왜곡을 계산할 때에 오류가 발생한 경우와 오류가 발생하지 않았을 경우로 양분한 것을 더 발전시켜서 오류가 발생하지 않았을 경우에 움직임 보상을 하기 위해 사용되는 이전 영상의 지역이 오류가 발생하였을 경우를 추가하였다.

이상의 방법에서는 주로 오류에 의한 왜곡을 고려하

는 방법을 사용하였으며 대부분 H.263에 적용하였다. 비록 비트율 제어와 연관된 알고리즘들이 제안되었으나 라그랑지안 최적화 과정에서 왜곡에 대한 항만 오류를 고려한 왜곡으로 바뀌었을 뿐이다. 그리고, 오류 은닉에 의한 왜곡의 값을 추정하는 방법에 대해서도 자세히 다루고 있지 않다.

2. 기존의 비트율 제어 방법

부호화 과정에서 많은 비트를 할당할수록 화질은 좋아지고 적은 비트를 할당할수록 화질은 나빠지게 된다. 그런데 대부분의 전송 환경 또는 저장 매체는 전송 대역폭 또는 저장 공간의 제한으로 인해 할당할 수 있는 비트 수에 제한이 있다. 비트율 제어 알고리즘은 제한된 비트 수 안에서 보다 좋은 화질을 얻기 위한 방법이다.

비트 수와 왜곡의 관계는 실험을 통해 알 수 있는데 실험을 통해 얻어진 동작점들은 하나의 곡선을 이루게 된다. 이 곡선을 추정하여 비트율과 왜곡간의 모델을 만들 수 있다. 이 곡선에서 주어진 비트율에 따른 최적화된 부호화를 수행할 수 있는데 이를 위한 방법으로는 라그랑지안 최적화 방법과 동적 프로그래밍 방법이 있다.^[10,11,21,22]

Ribas-Corbera와 Lei^[12]는 저지연 통신을 위한 DCT 기반 동영상 압축에서의 비트율 제어 알고리즘을 제안하였는데 이 방법에서 비트 수와 왜곡은 각각 Q_i 에 대한 수식으로 나타난다. 먼저 i 번째 MB에서의 비트 양 B_i 는

$$B_i = A \left(K \frac{\sigma_i^2}{Q_i^2} + C \right) \quad (6)$$

로 표현되는데 이 때 A 는 MB의 화소의 수이고 K 는 상수로서 DCT 계수가 라플라시안 분포를 가진다면 $e/\ln 2$ 의 값을 갖는다. 그리고 σ_i 와 Q_i 는 각각 해당 MB에서의 표준편차와 양자화 파라미터이다. C 는 양자화된 DCT 계수를 부호화하는 데 필요한 비트 수 외에 움직임 벡터와 다른 구문들을 부호화하기 위해 필요한 비트 수이다. 그리고 DCT 계수를 양자화했을 때의 왜곡 D 는

$$D = \frac{1}{N} \sum_{i=1}^N a_i^2 \frac{Q_i^2}{12} \quad (7)$$

와 같이 된다. 이 때 N 은 MB의 개수이고 a_i 는 각

MB의 왜곡 가중치이다. 계산된 비트 수와 왜곡의 값은 라그랑지안 함수를 최소화시키는 Q_i 를 찾기 위해 사용된다.

한편 인트라 모드와 인터 모드의 결정은 비트율 제어에 많은 영향을 미치기 때문에 비트율 제어와 모드 결정을 함께 고려한 방법이 제안되었다.^[23-26]

먼저 Lee와 Dickinson의 방법^[23]에서는 라그랑지안 함수 J 를

$$J(N, P, Q, \lambda) = D(N, P, Q) + \lambda R(N, P, Q) \quad (8)$$

와 같이 정의한다. 여기서 N 은 인터 프레임의 개수, P 는 인터 프레임의 위치, Q 는 양자화 파라미터이다. 이 함수를 최소화하기 위해서 먼저 N, P 를 고정된 값으로 보고 $J(Q)$ 를 최소화시키는 Q 를 찾는다. 이 과정에서 동적 프로그래밍을 사용하여 이전 프레임이 현재 프레임에 미치는 영향을 고려하였다. 그 다음 Q 를 고정시킨 후 $J(N)$ 을 최소화시키는 N 을 찾은 후 P 의 값을 결정한다.

Wiegand 등의 방법^[24]과 Schuster와 Katsaggelos의 방법^[25]에서는 현재 MB와 그 바로 전후의 MB의 각 모드별로 라그랑지안 함수 값을 계산하고 각각을 동적 프로그래밍을 사용하여 트래블리스로 배치하고 라그랑지안 함수를 최소화시키는 경로를 찾는 방법을 사용하였다. 그리고, Wiegand과 Andrews의 방법^[26]에서는 각 모드별로 라그랑지안 함수를 계산하되 움직임 보상을 사용할 때와 사용하지 않을 때 라그랑지안 승수를 각각 λ_{mode} 와 λ_{motion} 으로 나누어 사용하여 모드 선택을 하도록 했다. 이 때 각각의 값은 각 MB의 양자화 파라미터 값을 사용하여

$$\begin{aligned} \lambda_{mode} &= 0.85 \times QP^2 \\ \lambda_{motion} &= 0.92 \times QP \end{aligned} \quad (9)$$

와 같이 계산할 수 있다.

IV. 비트율과 오류 내성을 고려한 모드 선택 최적화 알고리즘

앞에서 살펴본 바와 같이 인트라 업데이트 방법은 움직임 보상으로 인한 오류의 확산을 막는 데 효과적이다. 반면에 오류가 발생하지 않았을 경우에는 부호화의 효율이 떨어진다는 단점이 있다. III장에서 기존의

여러 가지 인트라 업데이트 방법을 살펴보았는데 오류에 의한 왜곡을 고려하여 모드 선택을 결정하는 방법들을 볼 수 있었다. 그리고 이들 방법들은 H.263을 이용하여 실험되었다.

본 논문에서는 오류 환경으로 인해 생겨난 왜곡을 고려할 뿐만 아니라 MB 단위의 비트율 제어를 함께 고려하여 최적화된 부호화 모드를 선택하는 방법을 제안한다. MB 단위의 비트율 제어를 위해서 MPEG-4에 사용되는 양자화 파라미터 변경 방법^[6]을 사용하고 VOP 단위의 비트율 제어를 위해서는 MPEG-4 VM [13]에 제안된 방법을 사용한다.

만약 부호화에 의한 비트 양과 디코딩에 의한 왜곡을 정확히 알 수 있다면 정확한 비트율 제어가 가능하다. 그러나 정확한 값은 디코딩을 수행한 후에야 알 수 있다. 그러나, 여러 가지 모드와 여러 가지의 양자화 값을 결정하기 위해 이들 값을 사용하려면 많은 수의 디코딩 수행이 필요하게 되고 이러한 일은 너무 많은 계산량으로 인해 실용적이지 않다. 그러므로 이들 값을 미리 알고 있는 값으로 추정하여 사용하는 것이 필요하다.

먼저 인트라 모드로 부호화할 경우에 하나의 MB를 부호화하는 데 필요한 비트 수는 양자화 파라미터와 밀접한 관계를 가지며 이 관계는 식 (8)과 같다. 이 식을 사용하기 위해서 제안한 방법에서 QP 는 이전 프레임의 QP 를 사용하고 표준 편차의 값을 사용하는 대신에 움직임 보상의 과정에서 식 (3)을 통해 계산된 MB의 편차 값인 A 를 대신 사용한다. 즉, i 번째 MB를 부호화했을 때의 비트 양 $R_{i,intra}$ 는

$$R_{i,intra} = \frac{\sum_{j=1}^{16} \sum_{x=1}^{16} \left| p_{i,j} - \frac{1}{N_c} \sum_{x=1}^{16} \sum_{y=1}^{16} p_{x,y} \right|}{QP^2} \quad (10)$$

와 같다. 이 때 N_c 는 화소의 수, 그리고 $p_{i,j}$ 는 화소의 값을 나타낸다. 여기서 QP 는 [1, 31]의 범위를 갖는다. 식 (3)에서는 구문과 움직임 벡터를 고려한 상수를 더해주고 있지만 구문에 의한 비트 수는 인트라 모드와 인터 모드에서 모두 무시하고 인터 모드에서만 움직임 벡터를 부호화하기 위한 값을 더해준다.

그리고 i 번째 MB에서의 왜곡의 값 D_i 는 식 (9)를 이용하여

$$D_{i,intra} = \frac{QP^2}{12} \quad (11)$$

와 같이 계산한다.

두 번째로 인터 모드로 부호화했을 경우에 i 번째 MB를 부호화했을 때의 비트 양 $R_{i,inter}$ 는 인트라 모드에서 사용한 편차 A 대신에 식 (1)을 통해 계산한 SAD 를 사용하여 구한다. 즉,

$$R_{i,inter} = \frac{\sum_{j=1}^N \sum_{l=1}^N |p_{i,j} - q_{i,l}|}{QP^2} + C \quad (12)$$

와 같다. 이 때 $q_{i,l}$ 는 움직임 보상을 위해 사용되는 이전 영상의 화소 값이고 C 는 움직임 벡터를 부호화하기 위한 비트 수이다.

다음으로 인터 모드로 부호화했을 때의 i 번째 MB에서의 왜곡 $D_{i,inter}$ 는 움직임 보상을 위해 사용될 이전 영상의 지역에 오류가 발생하였을 경우와 오류가 발생하지 않았을 경우로 나누어 생각한다. 먼저 오류가 발생하지 않았을 경우에는 인트라 모드와 마찬가지로 양자화 파라미터를 사용하여 왜곡의 값을 추정하고 오류가 발생하였을 경우에는 움직임 벡터를 사용하여 오류를 하는 것으로 가정하여 움직임 예측을 했을 때 계산된 SAD 를 왜곡의 값으로 사용한다. 즉,

$$D_{i,inter} = (1 - P_{loss}) \frac{QP^2}{12} + P_{loss} \sum_{j=1}^N |p_{i,j} - q_{i,l}| \quad (13)$$

와 같다. 여기서 P_{loss} 는 패킷 오류 발생률을 말한다.

이상과 같이 이전 프레임에 사용된 양자화 파라미터 QP 와 움직임 예측 과정에서 계산된 A 와 SAD 의 값을 통해 계산된 인트라 모드와 인터 모드에서의 비트 양과 왜곡의 값은 $DQUANT$ 의 값에 따라 조정된 QP 의 값에 따라 각각 계산된다. 즉, $DQUANT$ 의 값은 -2, -1, 1, 2의 4가지이므로 $DQUANT$ 를 사용하지 않는 경우까지 포함하여 모두 5가지의 값으로 QP 의 값이 바뀌게 된다. 이 때의 비트 양과 왜곡의 값을 각각 $R_{i,mode}$, DQ , 그리고 $D_{i,mode}$, DQ 라고 하자. 이 때 $mode$ 는 인트라 모드와 인터 모드, 그리고 DQ 는 $DQUANT$ 에 의해 조정된 QP 값을 의미한다.

이와 같이 각각의 모드와 DQ 값에 따라 계산된 비트 양과 왜곡의 값을 이용하여 주어진 비트율에서 최적화된 모드와 DQ 값을 선택하기 위해 라그랑지안 최적화 방법을 사용한다. 즉, 다음과 같이 정의된 라그랑지안 함수

$$J = D_{i, mode, DQ} + \lambda_{mode} R_{i, mode, DQ} \quad (14)$$

를 최소화시키는 모드와 DQ 의 값을 선택한다. 이 때 λ_{mode} 는 식 (11)의 값을 사용한다.

제안한 방법에서 계산량을 살펴보면, 부호화시에는 움직임 예측 과정에서 식 (1)과 식(3)을 통해 이미 계산된 값을 사용하여 비트율과 왜곡을 추정하기 때문에 추가적인 계산량이 거의 필요하지 않다. 그리고 복호화시에는 이미 결정된 모드에 따라 복호화하므로 추가적인 계산량이 필요하지 않다.

V. 실험 결과 및 토의

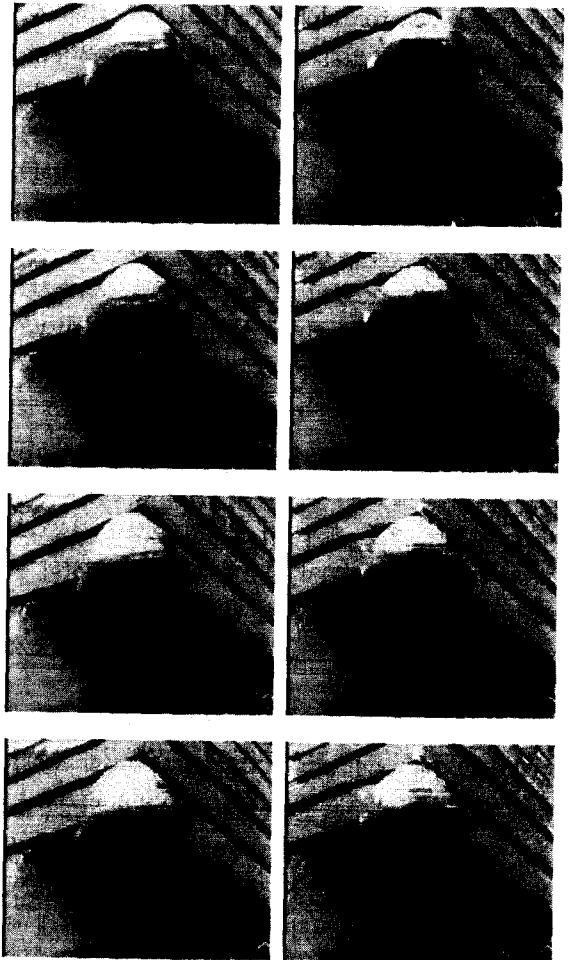
이번 장에서는 IV장에서 제안한 방법을 MPEG-4 simple profile에 적용하여 실험한 결과를 살펴보고 그에 대한 토의를 한다. 제안한 방법은 MPEG-4 VM⁽⁶⁾을 구현한 Mobile Multimedia Systems (MoMuSys) 코덱을 수정하여 구현되었다. MPEG-4의 오류 내성 방법 중에서 재동기화와 자료 분할 방법이 사용되었으며 디코더에서 움직임 벡터를 사용한 오류 은닉 방법이 사용되었다. 이 때 재동기화를 위한 패킷의 크기는 800 bit로 고정된 값을 사용하였다. 역가변장 부호화는 사용되지 않았다. 프레임 단위의 비트율 제어를 위해서는 MPEG-4 VM의 방법이 사용되었으며 움직임 부호화는 사용하지 않았다. 그리고, 부호화 모드로는 <표 1>에 나타난 MB 형태 중 INTRA, INTRA+Q, INTER, 그리고 INTER+Q가 사용되었다.

제안한 방법과의 비교를 위해서 위와 동일한 환경에서 MPEG-4 MoMuSys 코덱을 사용하여 실험을 하였다. 이 때 부호화 모드 선택은 II장에서 설명한 방법을 사용하였다. 즉, MPEG-4를 사용한 방법에서도 인트라 업데이트가 이루어지지만 이 방법은 부호화의 효율만을 고려한 것으로서 움직임 보상에 의한 왜곡과 부호화 양을 비교하여 부호화 모드를 결정하는 방법이다. 반면에 제안한 방법에서는 부호화 효율과 오류의 발생률, 그리고 오류 은닉에 의한 효과를 고려하여 인트라 업데이트를 한다.

오류 환경은 H.323 또는 무선 H.324 환경을 가정하고 비트 오류는 없고 패킷 손실이 발생한 경우를 고려하였고 패킷의 크기로는 100 byte를 사용하였다. 인터넷 환경에서와 같이 더 큰 크기의 패킷을 사용할 경우

패킷 손실에 의한 오류의 영향이 더 커지게 되어 제안한 방법의 효과가 더욱 커질 것으로 기대된다. 오류 발생은 난수발생기를 사용하였는데 이 때 seed 값은 제안한 방법과 기존의 방법에 동일한 값을 사용하였으며 seed 값을 50가지를 사용하여 하나의 비트열에서 패킷 손실이 일어난 비트열 50개를 만들었다. 실험 결과는 50개의 결과의 평균값이다.

실험 영상으로는 quarter common intermediate format (QCIF)의 Foreman 영상과 common intermediate format (CIF)의 Dancer 영상을 사용하였다. 각각은 176×144와 352×288 화소 크기를 가지며 색상 형태는 4:2:0을 사용하는데, 이 형태에서 하나의 MB

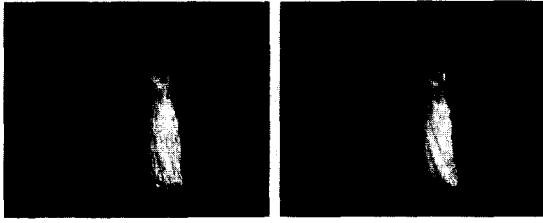


(a) 제안한 방법

(b) MPEG-4

그림 1. 오류 환경 하에서 Foreman 결과 영상.

Fig. 1. Comparison of the decoded images (Foreman sequence), (a) proposed method, (b) MPEG-4.



(a) 제안한 방법 (b) MPEG-4

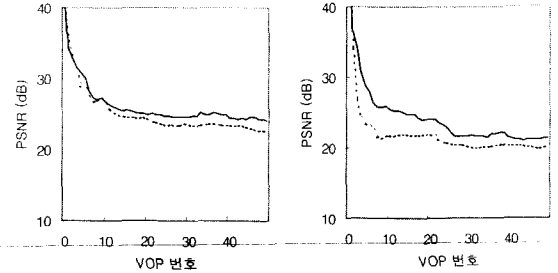
그림 2. 오류 환경 하에서 Dancer 결과 영상
Fig. 2. Comparison of the decoded images (Dancer sequence), (a) proposed method, (b) MPEG-4.

는 4개의 밝기 정보를 나타내는 8×8 화소 크기의 블록과 2개의 색상 정보를 나타내는 같은 크기의 블록으로 이루어진다. Foreman 영상은 움직임이 매우 심한 영상으로 전체적인 움직임과 함께 급격한 사람의 움직임이 나타난다. 실험에서는 보다 큰 움직임을 고려하기 위해 원래의 영상의 프레임 율의 1/3을 사용하여 부호화하였다. 즉, 원래 영상에서 두 개 건너 하나씩 부호화하였다. Dancer 영상은 중간 정도의 움직임을 가지며 원래 영상과 동일한 프레임 율을 사용하였다.

<그림 1>과 <그림 2>는 각각 Foreman 영상과 Dancer 영상을 제안한 방법과 MPEG-4 코덱으로 부호화하여 제작한 비트율에 1%의 패킷 손실을 발생시켜서 얻은 비트율을 디코딩한 영상이다. 이 때 비트율은 각각 100 Kbps와 400 Kbps를 사용하였다. 먼저 Foreman 영상을 살펴보면 MPEG-4로 디코딩된 영상에서는 오류가 발생한 지역에 계속해서 움직임 보상에 사용함으로써 오류가 확산되어 사람의 움직임이 정확히 표현되지 않았다. 반면에 제안한 방법에서는 인트라 업데이트를 사용하여 이러한 오류의 확산을 차단함으로써 사람의 표정이 변화하는 것이 나타난다.

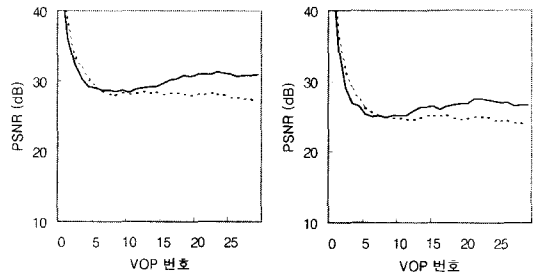
Dancer 영상에서는 Foreman 영상에 비해 움직임이 크지는 않지만 배경 부분에 작은 네모 모양의 무늬가 많이 있는 것을 볼 수 있다. 이것은 오류가 발생한 지역을 움직임 보상에 사용하는 일이 몇 개의 VOP에 걸쳐 계속해서 발생하여 생긴 오류의 확산이다. 반면에 제안한 방법에서는 전체적으로 깨끗한 영상 가운데 배경 부분에 보다 큰 네모가 보이는데 이 것은 MB 단위로 인트라 업데이트가 이루어져 생겨난 것이다. 비록 인트라 업데이트가 이루어진 VOP에서는 네모가 드러나게 보일 수 있지만 이 부분은 오류의 확산이 차단되어 올바르게 디코딩된 것이므로 이 부분을 움직임 보상

에 사용하는 이 후의 VOP에서도 올바르게 디코딩이 이루어질 수 있다.



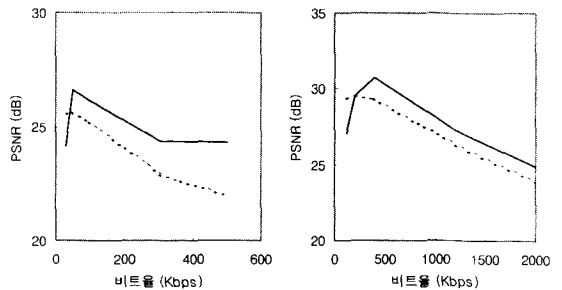
(a) 100 Kbps (b) 300 Kbps
(— 제안한 방법, - - - MPEG-4)

그림 3. Foreman 영상에서 비트율에 따른 결과.
Fig. 3. Comparison of the average PSNR as a function of the frame index (Foreman sequence), (a) 100 Kbps, (b) 300 Kbps.



(a) 400 Kbps (b) 1200 Kbps
(— 제안한 방법, - - - MPEG-4)

그림 4. Dancer 영상에서 비트율에 따른 결과
Fig. 4. Comparison of the average PSNR as a function of the frame index (Dancer sequence), (a) 400 Kbps, (b) 1200 Kbps.



(a) Foreman (b) Dancer
(— 제안한 방법, - - - MPEG-4)

그림 5. 비트율에 따른 제안한 방법과 MPEG-4의 결과 비교
Fig. 5. Comparison of the average PSNR as a function of the bit rate, (a) Foreman sequence, (b) Dancer sequence.

<그림 3>, <그림 4>, 그리고 <그림 5>는 각각 Foreman과 Dancer 영상을 사용하여 다양한 비트율을 사용하여 실험한 결과를 그래프로 나타낸 것이다. Foreman 영상은 30, 50, 100, 300 그리고 500 Kbps를 사용하여 각각 50장의 영상이 부호화되었고 Dancer 영상은 120, 200, 400, 1200, 그리고 2000 Kbps를 사용하여 각각 30장의 영상이 부호화되었다. 이 때 첫 번째 영상만 I-VOP로 부호화되고 나머지는 P-VOP로 부호화되었다. 패킷 손실률은 1%를 사용하였는데 50개의 각각 다른 오류 발생으로 만들어진 비트열을 디코딩하여 각 VOP별로 PSNR을 구한 후 50개의 평균값을 사용하여 그래프로 표현하였다. 각각의 비트율에 대해 제안한 방법과 MPEG-4의 방법이 비교되어 나타나 있다.

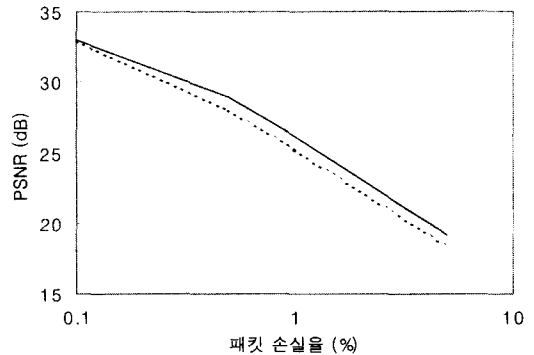
<그림 5>에서 실험 결과를 살펴보면 Foreman에서 30 Kbps, 그리고 Dancer의 120 Kbps에서 각각 MPEG-4보다 좋지 못한 성능을 보이고 있는데 그 이유는 인코더 버퍼가 가득 차서 버퍼 오버플로우가 발생하는 것을 막기 위해서 프레임 스킵이 일어났기 때문이다. 프레임 스킵이란 부호화 양을 감소시켜 버퍼가 넘치지 않게 하기 위하여 하나의 프레임을 이전의 프레임으로 그대로 대체하여 사용하는 것을 말한다. 그러므로 이 부분에서 PSNR이 크게 감소한 것이다. 이보다 약간 높은 비트율인 Foreman의 50 Kbps와 Dancer의 200 Kbps에서도 마지막 부분에 마찬가지로 프레임 스킵이 일어났다. 그러나 전반적인 PSNR은 인트라 업데이트로 인한 오류 확산의 차단으로 MPEG-4 방법보다 더 좋은 성능을 보이고, 그보다 더 높은 비트율에서는 더 좋은 성능을 보인다. 그리고 대체적으로 영상 시퀀스의 앞부분에서는 제안한 방법이 MPEG-4 보다 약간 성능이 좋지 않은 것을 볼 수 있다. 이것은 인트라 업데이트로 인해 압축의 효율이 떨어졌기 때문인데 뒤로 갈수록 오차에 의한 영향이 커짐에 따라 인트라 업데이트에 의한 효과가 떨어진 압축 효율을 넘어선 것을 볼 수 있다.

실험에서 얻은 PSNR 값을 <표 2>에 나타냈다. 실험 결과를 살펴보면 매우 낮은 비트율에서 프레임 스킵으로 인해 성능이 좋지 않은 것을 제외하고는 제안한 방법이 MPEG-4에 비해 더 좋은 성능을 보이는 것을 알 수 있다. 다만 그래프에서 비트율이 증가할수록 PSNR이 증가하지 않고 오히려 약간 더 떨어지는 추세를 보이는데 이것은 비트율이 높아질수록 텍스처 정보가 더 많아지게 되고, 즉 DCT 계수가 비트열에서 차지하는

비중이 커지게 되고 여기에 오류가 발생하였을 때 MPEG-4 코덱에서 오류 은닉을 효율적으로 하지 못하기 때문이다. 만약 역가변장 부호화라던가 텍스처 정보를 효율적으로 복원할 수 있는 오류 은닉 방법이 사용된다면 보다 더 좋은 성능을 기대할 수 있다.

표 2. 비트율에 따른 PSNR 비교
Table 2.

비트율 (Kbps)		PSNR (dB)		
		제안한 방법	MPEG-4	차
Foreman	30	24.15	25.58	-1.43
	50	26.61	25.63	0.98
	100	26.14	25.20	0.94
	300	24.29	22.90	1.39
Dancer	120	27.10	29.41	-2.41
	200	29.55	29.53	0.02
	400	30.75	29.39	1.36
	1200	27.31	26.37	0.94



(—— 제안한 방법, --- MPEG-4)

그림 6. 패킷 손실률에 따른 성능 비교
Fig. 6. Comparison of the average PSNR as a function of the packet loss rate (Foreman sequence, 100 Kbps).

표 3. 패킷 손실률에 따른 PSNR 비교
Table 3.

패킷 손실률 (%)	PSNR (dB)		
	제안한 방법	MPEG-4	차
0.1	33.05	32.94	0.11
0.5	28.98	27.99	0.99
1	26.18	25.20	0.98
5	19.17	18.33	0.84

다음에는 오류 발생률에 따른 제안한 방법의 성능을 MPEG-4의 성능과 비교하여 <표 3>과 <그림 6>에 나타내었다. 오류 발생은 패킷 손실을 가정하여 패킷 손실률을 0.1%에서 5%까지를 사용하였으며 실험 영상으로는 Foreman을 사용하고 100 Kbps의 비트율을 사용하여 부호화하였다. 결과에서 보이듯이 제안한 방법이 MPEG-4보다 더 좋은 성능을 보이는 것을 알 수 있다. 그래프에서 패킷 손실률을 나타내는 xx축은 로그 스케일로 표현되었다.

VI. 결론 및 추후 과제

인트라 업데이트는 움직임 보상에 의해 오류가 확산 되는 것을 막는 데 효율적인 방법이다. 반면에 움직임 보상을 사용하지 않기 때문에 압축 효율이 떨어진다. 본 논문에서는 비트율과 오류에 의한 왜곡을 고려한 인트라 업데이트 방법을 제안하였다. 제안한 방법에서는 비트율과 오류에 의한 왜곡을 움직임 보상 과정에서 미리 산출된 값을 사용하여 추정함으로써 빠른 계산이 이루어지도록 하였다. 이 값들과 미리 알고 있는 패킷 손실률을 사용하여 라그랑지안 함수를 최소화시키는 최적화된 부호화 모드와 양자화 파라미터를 선택하였다.

실험을 통해 기존의 MPEG-4 방법과 비교하였는데 MPEG-4에서는 비트율과 움직임 보상에 의한 왜곡을 비교하여 인트라 모드로 부호화하는 방법을 사용한다. 오류의 발생은 패킷을 기반으로 한 통신 환경에서 패킷 손실이 일어난 경우를 가정하였으며, 다양한 영상, 비트율, 그리고 패킷 손실률에 따라 실험하여 제안한 방법의 성능을 고찰하였다. 실험 결과를 통해 제안한 방법이 기존의 방법에 비해 더 좋은 성능을 가진다는 것을 보였다.

추후 과제는 제안한 방법에 버퍼 상태를 고려한 비트율 제어 알고리즘^[27]을 결합하는 것인데 이를 통해 제안한 방법의 낮은 비트율에서의 프레임 스킵으로 인한 성능 저하를 개선할 수 있다. 그리고 높은 비트율에서 PSNR 성능을 개선하는 오류 은닉 방법^[28]에 대한 연구이다.

참 고 문 헌

[1] ISO/IEC JTC1/SC29/WG11 N3342, *Overview of*

the MPEG-4 Standard (Noordwijkerhout Version), Mar. 2000.

- [2] T. Ebrahimi and C. Horne, "MPEG-4 natural video coding- An overview," *Signal Processing: Image Communication*, Vol. 15, nos. 4-5, pp. 365~385, Jan. 2000.
- [3] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, Vol. 86, No. 5, pp. 974~997, May 1998.
- [4] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Error resilient video coding techniques," *IEEE Signal Processing Mag.*, Vol. 17, No. 4, pp. 61~82, July 2000.
- [5] S. Wenger, G. Knorr, J. Ott, and F. Kossentini, "Error resilience support in H.263+," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. CSVT-8, No. 7, pp. 867~877, Nov. 1998.
- [6] ISO/IEC 14496-2, *Information Technology- Generic Coding of Audio-Visual Objects, Part 2: Visual (Final Draft of International Standard)*, Dec. 1998.
- [7] M. Budagavi, W. R. Heinzelman, J. Webb, and R. Talluri, "Combining efficiency and error robustness: Wireless MPEG-4 video communication on DSP chips," *IEEE Signal Processing Mag.*, Vol. 17, No. 1, pp. 36~53, Jan. 2000.
- [8] P. Haskell and D. Messerschmitt, "Resynchronization of motion compensated video affected by ATM cell loss," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, ICASSP'92*, Vol. III, San Francisco, CA, USA, Mar. 1992, pp. 545~548.
- [9] K. R. Rao and J. J. Hwang, *Techniques and Standards for Image, Video, and Audio Coding*, Prentice Hall, 1996.
- [10] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Processing Mag.*, Vol. 15, No. 6, pp. 23~50, Nov. 1998.
- [11] G. J. Sullivan and T. Wiegand, "Rate-distortion

- optimization for video compression," *IEEE Signal Processing Mag.*, Vol. 15, No. 6, pp. 74~90, Nov. 1998.
- [12] J. Ribas-Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. CSVT-9, No. 1, pp. 172~185, Feb. 1999.
- [13] ISO/IEC JTC1/SC29/WG11 N3312, *MPEG-4 Video Verification Model Version 16.0*, Mar. 2000.
- [14] J. Y. Liao and J. Villasenor, "Adaptive intra block update for robust transmission of H.263," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. CSVT-10, No. 1, pp. 30~35, Feb. 2000.
- [15] ITU-T Recommendation H.263, *Video Coding for Low Bit Rate Communication*, Jan. 1998.
- [16] K. Stuhlmüller, N. Färber, and B. Girod, "Adaptive optimal intra update for lossy video transmission," in *Proc. SPIE Visual Commun. Image Proc., VCIP 2000*, Vol. 4067, Perth, Australia, June 2000, pp. 286~295.
- [17] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Select. Areas Commun.*, Vol. SAC-18, No. 6, pp. 966~976, June 2000.
- [18] G. Côté and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the internet," *Signal Processing: Image Communication*, Vol. 15, nos. 1~2, pp. 25~34, Sep. 1999.
- [19] S. Wenger and G. Côté, "Using RFC2429 and H.263+ at low to medium bit-rates for low-latency applications," in *Proc. Packet Video '99*, New York, USA, Apr. 1999. Available from <http://spmg.ece.ubc.ca/pub/pvw99>.
- [20] G. Côté, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," *IEEE J. Select. Areas Commun.*, Vol. SAC-18, No. 6, June 2000.
- [21] F. S. Hiller and G. J. Lieberman, *Introduction to Mathematical Programming*, 2nd Ed., McGraw-Hill, 1995.
- [22] D. K. Smith, *Dynamic Programming- A Practical Introduction*, Ellis Horwood, 1991.
- [23] J. Lee and B. W. Dickinson, "Rate-distortion optimized frame type selection for MPEG encoding," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. CSVT-7, No. 3, pp. 501~510, June 1997.
- [24] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. CSVT-6, No. 2, pp. 182~190, Apr. 1996.
- [25] G. M. Schuster and A. K. Katsaggelos, "Fast and efficient mode and quantizer selection in the rate distortion sense for H.263," in *Proc. SPIE Visual Commun. Image Proc., VCIP '96*, Vol. 2727, Orlando, FL, USA, Mar. 1996, pp. 784~795.
- [26] T. Wiegand and B. D. Andrews, "An improved H.263 coder using rate-distortion optimization," *ITU-T SG16/Q15 Document Q15-D-13*, Apr. 1998. Available from <ftp://standard.pictel.com/video-site>.
- [27] 이정우, 호요성, "MPEG-2 비디오 부호화를 위한 정확한 비트율 제어 알고리즘," *전자공학회논문지*, 제 38권 SP편, 제 2호, pp. 102~110, 2001년 3월
- [28] 서재원, 호요성, "디지털 TV 수신기 성능을 개선하기 위한 전송 오류 은폐 기법," *전자공학회지*, 제 28권, 제 11호, pp. 55~61, 2001년 11월

 저 자 소 개

金 祐 湜(正會員) - 第38卷 SP編 第1號 參照

朴 來 弘(正會員) - 第33卷 B編 第9號 參照