DBpia

실시간 H.264/AVC를 위한 적응적인 Unit-level 비트율 제어 기법

정회원 김 명 진*, 준회원 주 원 희*, 종신회원 홍 민 철*

Adaptive Rate Control in Unit-level for Real-time H.264/AVC

Myoung-Jin Kim* ***Regular Member***, Won Hee Joo* ***Associate Member***,
Min-Cheol Hong* ***Lifelong Member***

요 약

본 논문에서는 실시간 H.264/AVC를 위한 적응적인 유닛 단위의 비트율 제어 기법에 대해 제안한다. 부호화하고자 하는 프레임은 인접한 프레임들과의 밀접한 상관관계를 가지고 있으며, 프레임당 발생되는 비트량은 QP값과 영상의 특성에 따라서 발생한다. 제안 방식은 인접한 프레임들의 각 유닛별 통계적 특성을 이용하여, 예측된 영상의 복잡도에 따라 QP값에 의한 발생 비트량의 변화를 구하여 부호화하고자 하는 프레임내 각 유닛의 예측 비트량에 적용하고 이전 프레임들의 통계 정보인 QP와 발생 비트량과의 가중치를 이용하여 현재 프레임내의 각 유닛별 복잡도를 예측한다. 실험 결과를 통해 동일 비트율에서 PSNR 성능 저하없이 기존 방식 대비 99%이상 계산량 감소가 있음을 확인할 수 있었다.

Key Words : 비트율 제어, 선형 관계, 유닛 단위, H.264, 통계특성

ABSTRACT

In this paper, we propose an adaptive rate control in Unit-level for real-time H.264/AVC. For given QP, bits according to video characteristics, and current frame is close correlation between the adjacent frames. Using the statistical characteristic, we obtain change of occurrence bit about QP to apply the bit amount by QP from the video characteristic and applied in the estimated bit amount of the each unit of current frame. In addition, we use weight with QP and occurrence bit amount that is statistical information of encoded previous frames. Simulation results show that the proposed rate control scheme achieves time saving of more than 99% over JM 12.1 rate control algorithm. Nevertheless, PSNR and bit rate were almost same as the performances of JM.

Ⅰ. 서 론

H.264/AVC는 ITU와 ISO/IEC가 함께 진행하여 향상된 비디오 부호화 표준을 제공하기 위해서 개발된 국제 압축 표준으로서, 다양한 예측 부호화 기법과 엔트로피 부호화 방법들을 통해, H.264/AVC 비디오 부호화는 기존의 비디오 부호화 표준들과 비교하여 최대 50%까지의 비트율을 절감할 수 있어 보다 향상된 부호화 효율을 제공한다[1]-[8].

영상정보를 실시간으로 전송시, 영상의 특징과 종류에 따라 생성되는 비트량은 시간적으로 변화가 심할 수 있게 되는데, 이는 전송되는 채널의 특성상 매우 제한적이며, 고정된 대역폭을 가지고 있기 때문이다. 따라서 영상의 품질과 채널 용량 간의 균형

※ 본 연구는 서울시 산학협력 사업(10570)으로 구축된 서울 미래형콘텐츠 컨버전스 클러스터 지원 및 2009년도 교육과학기술부의 재원으로 한국과학재단의 지원 (No. 2009-0079776)으로 수행되었음.
* 숭실대학교 정보통신전자 공학부(webzealer@ssu.ac.kr, jinini@ssu.ac.kr, mhong@ssu.ac.kr)
논문번호 : KICS2009-12-642, 접수일자 : 2009년 12월 30일, 최종논문접수일자 : 2010년 1월 20일

www.dbpia.co.kr

을 조절하여 가장 좋은 부호화 성능을 제공하기 위해서는 부호화 변수들을 조절하는 것이 필수적이라 할 수 있다. 이와 같은 부호화 성능을 조절하기 위하여 H.264의 비트율 제어 알고리즘은 비트율-왜곡 최적화(RDO: Rate Distortion Optimization) 방법을 기반으로 하여 발생 비트와 양자화 왜곡의 비율을 효율적으로 조절함으로써 부호화 전에 각 프레임을 위한 적절한 QP(Quantization Parameter)를 결정하고, 각 부호화 모드의 비용을 측정하여 최소 비용의 모드를 최종 모드로 결정한다[9-13]. 그러므로 H.264/AVC에서의 비트율 제어 방식은 모드 결정 및 움직임 벡터 결정을 하기 이전에 최적의 QP를 선택해야 하는 문제점이 발생한다. 이러한 RDO와 현재 영상의 통계치 계산 간의 상호 의존적인 구조 때문에 H.264에서의 비트율 제어는 복잡한 형태가 된다. 이러한 문제를 해결하기 위해 해당 매크로 블록의 왜곡 값을 MAD(Mean Absolute Different)를 통해 이전 프레임의 매크로 블록으로부터 추정되는 기법이 사용된다. M. Jiang은 이런 MAD 기법을 개선하여, GOP 구간에서의 부호화된 P 프레임들의 평균 MAD값에 대한 현재 프레임의 예측된 MAD 율을 이용하여 프레임 복잡도를 예측하는 모델[15]과 이 MAD 율과 함께 부호화된 P 프레임들의 평균 PSNR 오차율을 이용하여 현재 프레임의 복잡도를 예측하여 개선된 성능을 보여주고 있다[14]. Z.G. L는 Quadratic R-D 모델[10]에서 제안된 2차 비트율 모델을 이용하여 유닛의 QP값을 예측한다. 여기서 유닛은 프레임, 슬라이스 또는 매크로블록이 될 수 있다. 현재의 유닛에 대한 MAD를 계산할 수 없기 때문에 이전에 부호화된 프레임의 MAD값을 선형 모델에 적용하여 현재의 MAD값을 예측한다. 그러나 이와 같은 MAD 기법들은 많은 계산량을 필요로 하기 때문에 실시간으로 서비스되는 동영상의 응용에 많은 제약을 갖게 된다.

본 논문에서는 이와 같은 문제점을 해결하기 위하여 QP와 부호화 이후에 발생되는 실제 비트량과의 선형적 모델을 기반으로 하여 기존의 방식보다 복잡도를 줄이는 방식을 사용함으로써, 적은 계산량으로도 프레임의 각 유닛의 발생 비트량을 정확히 예측하여 기존의 방식보다 비트율 제어 알고리즘의 전체적인 계산량을 줄이는 기법을 제안한다. 이 논문에서는 유닛 단위를 기반으로 한 CBR 비트율 조절 방법만을 논문의 주제로 고려하겠다.

본 논문의 구성은 다음과 같다. Ⅱ장에서는 기존 H.264/AVC의 비트율 제어 알고리즘에 대하여 소개하고, Ⅲ장에서는 QP와 발생 비트량과의 선형적 모델을 기반으로, 프레임내 각 유닛의 복잡도를 예측하여 적은 계산량으로도 효율적인 비트량을 할당하는 알고리즘을 제안한다. Ⅳ장에서는 기존 비트율 제어 알고리즘과 제안한 비트율 제어 알고리즘의 시뮬레이션 결과를 비교 분석한다. 마지막으로 Ⅴ장에서는 본 논문의 결론을 맺는다.

Ⅱ. H.264/AVC의 비트율 제어 알고리즘

H.264/AVC를 효율적인 비디오 부호화 표준으로 만드는 많은 기술들 중 하나로, 율-왜곡 최적화(Rate-Distortion Optimization) 방법을 들 수 있는데, 움직임 추정에 쓰이는 가변블록 모드와 공간 예측 모드(Intra16x16, Intra4x4), Skip 모드 등 모든 가능한 부호화 모드 중에서 가장 최적의 모드를 선택하는 방법이다. 이 방법은 효율화의 장점과 함께 연산량 증가에 의한 부호화기 복잡도의 증가라는 단점이 있다[16]. 또한 율-왜곡 최적화 방법은 비트율 조절의 측면에서도 매우 다루기 까다로운 방법으로, 율-왜곡 최적화를 위해서는 모든 부호화 모드를 재귀적으로 검사를 하면서 최적의 모드를 선택하는데, 최적화가 끝난 후에 QP의 선택을 위해 사용되는 여러 변수들이 결정된다.

그림 1은 H.264/AVC 참조 모델의 비트율 제어 방법을 나타낸 것으로 목표 비트율이 할당되면 GOP 구간에서의 비트량이 결정된다. 그리고 부호화기의 버퍼 상태를 고려하여 각 유닛 단위별 비트량과 양자화 계수를 결정하여 부호화한다. 그런 다음 비트율-왜곡 최적화를 수행한 후, 화질과 비트율

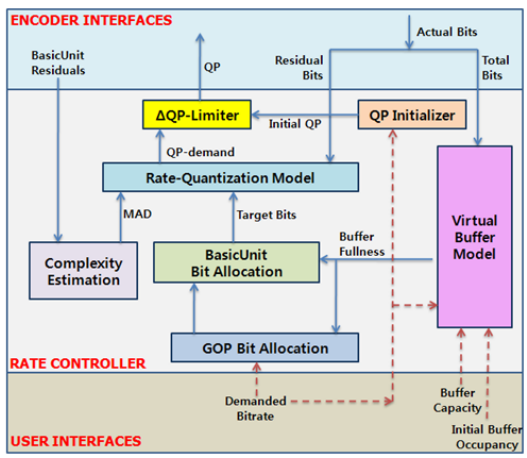

그림 1. H.264/AVC 참조 모델의 비트율 제어 방법

www.dbpia.co.kr

을 모두 만족하는 매크로블록 모드를 부호화한다.

2.1 프레임에 대한 목표 비트량 계산

비디오 시퀀스는 GOP(Group of Picture)로 구성되어 있고, GOP의 첫 번째 프레임은 인트라(Intra) 모드로 부호화하고, 그 이후의 프레임은 인터(Inter) 모드로 부호화하는 것이 일반적이다. 프레임 당 비트율을 제어하기 위해서는 먼저 첫 번째 프레임에 대한 가상버퍼 점유량(Virtual Buffer Occupancy)을 정해야 하는데, 이 첫 번째 가상 버퍼 점유량이 B_c(1)이면, 이때 초기 가상버퍼 점유량은 버퍼 사이즈(B_s)를 1/8로 설정한다. 그리고 버퍼의 넘침(Overflow)과 고갈(Underflow)을 방지하기 위하여 버퍼의 상한($U(1)$) 및 하한($L(1)$)의 초기값을 다음과 같이 설정한다.

$$\begin{aligned} L(1) &= \frac{u}{F_r} \\ U(1) &= \alpha \times B_s, (\alpha = 0.8) \end{aligned} \quad (1)$$

이때 u은 채널 대역폭으로 압축된 영상 시퀀스를 일정하게 전송하는 비트율이고, F_r은 초당 부호화된 프레임 율이다.

영상 시퀀스가 IPPP와 같은 GOP 구조로 부호화된다면 GOP의 첫 번째 I 프레임과 첫 번째 P 프레임은 미리 설정된 초기 양자화 파라미터로 부호화되기 때문에 목표 비트 할당이 요구되지 않고 이후의 프레임부터 목표 비트 할당이 이루어진다. n번째 프레임을 부호화할 때마다 가상버퍼 점유량($B_c(n)$) 및 목표 버퍼레벨($TBL(n)$)은 각각 식 (1)과 식 (2)와 같이 유동적으로 변하는 유동 트래픽 모델(Fluid traffic model) 식을 따르고 있다.

$$\begin{aligned} B_c(n) = \min(\max(0, \\ B_c(n-1) + A(n-1) - \frac{u}{F_r}), B_s), \\ 3 \le n \le N \end{aligned} \quad (2)$$

$$TBL(n) = TBL(n-1) - \frac{TBL(2) - B_s/8}{N_P - 1} \quad (3)$$

식 (2)에서 $A(n-1)$는 $(n-1)$번째 프레임에 대한 부호화된 비트량이고, 식 (3)에서 N_p는 하나의 GOP 내에 있는 P 프레임의 개수이다. n번째 프레임을 부호화하기 이전에, 프레임 당 할당할 수 있는 목표 비트량을 예측함에 있어서 버퍼량의 예측이 필요하다. 이때 버퍼량 예측은 가상버퍼 점유량 및 목표 버퍼레벨을 이용하여 다음과 같이 예측할 수 있다.

$$T'_{buf}(n) = \frac{u}{F_r} + r(TBL(n) - B_c(n)),\ r = 0.7 \quad (4)$$

이렇게 프레임 당 버퍼량이 예측되면 버퍼 상태를 고려하여 식 (5)와 같이 버퍼의 상한(U)과 하한(L) 범위까지 고려하여 최종적으로 버퍼량($T_{buf}(n)$) 값을 결정한다.

$$T_{buf}(n) = \min(U(n), \max(L(n), T_{buf}(n))) \quad (5)$$

n번째 프레임에 대한 목표 비트량 $T(n)$는 식 (5)에서 예측한 버퍼량과 식 (6)에서 예측한 부호 비트량을 고려하여 식 (7)과 같이 최종적으로 결정된다.

$$T_r = \frac{R}{N-n} \quad (6)$$

$$T(n) = \beta \times T_r + (1-\beta) \times T_{buf}, (\beta = 0.5) \quad (7)$$

식 (6)에서 T_r은 프레임당 부호화할 수 있는 비트량으로 현재 남아 있는 비트량(R)과 앞으로 부호화할 프레임 개수 즉, 남은 프레임 개수($N-n$)를 이용하여 계산한다.

2.2 MAD 예측 및 양자화 계수 예측

비트율 제어를 위해 먼저 현재 프레임과 참조 프레임간의 차이를 나타내는 MAD를 이용하며, 현재 프레임의 MAD_c를 예측하기 위해 식 (8)과 같이 계산한다.

$$\begin{aligned} MAD_c &= C_1 \times MAD_p + C_2 \\ C_1 &= \frac{1}{20}\sum_{i=1}^{20}\frac{MAD_i}{MAD_{i-1}} \end{aligned} \quad (8)$$

식 (8)에서 MAD_p는 이전 프레임의 MAD를 나타내며 C_2는 예측된 MAD값들의 예측 오차에 대한 보상 값이다. C_1과 C_2에 대한 초기값은 각각 1과 0이다. 현재 H.264/AVC의 비트율을 제어하기

www.dbpia.co.kr

위한 프레임 단위에 대한 양자화 계수는 식 (9)와 같이 2차 방정식 비트율-왜곡 모델[19]을 사용한다.

$$T(n) = \frac{x_1 \times MAD_n}{QP} + \frac{x_2 \times MAD_n}{QP^2} \qquad (9)$$

식 (9)에서 $T(n)$는 n번째 프레임에서의 목표 비트량, MAD_n는 n번째 프레임에서의 MAD, QP는 양자화 계수, 그리고 x_1와 x_2는 각 프레임에서 얻어지는 2차원 비트율-왜곡값을 나타내는 변수이다.

Ⅲ. 제안하는 비트율 제어 알고리즘

H.264/AVC에서 사용되고 있는 비트율 제어 기법은 실제 부호화에 의한 최적의 부호화 파라미터를 찾는 것으로, 라그랑지안 최적화 기법을 이용하여 가능한 부호화 파라미터 조합 중 주어진 비트율 조건을 만족시키고 왜곡을 최소화 하는 것을 예측하는 방식을 사용하고 있다. H.264/AVC에서 사용되는 비트율 조절 방법[13]은 비트와 왜곡이 양자화 스텝의 함수로서 표현되는 Quadratic R-D 모델에 기반을 두고 있다[10]. 이 방법은 이미지에 대한 분석의 간단함 및 양자화에 대한 직접적인 조절 방법 등의 특성에도 불구하고, 비디오 부호화의 다양한 특성 때문에, 경우에 따라서는 비효율적인 비트율 조절 결과를 나타내는 경우가 있다. 이러한 방식은 가장 이상적인 방법이지만 모든 양자화 파라미터에 따른 실제 비트율과 왜곡 값을 구해야 하므로 많은 계산량의 문제점으로 인해 실시간 멀티미디어 전송시에 부적합하다. 영상 정보의 실시간 전송은 대상 응용에 맞는 비트량을 예측하는 과정에서 불필요한 계산량을 줄이고 영상의 왜곡을 최소화하는 것을 목적으로 한다.

본 방식에서는 비트량을 예측하는 과정에서의 계산량을 줄이기 위하여 통계적 모델에 기반한 비트율 제어 기법을 이용하여 대상 응용에 적합하게 비트량을 산출하고 영상의 왜곡을 최소화하여 영상 정보를 실시간으로 부호화하는 기법에 대해 기술한다.

그림 2는 제안 방식에서의 전체적인 계층 구조를 나타낸 것으로, 3개의 주요 처리 단계로 이루어져 있다: 1) 프레임에 대한 목표 비트량 및 QP값 계산, 2) 각 유닛별 QP값 결정 및 부호화, 3) 복잡도 갱신.

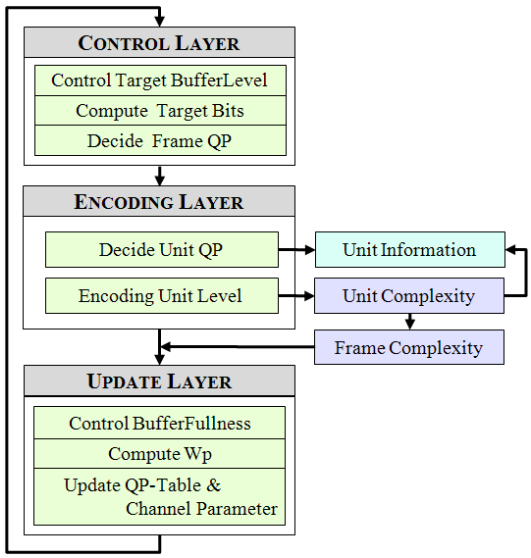

그림 2. 제안된 방식의 계층적인 구조

3.1 프레임에 대한 목표 비트량 및 QP값 계산

부호화 이후의 프레임에 대한 복잡도 갱신이 이루어진 정보들을 이용하여 현재 부호화 하고자 하는 프레임의 목표 비트량을 예측하는 단계로서, 부호화 하고자 하는 프레임에 대한 복잡도 가중치, 가상버퍼 점유율 그리고 목표 버퍼레벨을 고려하여 목표 비트량을 예측한다.

본 논문에서의 가상버퍼 점유량($B_c(n)$) 및 목표 버퍼레벨($TBL(n)$)은 각각 식 (10)과 식 (11)을 이용하여 구한다. 식 (10)에서 $A(n-1)$는 $(n-1)$번째 프레임에 대한 부호화된 비트량이고, 식 (11)에서 P_e는 GOP 구간에서의 부호화된 P 프레임 개수이다. 식 (11)과 같이 목표 버퍼레벨은 I 프레임이 부호화 된 이후의 첫 번째 P 프레임의 경우는 버퍼 점유율에서 목표 버퍼레벨을 줄이고, 첫 번째 P 프레임이 부호화 된 이후에는 이전 목표 버퍼레벨에서 값을 줄인다.

$$B_c(n) = A(n-1) - \frac{u}{F_r} + 0.5 \qquad (10)$$

$$TBL(n) = \begin{cases} TBL(n-1) - \dfrac{B_c}{N_P - 1}, & P_e > 1 \\ B_c - \dfrac{B_c}{N_P - 1}, & P_e = 0 \end{cases} \qquad (11)$$

현재 프레임을 부호화함에 있어서, 프레임에 할

www.dbpia.co.kr

당할 수 있는 목표 비트량을 예측하기 위해서는 다음 식 (12)와 같이 계산된 버퍼 점유량 및 목표 버퍼레벨을 이용하여 버퍼량의 예측을 수행한다.

$$T_{buf} = \frac{u}{F_r} + \gamma \times (TBL(n) - B_c(n)), (\gamma = 0.8) \quad (12)$$

그리고 프레임 당 버퍼량이 예측되면 버퍼 상태를 고려하여 식 (5)와 같이 버퍼의 상한과 하한 범위까지 함께 고려하여 최종적으로 버퍼량($T_{buf}(n)$) 값을 결정한다. n번째 프레임에 대한 최종 목표 비트량($T(n)$)은 식 (12)에서 예측한 버퍼량과 식 (13)에서 예측한 부호 비트량을 고려하여 식 (14)와 같이 최종적으로 결정된다.

$$\begin{aligned} T_{r1} &= \frac{W_p \times R}{N_p \times W_p} + 0.5 \\ T_{r2} &= \frac{u}{Fr} - \gamma \times (B_c(n) - TBL(n)) + 0.5) \end{aligned} \quad (13)$$

$$T(n) = \beta \times (T_{r1} - T_{r2}) + T_{r2} + 0.5 \quad (14)$$

W_p는 식 (21)을 통해 계산되며 현재 P 프레임의 복잡도 가중치를 예측한 값으로서, 본 논문에서는 이전에 부호화된 프레임들의 발생 비트량 가중치와 *QP*값으로 계산하였다. R은 현재 남아 있는 비트량이다. T_{r1}은 P 프레임에 대한 복잡도 가중치를 이용한 프레임당 부호화 할 수 있는 비트량이고 T_{r2}는 버퍼 점유량과 목표 버퍼레벨을 고려한 프레임당 부호화 할 수 있는 비트량이다. 식 (14)에서 β는 버퍼 점유량과 목표 버퍼레벨에 대한 의존도를 의미하는 것으로 실험에 의해 0.5로 설정하였다. 예측된 목표 비트량을 이용하여 부호화 하고자 하는 프레임의 *QP*값을 결정함에 있어서, 본 논문에서는 기존 H.264/AVC 방식의 식 (9)와 같은 *QP*변환 알고리즘을 사용하지 않고, 표 1과 같이 다양한 영상의 실험에 의한 통계 정보를 이용하여 결정하도록 하였다. 표 1은 영상의 각 QP에 따른 P 프레임들의 평균 비트량을 나타낸 것으로, 다양한 유형의 영상 통계 정보를 이용하기 위하여 빠르고 세밀한 영상의 "Mobile"와 "Stefan", 그리고 느리고 평탄한 영상의 "News"와 "Container"가 사용되었으며, 보통의 빠르기와 평탄함 및 세밀함이 골고루 있는 영상 "Foreman"이 사용되었다.

표 1. 각 *QP*에 따른 P 프레임의 평균 비트량

QP	container	foreman	mobile	news	stefan	QR
...						
17	12,849	20,692	47,388	7,136	42,849	24,559
18	10,621	17,164	42,426	6,103	38,352	21,819
19	9,091	14,943	38,998	5,426	35,062	19,301
20	7,212	12,277	34,350	4,645	31,002	16,823
21	5,285	10,522	30,895	4,108	27,926	14,893
22	5,007	9,018	27,715	3,636	24,809	13,095
23	4,006	7,575	24,176	3,153	21,849	11,313
24	3,224	6,358	21,084	2,749	18,947	9,905
25	2,659	5,538	18,972	2,458	17,057	8,593
26	2,093	4,572	16,009	2,100	14,467	7,317
27	1,704	3,921	13,845	1,856	12,594	6,303
...						

그림 3은 각 영상의 특성에 따라 *QP*값에 따른 P 프레임의 발생 비트량 특성을 나타내고, 표 1은 각 프레임의 비트량을 예측하기 위하여 산출된 실험 데이터이다. 실험(표 1)을 통하여 P 프레임들의 평균 비트량이 *QP*값과 밀접한 관계를 가지고 있음을 확인할 수 있으며, 그림 3과 같이 *QP*값에 따라 할당되는 비트량의 크기가 선형적으로 변하는 것을 보여주고 있다. *QP*값과 발생 비트량과의 선형 관계는 다음 식 (15)와 같이 유도될 수 있으며 발생된 P 프레임들의 평균 비트량과 거의 유사함을 알 수 있다.

$$QP_{bits,n} = \alpha \times e^{(\beta \times (QP_{n-1}+1))}, (1 \le n \le 51) \quad (15)$$

식 (15)의 n은 *QP*값 인덱스이며, $QP_{bits,n}$는 *QP*값 인덱스 n에 대한 예측된 발생 비트량을 의

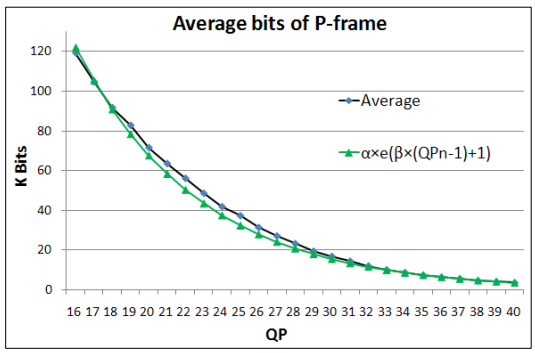

그림 3. 영상 특성에 따른 QP별 발생 비트량

www.dbpia.co.kr

미한다. α와 β는 그림 3과 표 1의 다양한 영상의 통계 특성에 의해 결정된 상수 값이다. 표 1에서의 $QR(QP\ Range)$은 현재 부호화하고자 하는 프레임에 할당 가능한 비트량 구간을 나타낸 것으로서, 양자화 인덱스 함수로서의 비트량이 가우시안 분포를 가질 수 있기 때문에 식 (15)의 QP 인덱스에 대한 예측 가능한 발생 비트량 $QP_{bits,n}$는 식 (16)과 같이 재계산될 수 있다.

$$QP_{bits,n} = QP_{bits,(n+1)} + \frac{|QP_{bits,n} - QP_{bits,(n+1)}|}{2}, \quad (0 \le n < 51) \tag{16}$$

이렇게 QP 인덱스에 대한 예측된 발생 비트량 $QP_{bits,(0\sim51)}$은 초기화 단계에서 한 번만 계산되며 1차원 테이블에 값을 저장하여 사용한다. 이후에는 식 (14)에서 계산된 목표 비트량을 이용하여 다음과 같이 현재 프레임에 대한 QP_c 값을 결정한다.

$$QP_c = frame\,QP(QP_{bits,n(1\sim51)}, T), \quad (1 \le n \le 51) \tag{17}$$

식 (17)에서 $frame\,QP()$는 QP 인덱스에 의한 예측 비트량 $QP_{bits,n(1\sim51)}$ 구간에서 해당 목표 비트량 T를 찾는 함수로써, 반복 루프를 통하여 구해진다.

3.2 각 유닛별 QP값 결정 및 부호화

프레임 내에서 처리되는 유닛의 기본 단위는 MB이며, 본 논문에서는 그림 4와 같이 영상의 가로폭의 크기로 고정하였다. 그림 4에 나타난 발생된 비트량에서 알 수 있듯이 이웃한 프레임들은 많은 유사성이 존재하며 이러한 유사성은 각각의 유닛별 특성에 따라 변화함을 알 수 있다. 즉, 그림 4는 유닛별 영상의 세밀함에 따라 QP값을 달리함으로써, 움직임이 거의 없는(Unit5~Unit9) 유닛에서의 QP값을 높이고, 움직임이 많은(Unit2~Unit4) 유닛에는 QP값을 낮추어 좀 더 많은 비트량을 제공하여 영상의 세밀함을 살릴 수 있음을 보여주고 있는 그림이다.

표 2는 그림 4의 100번째 프레임의 부호화 결과를 비교한 것으로, 제안된 방식이 JM보다 적은 비트량으로도 PSNR의 개선을 가져올 수 있음을 나타낸 것이다. 이는 앞에서 언급한 바와 같이 부호화 하고자 하는 프레임의 유닛별 특성 분포에 따라서 움직임이 있는 유닛에는 QP값을 낮추고, 움직임이 거의 없는 부분에서는 QP값을 높인 결과이기 때문이다.

식 (14)와 식 (17)의 결과로 프레임 목표 비트량(T)를 이용하여 프레임의 QP값이 결정되면, 프레임 내 각 유닛별 영상의 복잡도 가중치에 따라 각 유닛의 QP를 결정하여 부호화를 수행한다. 이 때 각 유닛에 대한 목표 비트량($T_{unit,i}$)는 식 (18)과 같이 계산된다.

$$uc = \frac{width/16}{MB_n}$$
$$T_{unit,i} = \frac{T(n)}{uc}\beta_i, (MB_{umin} \le i \le MB_{umax}) \tag{18}$$

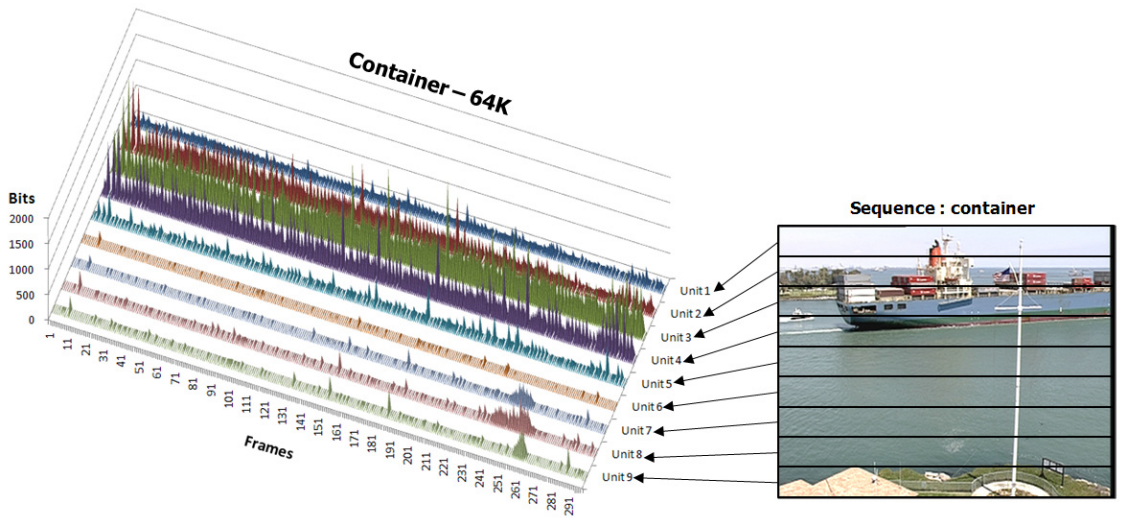

그림 4. Container(QCIF) 영상의 유닛별 발생 비트량 (300 frames)

www.dbpia.co.kr

여기서 *width*는 영상의 가로폭, MB_n는 영상내의 총 MB 개수, i는 Unit 번호, uc는 프레임내 Unit 개수, 그리고 β_i는 해당 Unit의 가중치를 나타낸다. 각 Unit의 가중치 β_i는 그림 5와 같이 예측된 목표 비트량 대비 이전 프레임들내의 각 유닛별 평균 비트량에 의해 정해진다. 그림 5에서 한 개의 유닛에 할당될 수 있는 비트량은 식 (19)와 같이 QP에 따라 평균 8~13%로 정해진다. 그리고 이전 프레임들에서 발생된 각각의 유닛별 비트량 통계정보를 이용하여 QP값이 결정되게 된다. 각 유닛의 QP값이 결정될 때는 각 유닛단위의 화질 열화를 막기 위해 ΔQP값이 ±2로 제한을 받지만, 그림 5와 같이 각 유닛간의 발생 비트량의 차이가 많을 경우에는 실험에 의해 ΔQP값의 제한을 최대 ±4로 설정하였다.

각 유닛에 대한 목표 비트량이 정해지면 다음과 같이 각 유닛에 대한 QP값을 결정하여 부호화를 수행한다.

$$QP_{unit,i} = unitQP(QP_{unit,n(1\sim 51)}, T_{unit,i}) \quad (19)$$

$unitQP()$는 식 (17)과 같이 QP 인덱스에 의한 예측 비트량 $QP_{unit,n(1\sim 51)}$ 구간에서 해당 목표 비트량 $T_{unit,i}$를 찾는 함수로써, 반복 루프를 통하여 구해진다.

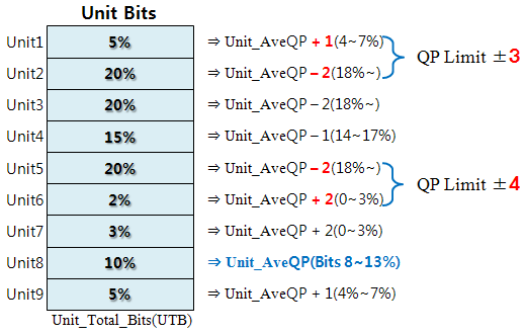

그림 5. 영상(QCIF) 특성에 따른 QP별 발생 비트량의 예

3.3 복잡도 갱신

이전 프레임을 부호화 한 후에 발생된 실제 비트량을 이용하여 현재 프레임내 각 유닛의 복잡도를 계산하는 단계로서, 가상 버퍼의 점유량 및 버퍼의 넘침과 고갈을 예방하기 위한 과정 그리고 이전에 부호화된 P 프레임들의 발생된 비트량과 현재 프레임의 발생 비트량을 이용하여 이후에 부호화시 사용될 P 프레임의 복잡도 가중치를 계산한다.

$$\begin{aligned} B_c &= B_c + (A(n-1) - \frac{u}{F_r} + 0.5) \\ U(n) &= U(n-1) - A(n-1) - \frac{u}{F_r} + 0.5) \\ L(n) &= L(n-1) - A(n-1) - \frac{u}{F_r} + 0.5) \end{aligned} \quad (20)$$

가상 버퍼의 점유량 B_c 및 버퍼의 넘침과 고갈을 예방하기 위한 버퍼의 상한 $U(n)$ 및 하한 $L(n)$값은 식 (20)을 통해서 구해진다.

그리고 다음에 부호화 할 P 프레임의 목표 비트량을 예측하기 위한 P 프레임의 복잡도 가중치 W_p는 식 (21)과 같이 구해진다.

$$\begin{aligned} W_P = &(\lambda \times A(n-1) + \\ &(1-\lambda) \times S_{bits}) \times QP_p + 0.5 \end{aligned} \quad (21)$$

본 논문에서는 P 프레임에 대한 복잡도 가중치를 예측하기 위한 부호화된 P 프레임들의 최대 개수를 20개로 제한하였으며, S_{bits}는 이전 P 프레임의 발생 비트량 $A(n-2) \sim A(n-20)$ 구간에서 QP_p와 동일한 QP값을 갖는 P 프레임들의 평균 비트량이다. 이는 프레임에 할당할 수 있는 비트량을 통하여 QP값을 결정하기 위해서는 각 QP값에 따라 발생되는 프레임 비트량을 알아야 하기 때문이다. 또한 부호화 하고자 하는 P 프레임은 인접한 프레임들과의 밀접한 상관관계를 가지고 있다. 따라서 목표 비트량과 실제 발생된 비트량과의 오차를 줄이기 위한 $A(n-1)$와 S_{bits}의 가중치 λ값을 0.67로 설정하였다. 이 λ값은 다양한 영상을 각각의 고정 QP값과 움직임 추정을 위한 참조 프레임을 최대 20으로 설정했을 때, 현재 프레임을 부호화하기 위한 이전 프레임의 참조율(T-1 참조율 67%, T-2~T-4참조율 12%, T-5~T-8 참조율 8%)을 나타낸 것으로 시뮬레이션하여 얻어낸 결과 값이다.

Ⅳ. 실험 및 결과

본 논문에서는 H.264/AVC 부호기를 기반으로 한 효율적인 비트율 제어 방법을 제시하였다. 제안한 알고리즘의 성능을 검증하기 위하여 H.264/AVC 참조 소프트웨어 12.1 부호기를 사용하였다[13].

실험 환경은 표 3과 같고 실험에 사용된 영상은 부호기에서의 성능평가 테스트로 광범위하게 사용되는 QCIF급 영상을 사용하였다. 영상의 GOP 구조

www.dbpia.co.kr

표 3. 실험환경

Profile	Baseline 3.0
MV Resolution	1/4 Pel
Hardamard Transform	ON
RDO	ON
Search Range	±16
Reference Frames	1
Symbol Mode	CAVLC
GOP size	30
Encoding Frame	300
Test Channel Bandwidth	64K ~ 384K

는 초기 I 프레임과 그 이후의 모든 프레임들은 P 프레임으로 구성하였고 B 프레임과 프레임 스킵은 고려하지 않았다. 그리고 각 알고리즘의 계산량 비교를 위하여 고밀도 측정 타이머를 사용하였으며, 계산량 측정 범위는 각 알고리즘의 데이터 및 버퍼의 초기화 부분을 제외한 비트율을 제어하는 부분에서만 측정하였다. 표 4에서 표 6은 채널 전송대역폭이 각각 64K, 96K, 128Kbps이고 프레임율이 30fps일 때 기존 비트율 제어 방법들과 제안한 비트율 제어 알고리즘에 대한 부호화된 비트량, PSNR 및 계산량 비교이다.

실험 결과표에서 알 수 있듯이 참고 논문(Ref.14)이 JM 12.1보다는 평균적으로 0.07Kbps의 적은 비트량을 절약하고도 PSNR 측면에서는 평균적으로 0.02~0.03dB 정도 높게 나와 더 좋은 성능을 보이고 있다. 그러나 Ref14의 방법은 PSNR 기반의 비트율을 제어하기 때문에 이에 따른 계산량 증가로 JM 12.1의 방법보다는 약 13%의 계산량이 더 필요한 것으로 나타났다. 그리고 JM 12.1과 Ref14의 방법은 제안한 방법보다 채널 전송환경에 부적합하게 평균적으로 0.2~0.3Kbps 비트량을 초과 발생하였다. 반면, 제안한 방법이 PSNR 측면에서 기존 방법들과 거의 같고 일부 영상에서는 평균적으로 0.02dB 정도 높게 나타났다. 채널 전송 환경에서의 발생 비트량은 JM 12.1 방법보다 0.2~0.4Kbps 정도 절약하였고, Ref.14 방법보다는 0.12Kbps 정도 절약하였다. 또한 제안한 방법이 전체적으로 채널 전송 환경에 적합하게 비트량을 잘 조절하고 있음을 알 수 있다. 계산량에 있어서는 JM 12.1 방법과 Ref.14 방법 모두 복잡한 MAD 계산과 이에 따른

표 4. 64 kbps 채널 전송 환경에서 각 영상들의 부호화 비트율, PSNR 및 계산량 비교

Sequence	JM 12.1			Ref.14			Proposed		
	Bit rate	PSNR	계산량 (㎲)	Bit rate	PSNR	계산량 (㎲)	Bit rate	PSNR	계산량 (㎲)
news	64.04	35.82	1,168.595	63.97	35.87	1,173.773	64.00	35.86	43.464
foreman	64.03	32.15	1,183.141	63.99	32.17	1,187.741	64.03	32.06	43.384
stefan	64.14	24.56	1,155.174	64.11	24.52	1,156.785	64.07	24.48	45.119
pamphlet	64.10	39.68	1,157.375	64.11	39.58	1,161.497	64.08	39.72	43.418
paris	64.34	29.05	1,184.514	64.12	29.05	1,187.151	64.11	29.06	43.035
sean	64.06	39.43	1,188.814	63.96	39.50	1,191.837	64.00	39.78	42.904

표 5. 96 kbps 채널 전송 환경에서 각 영상들의 부호화 비트율, PSNR 및 계산량 비교

Sequence	JM 12.1			Ref.14			Proposed		
	Bit rate	PSNR	계산량 (㎲)	Bit rate	PSNR	계산량 (㎲)	Bit rate	PSNR	계산량 (㎲)
news	96.05	38.59	1,185.668	96.14	38.64	1,191.996	96.08	38.64	43.175
foreman	96.14	34.26	1,210.393	95.95	34.29	1,215.146	95.92	32.20	44.598
stefan	96.16	26.24	1,175.496	96.09	26.14	1,176.321	96.09	26.22	48.535
pamphlet	96.07	42.24	1,175.940	96.11	42.35	1,182.151	96.07	42.09	43.553
paris	96.32	32.14	1,188.900	96.03	32.15	1,194.143	96.25	32.16	43.007
sean	96.12	41.91	1,199.055	96.14	41.93	1,198.868	96.01	42.15	42.906

www.dbpia.co.kr

표 6. 128 kbps 채널 전송 환경에서 각 영상들의 부호화 비트율, PSNR 및 계산량 비교

Sequence	JM 12.1			Ref.14			Proposed		
	Bit rate	PSNR	계산량 (㎲)	Bit rate	PSNR	계산량 (㎲)	Bit rate	PSNR	계산량 (㎲)
news	128.09	40.15	1,193.926	128.07	40.17	1,199.038	127.94	40.47	43.433
foreman	128.01	35.75	1,225.928	128.02	35.78	1,232.673	128.21	35.63	43.540
stefan	128.36	27.78	1,189.314	128.12	27.73	1,193.160	128.24	27.64	45.118
pamphlet	128.23	43.69	1,175.940	127.96	43.75	1,182.151	127.98	43.51	43.553
paris	128.21	33.65	1,204.631	128.01	33.69	1,206.749	128.18	33.66	42.952
sean	128.03	43.43	1,196.702	127.95	43.49	1,200.223	128.01	43.63	43.372

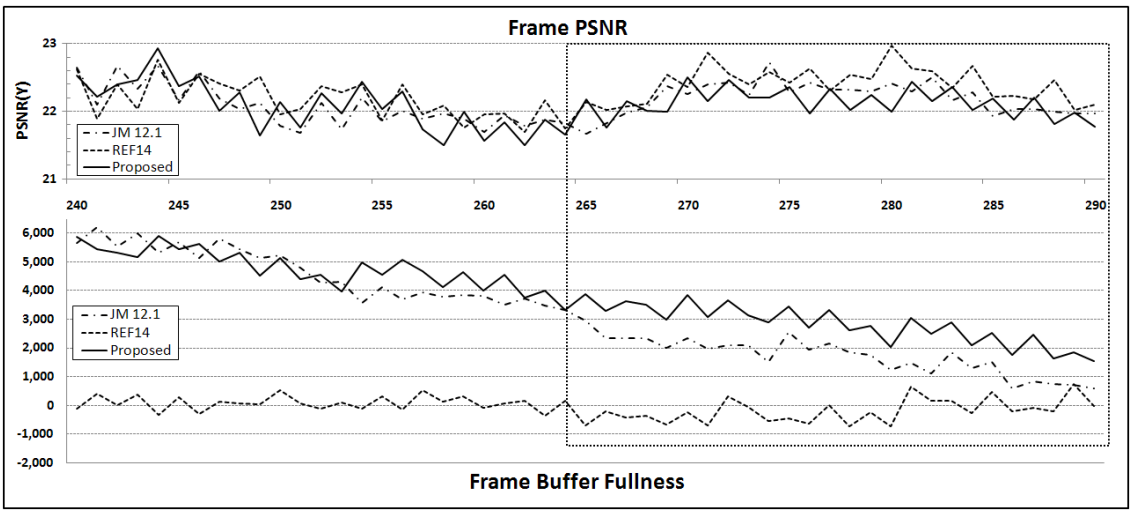

그림 6. STEFAN 영상의 제안 및 기존 방법들의 프레임별 PSNR 및 가상버퍼 점유율 비교(64Kbps)

비트량 조절을 위한 연산량이 제안한 방법보다 월등히 많음을 알 수 있다. 특히 현재 프레임에 대한 복잡도를 예측하는데 사용되는 MAD 계산량이 전체 비트율 제어 계산량의 평균적으로 약 12% ~ 14%정도 차지하고 있음을 실험결과 확인할 수 있었다.

실험 결과에서 알 수 있듯이 제안한 방법이 PSNR 측면에서 기존 JM 12.1 방법과 Ref.14 방법과는 평균적으로 거의 비슷하였지만, 계산량에서는 평균적으로 월등히 적은 0.1% 미만의 계산량으로도 효율적인 비트율을 제어할 수 있음을 확인하였다. 제안한 방법의 많은 계산량 절감은 기존 방법들과는 달리 복잡한 MAD 계산과 이에 따른 계산 과정을 따르지 않고, 기존 영상들의 통계정보를 이용하고 실시간적으로 발생하는 발생 비트량을 다시 통계정보로 활용함으로써 영상의 특성에 적응적으로 비트율 제어가 이루어졌기 때문이다.

그림 6은 Stefan 영상을 64Kbps 채널로 전송할 경우, 제안한 방법과 기존 방법들에 대하여 프레임별 부호화된 PSNR과 가상버퍼 점유율 상태를 비교한 것으로서, Ref.14 방법이 일부 프레임들에서 JM과 제안한 방법보다 많은 비트량을 발생시켜 PSNR 측면에서 다소 높게 나오고 있으나, 버퍼 점유율에서는 거의 모든 영상에서 평균적으로 언더플로우가 발생하고 있음을 보여주고 있다. 이는 영상 데이터를 전송함에 있어서 버퍼의 고갈 상태를 의미하는 것으로, 실시간으로 영상을 부호화하여 전송함에 있어서 적합하지 않음을 확인할 수 있다.

V. 결 론

본 논문에서는 H.264/AVC 부호기에서 부호화된

www.dbpia.co.kr

비트스트림을 고정된 채널로 전송할 경우, 기존의 비트율 제어 방법을 소개하고 문제점을 제시하였다. 기존의 비트율 제어 방법의 큰 문제점은 현재 프레임을 부호화하는데 있어서 이전 프레임의 차분부호 정보량을 이용하기 때문에 많은 계산량을 필요로 한다는 것이다. 이러한 문제점들을 해결하기 위하여 본 논문에서는 이전 프레임의 차분부호 정보량이 아닌, 다양한 영상의 통계정보와 이전에 발생된 비트량을 통계정보로 이용하여 프레임내 각 유닛별 영상 특성에 적응적으로 비트율을 제어하는 방법을 제안하였다.

본 논문에서 제안한 비트율 제어 알고리즘의 성능을 측정하기 위하여 기존의 비트율 제어 알고리즘과 비교 실험을 하였다. 실험 결과에서 알 수 있듯이 기존 비트율 제어 방법의 PSNR 성능과 거의 비슷하거나 평균적으로 0.02dB 개선이 있었다. 또한 다양한 채널 전송 환경에서도 기존 비트율 제어 방법들보다 목표 비트율에 적합하게 부호화되는 것으로 나타났다. 특히 계산량 절감에 있어서는 제안한 방법이 기존 방법들보다 99% 이상의 계산량이 감소하여 실시간 처리에 적합한 것으로 나타났다. 이는 영상과 관련된 응용 분야에 많은 활용이 있을 것으로 사료된다.

참 고 문 헌

〔1〕 *Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264|ISO/IEC 14496-10 AVC)* JVT-G050, May 2003.

〔2〕 T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthar, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. Circuit and Systems for Video Technology*, Vol.13, pp.560-576, July 2003.

〔3〕 Iain E.G. Richardson, H.264 and MPEG-4 Video Compression, Wiley, 2003.

〔4〕 J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: tools, performance, and complexity," *IEEE Circuits and Systems Magazine*, Vol.4, pp.7-28, 2004.

〔5〕 T. Wiegand, X. Zhang, and B. Girod, "Motion-compensating long-term memory prediction," *Proc. IEEE Int. Conf. Image Processing*, Vol.2, pp.53-56, Oct. 1997.

〔6〕 T. Wiegand, X. Zhang, and B. Girod, "Block-based hybrid video coding using motion-compensated long-term memory prediction," *Proc. Picture Coding Symp.*, pp. 153-158, Sep. 1997.

〔7〕 T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Trans. Circuit and Systems for Video Technology*, Vol.9, pp.70-84, Feb. 1999.

〔8〕 T. Wiegand and B. Girod, *Multi-Frame Motion-Compensated Prediction for Video Transmission*. Norwell, MA: Kluwer, 2001.

〔9〕 *Proposed Draft of Adaptive Rate Control, Joint Video Team (JVT) of ISOKEC MPEG & ITU-T VCEG (ISOWC JTCIISC29McGll and ITU-T SG16 4.6)*, May 2003.

〔10〕 T. Chiang, Y.-Q. Zhang, "A new rate -control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technoogy.*, Vol.7, No.1, pp.246-250, Feb. 1997.

〔11〕 Feng Pan, Z. Li, K. Lim and G. Feng, "A Study of MPEG-4 Rate Control Scheme and Its Improvements," *IEEE Trans. Circuits Syst. Video Technology*, Vol.13, No.5, pp.440-446, May 2003.

〔12〕 Lee H J, Chiang T H. Zhang Y Q. "Scalable rate control for MPEG-4 video," *IEEE Trans. Circuits System Video Technology*, Vol.10, No.6, pp.878-894, Oct. 2000.

〔13〕 JM12.1, H.264/AVC reference software http://iphome.hhi.de/suehring/tml/

〔14〕 M. Jiang and N.Ling, "On Enhancing H.264/AVC Video Rate Control by PSNR-Based Frame Complexity Estimation," *IEEE Trans. Consumer. Electronics.* Vol.51, No.1, pp.281-286, Feb. 2005.

〔15〕 M. Jiang X. Yi, and N.Ling, "Improved frame-layer rate control for H.264 using MAD ratio," *IEEE International Symposium on Circuits and Systems*, Vol. III, pp.813-816, Vancouver, Canada, May 2004.

〔16〕 T. Wiegand, H. Schwarz, A. Joch, F. Kossentini

www.dbpia.co.kr

and G. Sullivan, “Rate-Constrained Coder Control and Comparison of Video Coding Standards,” *IEEE Trans. Circuit and Systems for Video Technology*, Vol.13, No.7, July 2003.

〔17〕 G. J. Sullivan and T. Wiegand, “Rate-distortion optimization for video compression,” *IEEE Signal Processing Magazine*, pp.74-90, Nov. 1998.

〔18〕 A. Ortega and K. Ramchandran, “Rate-distortion methods for image and video compression,” *IEEE Signal Processing Magazine*, pp.23-50, Nov. 1998.

〔19〕 T. Chiang and Y.-Q. Zhang, “A new rate control scheme using quadratic rate distortion model,” *IEEE Trans. Circuit and Systems for Video Technology*, Vol.7, pp.246-250, Feb. 1997.

〔20〕 Z.G. Li, W. Gao, F. Pan, S.W. Ma, K.P. Lim, G.N, Feng, X. Lin, S. Rahardja, H.Q. Lu, and Y. Lu, “Adaptive rate control for H.264,” J.Vis. Commun. Image R., Vol.17, pp.376-406, April 2006.

김 명 진 (Myoung-Jin Kim) 정회원

2002년 2월 한국방송대학교 컴퓨터과학과(학사)
2005년 2월 한국방송대학교 정보과학과(석사)
2005년 3월~현재 숭실대학교 박사과정 재학 중
<관심분야> 영상통신 및 Rate Control, 동영상 부호화

주 원 희 (Won-Hee Joo) 준회원

2008년 2월 수원대학교 정보통신공학과 (학사)
2010년 2월~현재 숭실대학교 석사과정 재학 중
<관심분야> 영상신호처리 및 Rate Control, 동영상 부호화

홍 민 철 (Min-Cheol Hong) 종신회원

한국통신학회 논문지 제34권 10호 참조

www.dbpia.co.kr