

# H.264/AVC를 위한 통계 모델 기반 프레임 단위의 비트율 제어 기법

정희원 김 명 진\*, 종신회원 홍 민 철\*

## Adaptive Rate Control Based on Statistical Modeling in Frame-layer for H.264/AVC

Myoung-Jin Kim\* *Regular Member*, Min-Cheol Hong\* *Lifelong Member*

### 요 약

본 논문에서는 H.264/AVC를 위한 효율적이고 적응적인 프레임 단위의 비트율 제어 기법에 대해 제안한다. QP 값에 의한 프레임당 발생 비트량은 영상의 특성에 따라 다양하게 발생하고 있으며, 부호화하고자 하는 프레임은 인접한 프레임들과의 일정한 상관관계를 유지하는 특성이 존재한다. 제안 방식은 이러한 통계적 특성을 이용하여, 영상의 특성별 QP값에 따른 발생 비트량을 적용시키기 위하여 예측된 영상의 복잡도에 따라 QP값에 대한 발생 비트량의 변화를 구하여 부호화하고자 하는 프레임의 예측 비트량에 적용하고 이전 프레임들의 통계 정보인 QP와 발생비트량과의 가중치를 이용하여 현재 프레임의 복잡도를 예측한다. 실험 결과를 통해, 실험에 사용된 모든 CIF 영상에서 평균 PSNR 이득이 0.02~0.43dB에 이르는 성능 향상이 있었고, 계산량 또한 기존 방식 대비 99% 이상의 감소가 있음을 확인할 수 있었다.

**Key Words** : 비트율 제어, 통계적 특성, 영상 복잡도, MAD, PSNR, H.264/AVC

### ABSTRACT

In this paper, we propose an efficient and adaptive rate control in Frame-layer for 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 could not only achieves time saving of more than 99% over existing rate control algorithm, but also increase the average PSNR of reconstructed video for around 0.02~0.43 dB in all the sequences.

### I. 서 론

H.264/AVC는 ITU와 ISO/IEC가 함께 진행하여 향상된 비디오 부호화 표준을 제공하기 위해서 개발된 국제 압축 표준으로서, 다양한 예측 부호화 기

법과 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<sup>1-8)</sup>.

영상정보를 실시간으로 전송시, 영상의 특징과

\* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업 (NIPA-2010-C1090-1021-0010) 및 서울시 산학협력 사업(10570)으로 구축된 서울 미래형콘텐츠 컨버전스 클러스터 지원으로 수행되었음.

\* 숭실대학교 정보통신전자 공학부 (webzealer@ssu.ac.kr, mhong@ssu.ac.kr)

논문번호 : KICS2010-06-280, 접수일자 : 2010년 6월 28일, 최종논문접수일자 : 2010년 10월 11일

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 균형을 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<sup>9)~13)</sup>. 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 통계적 계산 간의 상호 의존적인 구조 때문에 H.264에서의 비트율 제어는 복잡한 형태가 되고 이에 따른 연산량이 증가된다. 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 값을 이용하여 프레임 복잡도를 예측하는 모델<sup>15)</sup>과 이 MAD 율과 함께 부호화된 P 프레임들의 평균 PSNR 오차율을 이용하여 현재 프레임의 복잡도를 예측하여 개선된 성능을 보여주고 있다<sup>14)</sup>. 그러나 이와 같은 MAD 기법들은 비트율 제어부분에서 약 14~15%의 계산량을 필요로 하기 때문에 동영상의 응용에 많은 제약을 갖게 된다<sup>21)</sup>.

본 논문에서는 이와 같은 문제점을 해결하기 위하여 QP와 부호화 이후에 발생하는 실제 비트량과의 통계적 모델을 기반으로 하여 기존의 방식보다 복잡도를 줄이는 방식을 사용함으로써, 적은 계산량으로도 프레임의 발생 비트량을 정확히 예측하여 기존의 방식보다 비트율 제어 알고리즘의 전체적인 계산량을 줄이는 기법을 제안한다. H.264/AVC는 VBR(가변 비트율 부호화 방법)과 CBR(고정 비트율 부호화 방법)을 동시에 지원하지 않지만, 이 논문에서는 프레임 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, 프레임의 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

### 2.1 비트율-왜곡 최적화

H.264/AVC의 비트율-왜곡 최적화(RDO:Rate Distortion Optimization)의 최종 목적은 부호화측에서는 최소의 비트율을 부호화하여 손실없이 최종단 말기에 전송하는데 목적이 있고, 복호화측에서는 최대의 화질을 제공받고자 왜곡 값이 최소인 매크로 블록을 최적의 매크로블록 모드로 결정한다. H.264/AVC의 테스트 모델은 라그랑지 최적화 모드와 움직임 추정 방법<sup>16)</sup>에 대한 방법을 채택하고 있다. 이 방법에서 라그랑지 계수( $\lambda$ )는 다음 식 (1)과 같이 각각의 매크로블록에 대하여 양자화 계수(QP)를 사용하여 라그랑지 값을 결정한다.

$$\lambda = 0.85 \times 2^{\frac{QP-12}{3}} \quad (1)$$

라그랑지 최적화 방법<sup>17)~18)</sup>에서 라그랑지 계수가 작을수록 높은 비트율과 좋은 화질을 보인 반면에 라그랑지 계수가 높을수록 낮은 비트율과 나쁜 화질을 초래한다는 결과와 같이 라그랑지 계수는 화질과 비트율에 영향을 끼치는 것을 알 수 있다.

### 2.2 프레임에 대한 초기화

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 식 (2)와 같이 1/8로 설정하고, 버퍼의 넘침(Overflow)과 고갈(Underflow)을 방지하기 위하여 버퍼의 상한( $U(1)$ ) 및 하한( $L(1)$ )의 초기값을 식

(3)과 같이 설정한다.

$$B_c(1) = \frac{B_s}{8} \quad (2)$$

$$L(1) = \frac{u}{F_r} \quad (3)$$

$$U(1) = \alpha \times B_s, (\alpha = 0.8)$$

식 (3)에서  $u$ 은 채널 대역폭으로 압축된 영상 시퀀스를 일정하게 전송하는 비트율이고,  $F_r$ 은 초당 부호화된 프레임 율을 나타낸다. 그리고 현재 버퍼 상태에 적합하도록 다음 식 (4)과 같이 초기 목표 버퍼레벨(TBL: Target Buffer Level)을 설정한다.

$$TBL(2) = B_c(2) \quad (4)$$

### 2.3 프레임에 대한 목표 비트량 계산

$n$ 번째 프레임을 부호화할 때마다 가상버퍼 점유량( $B_c(n)$ ) 및 목표 버퍼레벨( $TBL(n)$ )은 각각 식 (5)와 식 (6)과 같이 유동적으로 변하는 유동 트래픽 모델(Fluid traffic model) 식을 따르고 있다.

$$B_c(n) = B_c(n-1) + \left( A(n-1) - \frac{u}{F_r} \right) \quad (5)$$

$$TBL(n) = TBL(n-1) - \frac{TBL(2) - B_s/8}{N_p - 1} \quad (6)$$

식 (5)에서  $A(n-1)$ 는  $(n-1)$ 번째 프레임에 대한 부호화된 비트량이고, 식 (6)에서  $N_p$ 는 하나의 GOP 내에 있는 P 프레임의 개수를 의미한다.  $n$ 번째 프레임을 부호화하기 이전에, 프레임 당 할당할 수 있는 목표 비트량을 예측하기 위한 버퍼량 예측은 가상버퍼 점유량 및 목표 버퍼레벨을 이용하여 식 (7)과 같이 예측할 수 있다.

$$T_{buf}(n) = \frac{u}{F_r} + \gamma(TBL(n) - B_c(n)), (\gamma = 0.7) \quad (7)$$

이렇게 프레임 당 버퍼량이 예측되면 버퍼 상태를 고려하여 식 (8)과 같이 버퍼의 상한과 하한 범위까지 고려하여 최종적으로 버퍼량( $T_{buf}(n)$ ) 값을 결정한다.

$$T_{buf}(n) = \min(U(n), \max(L(n), T_{buf}(n))) \quad (8)$$

즉,  $n$ 번째 프레임에 대한 목표 비트량  $T(n)$ 은 식 (8)에서 예측한 버퍼량과 식 (9)에서 예측한 부호 비트량을 고려하여 식 (10)과 같이 최종적으로 결정된다.

$$T_r = \frac{R}{N-n} \quad (9)$$

$$T(n) = \beta \times T_r + (1-\beta) \times T_{buf}, (\beta = 0.5) \quad (10)$$

식 (9)에서  $T_r$ 은 프레임당 부호화할 수 있는 비트량으로 현재 남아 있는 비트량( $R$ )과 앞으로 부호화할 남아있는 프레임의 개수( $N-n$ )를 이용하여 계산한다.

### 2.4 MAD 예측 및 양자화 계수 예측

현재 프레임에 대한 복잡도를 예측하는데 사용되는  $MAD$ 는 다음과 같이 계산된다.

$$MAD_c = C_1 \times MAD_p + C_2 \quad (11)$$

$$C_1 = \frac{1}{20} \sum_{i=1}^{20} \frac{MAD_i}{MAD_{i-1}}$$

이때,  $MAD_c$ 는 현재 프레임의 복잡도를,  $MAD_p$ 는 이전 프레임의  $MAD$ 를 나타내며  $C_2$ 는 예측된  $MAD$ 값들의 예측 오차에 대한 보상 값이다.  $C_1$ 과  $C_2$ 에 대한 초기값은 각각 1과 0이다. 현재 H.264/AVC의 비트율을 제어하기 위한 프레임 단위에 대한 양자화 계수는 다음과 같이 2차 방정식 비트율-왜곡 모델<sup>[9]</sup>을 사용한다.

$$T(n) = \frac{x_1 \times MAD_n}{QP} + \frac{x_2 \times MAD_n}{QP^2} \quad (12)$$

식 (12)에서  $T(n)$ 은  $n$ 번째 프레임에서의 목표 비트량,  $MAD_n$ 는  $n$ 번째 프레임에서의  $MAD$ ,  $QP$ 는 양자화 계수, 그리고  $x_1$ 와  $x_2$ 는 각 프레임에서 얻어지는 2차원 비트율-왜곡값을 나타내는 변수이다.

### 2.5 부호화 이후의 버퍼 상태 갱신

예측한 목표 비트량을 이용하여 부호화 수행 후, 현재 버퍼의 상한 및 하한 값을 식 (13)과 같이 갱신하여, 다음 프레임에서 버퍼가 넘치거나 고갈되지

않도록 한다.  $\omega (= 0.8)$ 은 버퍼의 넘침을 방지하기 위한 상수이다.

$$\begin{aligned} L(n) &= L(n-1) + R - A(n-1) \\ U(n) &= U(n-1) + ((R - A(n-1)) \times \omega) \end{aligned} \quad (13)$$

### III. 제안하는 비트율 제어 알고리즘

H.264/AVC에서 사용되고 있는 Quadratic R-D 모델<sup>[10]</sup>은 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 비트율을 예측하는 과정에서 불필요한 계산량을 줄이고 영상의 왜곡을 최소화하는 것을 목적으로 한다.

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

제안된 방식은 3개의 주요 처리 단계로 이루어져 있다. 1) 프레임에 대한 목표 비트량 계산, 2) 목표 비트량에 대한 QP값 결정, 3) 프레임에 대한 복잡도 갱신.

#### 3.1 프레임에 대한 목표 비트량 계산

현재 부호화 하고자 하는 프레임의 목표 비트량을 예측하기 위해서는 부호화 하고자 하는 프레임에 대한 복잡도 가중치, 가상버퍼 점유율 그리고 목표 버퍼레벨을 고려해야 한다.

본 논문에서의 가상버퍼 점유율( $B_c(n)$ ) 및 목표 버퍼레벨( $TBL(n)$ )은 각각 식 (14)와 식 (15)를 이용하여 구한다. 이 때  $P_c$ 는 GOP 구간에서의 부호화된 P 프레임 개수를 나타낸다. 식 (15)와 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 버퍼 레벨에서 값을 줄인다.

$$B_c(n) = A(n-1) - \frac{u}{F_r} + 0.5 \quad (14)$$

$$TBL(n) = \begin{cases} TBL(n-1) - \frac{B_c}{N_p-1}, & P_c > 1 \\ B_c - \frac{B_c}{N_p-1}, & P_c = 0 \end{cases} \quad (15)$$

현재 프레임에 부호화함에 있어서, 프레임에 할당할 수 있는 목표 비트량을 예측하기 위해서는 다음 식 (16)와 같이 계산된 버퍼 점유량 및 목표 버퍼레벨을 이용하여 버퍼량을 예측한다.

$$T_{buf} = \frac{u}{F_r} + \gamma \times (TBL(n) - B_c(n)), (\gamma = 0.8) \quad (16)$$

그리고 버퍼량이 예측되면 버퍼 상태를 고려하여 식 (8)과 같이 버퍼의 상한과 하한 범위까지 함께 고려하여 최종적으로 버퍼량( $T_{buf}(n)$ ) 값을 결정한다. 기존 방식에서는  $\gamma$ 의 초기값으로 0.7을 사용하지만, 본 논문에서는 버퍼 점유량과 버퍼 레벨의 가중치를 높여 초기값으로 0.8을 사용하였다. 이는 기존 JM 방식의 다양한 조건으로 실험했을 경우에 발생하는 일부 조건에서의 버퍼 오버플로우를 방지하기 위한 것으로서, 실험에 의해 결정된 값이다. n 번째 프레임에 대한 최종 목표 비트량( $T(n)$ )은 식 (16)에서 예측한 목표 버퍼량과 식 (17)에서 예측한 부호 비트량을 고려하여 식 (18)과 같이 최종적으로 결정된다.

$$T_r = \left( \alpha \times \frac{B_c(n)}{N-n} \right) + ((1-\alpha) \times T_{buf} + 0.5) \quad (17)$$

$$T(n) = (\beta \times W_p) + ((1-\beta) \times T_r + 0.5) \quad (18)$$

여기서  $T_r$ 은 기존 방식의 식(9)와 달리 현재 프레임에 대한 버퍼 점유량과 예측된 목표 버퍼량의 가중치  $\alpha$ 를 이용하여 계산되며, 실험에 의해 0.5로 설정하였다. 이 또한 식 (16)과 같이 버퍼 오버플로우를 방지하기 위해 남아 있는 비트량 대신 현재 버퍼 점유량을 이용한 것이다. 그리고 최종 목표 비트량( $T(n)$ )의 경우, 기존 방식에서는 남아 있는 비트량을 기준으로 부호화할 수 있는 비트량과 목표 버퍼량의 가중치를 적용하지만, 계산된 비트량에는 이전 프레임들에 대한 비트량 정보가 고려되지 않았기 때문에, 본 논문에서는 현재 P 프레임의 복잡

도 가중치( $W_p$ )를 고려하여 최종 목표 비트량을 계산하도록 하였다.  $W_p$  값은 식 (26)을 통해 계산되며, 이전에 부호화된 프레임들의 발생 비트량 가중치와  $QP$ 값으로 계산된다.  $\beta$ 는 P 프레임의 복잡도 가중치와 프레임당 부호화 할 수 있는 비트량과의 가중치로써, 실험에 의해 0.5로 설정하였다.

### 3.2 목표 비트량에 대한 $QP$ 값 결정

예측된 목표 비트량을 이용하여 부호화 하고자 하는 프레임의  $QP$ 값을 결정함에 있어서, 본 논문에서는 기존 H.264/AVC 방식의 식 (12)와 같은  $QP$ 변환 알고리즘을 사용하지 않고, 표 1과 같이 다양한 영상의 실험에 의한 통계 정보를 이용하여 결정하도록 하였다.

표 1은 각 프레임의 비트량을 예측하기 위하여 산출된 실험 데이터로서 각 영상의 P 프레임들의 평균 비트량을 나타낸 것이다. 그리고 다양한 유형의 영상 통계 정보를 사용하기 위하여 사용된 CIF 영상들은 빠르고 세밀한 영상의 “Mobile”과 “Stefan”, 그리고 느리고 평탄한 영상의 “News”와 “Container”가 사용되었으며, 보통의 빠르기와 평탄함 및 세밀함이 골고루 있는 영상 “Foreman”이 사용되었다.

그림 1은 표 1의  $QP$ 값에 의해 발생된 평균 비트량의 특성을 나타낸 것으로, P 프레임들의 평균 비트량이  $QP$ 값과 밀접한 관계를 가지고 있음을 확인할 수 있다. 이는  $QP$ 값에 따라 할당되는 비트량의 크기가 일정한 형태로 변하는 것을 의미한다. 이렇게  $QP$ 값과 발생 비트량과의 관계는 식 (19)와

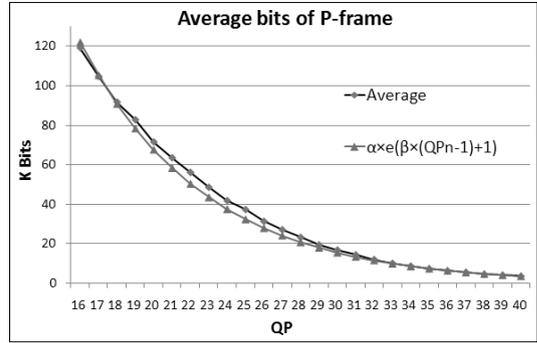


그림 1. 영상 특성에 따른  $QP$ 별 발생 비트량

같이 유도될 수 있으며 발생된 P 프레임들의 평균 비트량과 거의 유사함을 확인 할 수 있다.

$$QP_{bits,n} = \alpha \times e^{(\beta \times (QP_{n-1} + 1))}, (1 \leq n \leq 51) \quad (19)$$

여기서  $n$ 은  $QP$  값,  $QP_{bits,n}$ 는  $QP$  값  $n$ 에 대한 예측된 발생 비트량을 의미한다. 그리고  $\alpha$ 와  $\beta$ 는 그림 1과 표 1의 다양한 영상의 통계 특성에 의해 결정된 상수 값이다. 식 (19)에서 적용된  $QP_{n-1}$ 는 예측하고자 하는  $QP$ 값에서 -1을 한  $QP$  값을 의미한다. 즉, 예측하고자 하는  $QP$ 값이 24일 경우에 그림 1에 대한 통계적 특성에서 볼 수 있듯이  $QP$ 값 23의 영향을 받고 있음을 알 수 있다. 또한  $QP$ 값 25는 예측한  $QP$ 값 24에서 영향을 받는다. 이와 같이  $QP$ 값에 따른 발생 비트량의 변화가 영상의 특성에 따라 가우시안 분포의 형태를 가질 수 있기 때문에 현재  $QP$ 값에 대한 예측 가능한 발생 비트량은  $(QP-1)$ 값에 대한 발생 비트량과  $(QP+1)$ 값에 대한 발생 비트량의 영향을 받을 수 밖에 없다. 따라서 식 (19)의  $QP$  인덱스에 대한 예측 가능한 발생 비트량  $QP_{bits,n}$ 는 식 (20)과 같이 재계산 되어야 한다.

$$QP_{bits,n} = QP_{bits,(n+1)} + \frac{|QP_{bits,n} - QP_{bits,(n+1)}|}{2}, (20) \quad (0 \leq n < 51)$$

$QP$ 에 의해 예측된 발생 비트량  $QP_{bits,(0-51)}$ 은 초기화 단계에서 한 번 계산되고, 1차원 테이블에 값을 저장하여 사용한다. 이 테이블 정보는 매 프레임이 부호화 된 이후에 발생하는  $QP$  값에 따른 발생 비트량과 이전 프레임의 통계 정보를 고려하여 비트량의 변화가 업데이트 된다. 이렇게 예측된 발생

표 1. 각  $QP$ 에 따른 P 프레임의 평균 비트량

QP	container	foreman	mobile	news	stefan
...					
17	51,395	82,768	189,550	28,541	171,396
18	42,484	68,656	169,702	24,412	153,408
19	36,361	59,772	155,991	21,704	140,248
20	28,845	49,106	137,400	18,580	124,006
21	24,139	42,086	123,580	16,430	111,703
22	20,027	36,069	110,859	14,542	99,234
23	16,023	30,299	96,701	12,610	87,396
24	12,896	25,432	84,335	10,993	75,787
25	10,634	22,152	75,887	9,831	68,227
26	8,371	18,285	64,033	8,398	57,865
...					

비트량은 목표 비트량을 이용하여 다음과 같이 현재 프레임의  $QP_C$  값을 결정한다.

$$QP_C = \text{frame}QP(QP_{bits,n(1\sim 51)}, T), \quad (21)$$

$$(1 \leq n \leq 51)$$

여기서  $\text{frame}QP()$ 는  $QP$  인덱스에 의한 예측 비트량  $QP_{bits,n(1\sim 51)}$  구간에서 해당 목표 비트량  $T$ 를 찾는 함수로써, 반복 루프를 통하여 구해진다. 그리고 P 프레임의  $QP_C$ 값이 결정될 때는 프레임들 간의 화질 열화를 줄이기 위하여 다음과 같이  $\Delta QP$  값으로 제한하였다.

$$QP_C = \min(QP_P + \Delta QP, \max(QP_P - \Delta QP, QP_C)) \quad (22)$$

위 식에서  $QP_P$ 는 이전 프레임에서의  $QP$ 값을 나타내며 프레임간의  $QP$ 값을 제한하는  $\Delta QP$ 값은  $\pm 2$ 로 설정하였다.

### 3.3 프레임에 대한 복잡도 갱신

이전 프레임의 부호화를 수행한 이후에 발생된 실제 비트량을 이용하여 현재 프레임의 복잡도를 계산함에 있어서, 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 계산해야 한다.

본 논문에서의 가상 버퍼 점유량  $B_c$  및 버퍼의 넘침과 고갈을 예방하기 위한 버퍼의 상한  $U(n)$  및 하한  $L(n)$  값은 기존 방식의 식(13)과 달리 전송 채널 대역폭에 따른 프레임에 할당될 수 있는 비트량( $u/F_r$ )과 부호화 이후에 발생된 실제 비트량과의 오차를 고려하여 식 (23)~(25)와 같이 계산하였다.

$$B_c = B_c + (A(n-1) - \frac{u}{F_r} + 0.5) \quad (23)$$

$$U(n) = U(n-1) - (A(n-1) - \frac{u}{F_r} + 0.5) \quad (24)$$

$$L(n) = L(n-1) - (A(n-1) - \frac{u}{F_r} + 0.5) \quad (25)$$

다음에 부호화 할 P 프레임의 목표 비트량을 예측하기 위한 P 프레임의 복잡도 가중치  $W_P$ 는 부호

화된 P 프레임들의  $QP$ 값에 따른 발생 비트량의 통계 정보를 이용하여 식 (26)과 같이 계산된다.

$$W_P = (\lambda \times A(n-1) + (1-\lambda) \times S_{bits}) \times QP_P + 0.5 \quad (26)$$

여기서  $S_{bits}$ 는 이전 P 프레임의 발생 비트량  $A(n-1) \sim A(n-3)$  구간에서  $QP_P$ 와 동일한  $QP$  값을 갖는 P 프레임들의 평균 비트량이다. 이는 프레임에 할당할 수 있는 비트량을 통하여  $QP$ 값을 결정하기 위해서는 각  $QP$ 값에 따라 발생하는 프레임 비트량을 알아야 하기 때문이다. 또한 부호화하고자 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  $\lambda$ 값을 0.67로 설정하였다. 이  $\lambda$ 값은 각각의 고정  $QP$ 값과 움직임 추정을 위한 참조 프레임을 20으로 설정하고 시뮬레이션하여 얻어낸 결과 값이다<sup>22)</sup>. 그러나 현재 프레임의 복잡도를 예측하기 위해 참고되는 이전 영상들의 통계정보가 많을수록 정확하게 예측할 수도 있지만, 이는 움직임의 변화가 많고 장면 전환 같은 영상들이 많이 포함되어 있는 영상들에서는 현재 프레임의 복잡도 계산에 있어서 정확도를 감소시키는 원인이 되기도 한다.

따라서 본 논문에서는 현재 프레임을 부호화하기 위한 이전 프레임의 참조율(T-1 참조율 67%, T-2~T-4 참조율 12%, T-5~T-8 참조율 8%)<sup>22)</sup>에 따라서 복잡도 가중치를 예측하기 위한 P 프레임들의 참고 개수를 3개로 제한하였다. 이로 인하여 영상의 다양한 특성에 따른 변화를 적응적으로 현재 프레임의 복잡도 계산에 빠르게 적용할 수 있어, 실험 결과 모든 영상에서 기존 방식보다 좋은 결과를 얻을 수 있었다.

## 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<sup>13)</sup>. 실험 환경은 표 2와 같고 실험에 사용된 영상은 부호기에서의 성능평가 테스트로 광범위하게 사용되는 CIF(352×288)급 영상들을 사용하였다. 그리고 본 논문에서는 기존 방법과 달리 영상의 복잡도를 예

표 2. 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	128K, 192K, 256K, 320K, 384K, 448K, 512K, 576K, 672K, 768K
Frame Rate	@10, @15, @20, @30

측함에 있어서 복잡도 예측 시간의 손실없이 실시간 전송에 적합한 알고리즘을 나타내기 위하여 P 프레임의 통계적 특성만을 고려하였기 때문에 영상의 GOP 구조는 초기 I 프레임과 그 이후의 모든 프레임들은 B 프레임 없이 P 프레임만으로 구성하였다. 또한, 다양한 프레임율로 실험하였기 때문에 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. 기존 비트율 제어 알고리즘인 JM 12.1과 참고 논문<sup>14)</sup>, 그리고 본 논문에서 제안한 비트율 제어 알고리즘은 영상 화질의 객관적 평가를 위하여 PSNR(Peak Signal to Noise Ratio)을 사용하였다.

표 3은 채널 전송대역폭이 128Kbps ~ 768Kbps 이고 프레임율을 15fps 및 30fps으로 하였을 때, 기존 비트율 제어 방법들과 제안한 비트율 제어 알고리즘에 대한 평균 PSNR(Y) 및 계산량 비교를 나타낸 것으로, 전체적으로 0.02~0.43dB의 PSNR 성능 개선을 가져왔고, 계산량에 있어서도 기존 방식 대비 99%이상의 계산량 감소의 성능 개선을 얻을 수 있었다. 표 4의 부호화 비트율에서도, 프레임 율 @15의 Foreman와 @30의 Children를 제외한 모든 영상에서 기존 방식 대비 비트율이 안정적으로 목표 비트율에 맞게 발생하는 것을 알 수 있다. 이 결과에서 알 수 있듯이 더 적은 비트량을 발생하면서도 이에 따른 PSNR이 더 높은 것을 확인할 수 있다. 또한 영상의 복잡도가 높고 전체적인 변화가 많은 Mobile의 실험 결과 표 5와 그림 4에서, 전반적으로 채널 대역폭과 프레임 율에 관계 없이 기존 방식 대비 목표 비트율에 맞는 비트량으로 PSNR의 성능 개선이 있음을 확인 할 수 있다. 그리고 본 논문에서는 다양한 프레임율을(@10 ~ @30) 적용하여 실험 하였지만, @10와 @15 그리고 @20와

표 3. 각 영상별 전체평균 PSNR(Y) 및 계산량 비교 (128Kbps ~ 768Kbps)

Sequence (CIF)	PSNR(Y)						Encoding Time(μs)					
	JM @15	Ref.@15	Prop.@15	JM @30	Ref.@30	Prop.@30	JM@15	Ref@15	Prop.@15	JM@30	Ref.@30	Prop.@30
Container	40.69	40.69	40.79	37.93	37.92	38.02	162,784	161,602	111	156,555	158,370	18
News	44.25	44.32	44.36	41.06	41.08	41.16	164,039	159,269	109	156,338	156,083	19
Foreman	38.21	38.21	38.22	35.44	35.45	35.47	164,456	162,688	108	159,629	159,434	17
Mobile	29.65	29.67	29.78	26.73	26.74	26.84	164,232	159,722	108	158,381	156,528	17
Stefan	32.96	32.98	33.00	29.61	29.62	29.66	165,151	159,287	114	158,370	156,101	17
Akiyo	47.53	47.54	47.53	44.92	44.95	45.05	158,699	154,746	110	156,425	151,651	17
Costguard	33.32	33.33	33.36	30.68	30.70	30.73	163,179	158,797	106	159,084	155,621	17
H & M	40.12	40.12	40.13	38.52	38.53	38.57	161,476	159,166	113	158,802	155,982	18
M & D	44.92	44.93	44.92	42.79	42.79	42.80	161,239	157,873	108	158,784	154,716	17
Paris	38.07	38.09	38.21	34.04	34.08	34.23	158,836	156,041	106	157,483	152,920	17
Flower	30.12	30.13	30.18	26.83	26.84	26.90	160,111	158,837	106	158,818	155,660	17
Bus	32.43	32.43	32.47	29.13	29.12	29.16	160,654	159,273	108	161,687	156,088	17
Children	38.87	38.83	38.87	34.12	34.11	34.17	157,901	154,672	107	157,778	151,579	17
Tempete	32.29	32.30	32.36	29.49	29.50	29.55	161,707	159,512	107	160,527	156,321	17
Waterfall	37.93	37.94	38.01	35.17	35.17	35.27	162,510	161,076	106	161,906	157,854	17

표 4. 128 kbps 채널 전송 환경에서 각 영상들의 부호화 비트율 및 PSNR 비교

Sequence (CIF)	Bit rate(Kbps)						PSNR(Y)					
	JM @15	Ref. @15	Prop. @15	JM @30	Ref. @30	Prop. @30	JM @15	Ref@15	Prop. @15	JM@30	Ref. @30	Prop. @30
Container	127.94	127.99	127.78	127.85	128.06	127.89	36.53	36.46	36.60	34.25	34.21	34.29
News	128.19	128.04	128.27	128.24	127.94	128.13	39.21	39.26	39.30	35.55	35.43	35.67
Foreman	128.05	128.04	128.11	128.23	128.27	127.87	34.00	33.99	34.01	31.08	31.12	31.22
Mobile	127.98	128.03	127.77	127.99	127.93	127.94	25.20	25.23	25.32	22.35	22.33	22.50
Stefan	128.49	128.58	128.06	128.53	128.65	127.96	27.84	27.84	27.86	24.14	24.13	24.18
Akiyo	127.77	127.76	127.88	127.83	127.87	127.85	43.73	43.69	43.75	41.30	41.38	41.44
Costguard	128.10	128.05	127.66	128.05	128.09	127.78	29.42	29.42	29.43	27.43	27.46	27.50
H & M	128.10	128.09	127.96	128.12	128.08	127.98	37.71	37.75	37.76	35.72	35.70	35.87
M & D	128.23	128.10	127.90	128.32	128.41	127.86	41.56	41.58	41.59	38.85	38.79	38.84
Paris	128.18	128.19	127.82	128.45	128.53	127.96	31.90	31.62	31.93	29.20	29.20	29.32
Flower	128.03	127.79	127.98	128.25	127.95	127.80	25.19	25.18	25.25	22.48	22.43	22.53
Bus	128.27	128.08	128.12	130.56	129.02	128.22	27.43	27.42	27.47	24.59	24.52	24.50
Children	128.00	127.96	127.83	127.85	128.05	128.85	31.56	31.57	31.62	28.04	27.99	28.19
Tempete	127.96	128.05	127.92	128.27	127.97	127.90	28.07	28.08	28.13	25.59	25.57	25.64
Waterfall	128.01	128.03	127.73	128.24	127.86	127.64	33.75	33.70	33.83	30.94	30.93	31.04

표 5. Mobile(CIF) 영상의 부호화 비트율 및 PSNR 비교

Target (Kbps)	Bit rate(Kbps)						PSNR(Y)					
	JM @15	Ref. @15	Prop. @15	JM @30	Ref. @30	Prop. @30	JM @15	Ref@15	Prop. @15	JM@30	Ref. @30	Prop. @30
128	127.98	128.03	127.77	127.99	127.93	127.94	25.20	25.23	25.32	22.35	22.33	22.50
192	193.17	192.17	191.71	192.16	191.99	191.71	26.85	26.85	26.94	24.13	24.11	24.23
256	256.27	256.12	256.46	255.97	256.06	255.53	28.02	28.05	28.09	25.20	25.23	25.32
320	320.14	320.19	320.64	320.55	320.66	319.49	28.82	28.82	28.96	26.15	26.14	26.21
384	384.07	384.13	384.94	386.34	384.34	383.41	29.61	29.63	29.76	26.85	26.85	26.94
448	448.63	448.08	448.60	448.44	448.57	448.96	30.31	30.32	30.47	27.47	27.50	27.59
512	512.31	512.40	511.57	512.54	512.23	512.92	30.94	30.95	31.07	28.02	28.05	28.09
576	575.98	576.16	577.06	576.32	576.43	575.78	31.50	31.52	31.64	28.48	28.53	28.58
672	673.03	672.96	670.59	672.34	672.43	671.72	32.28	32.32	32.41	29.04	29.05	29.17

@30에서 거의 유사한 실험 결과를 보여 @15와 @30만을 비교 설명하였다.

그림 2와 그림 3은 Mobile 영상 128Kbps 에서의 프레임 별 PSNR의 변화 및 가상버퍼(B<sub>v</sub>)의 점유량을 나타낸 것이다. 기존 방법들 대비 제안한 방식의 프레임별 PSNR 값 및 가상 버퍼 점유량의 변화가 크지 않고 전체 프레임 구간에서 안정적으로 유지함으로써, 영상의 균일한 화질을 보장하여 피로감을 덜 느끼게 하고, 버퍼의 고갈 및 넘침을 방지한다. 특히 참고 논문<sup>[14]</sup>(Ref.14) 방식은 235 프레임 이후에 버퍼 오버플로우가 계속해서 발생하였다. 실험

결과표에서 Ref.14이 JM 12.1 보다는 PSNR 값이 평균적으로 0.02 ~ 0.03dB 정도 높게 나와 더 좋은 성능을 보이고 있다.

그러나 JM 12.1과 같이 제안한 방법보다 채널 전송환경에 부적합하게 평균적으로 약 0.2~0.3Kbps 비트량을 초과 발생하였다. 반면, 제안한 방법은 더 안정적으로 목표 비트량에 맞게 비트량을 발생시키면서도 더 높은 PSNR 성능 개선을 가져왔다.

계산량에 있어서도 JM 12.1 방법과 Ref.14 방법 모두 복잡한 MAD 계산과 이에 따른 비트량 조절을 위한 연산량이 제안한 방법보다 월등히 많음을

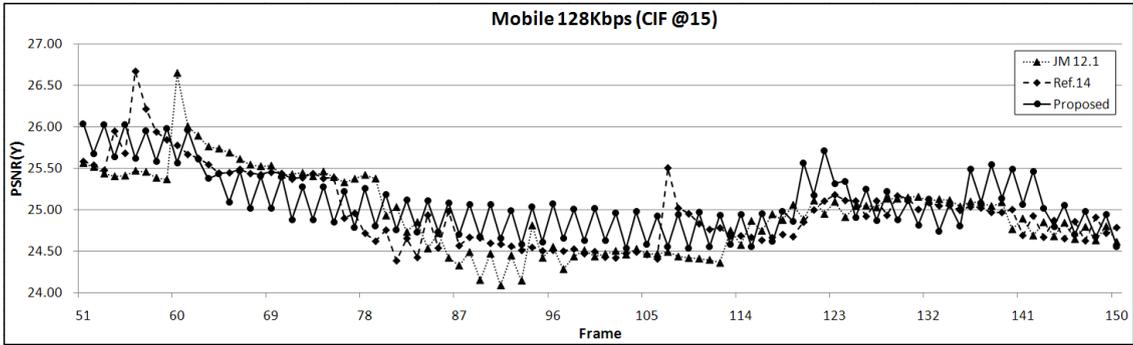


그림 2. Mobile(CIF) 영상의 제안 방법 및 기존 방법들의 프레임별 PSNR 비교(128Kbps, @15)

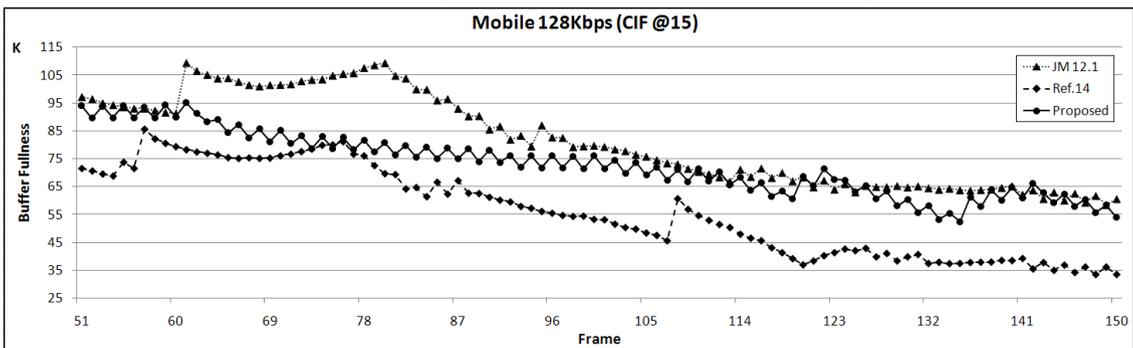


그림 3. Mobile(CIF) 영상의 제안 방법 및 기존 방법들의 프레임별 가상버퍼 점유율 비교(128Kbps, @15)

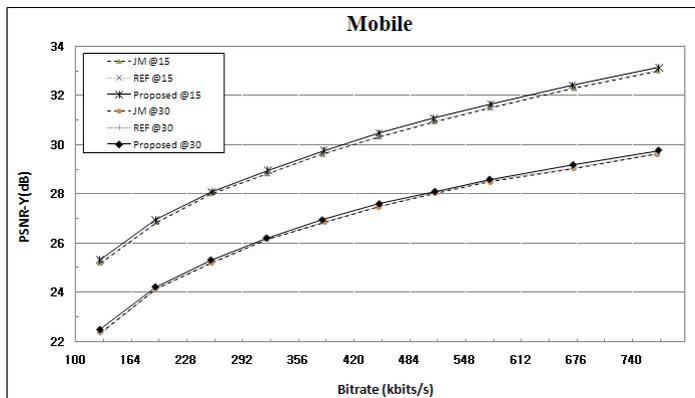


그림 4. Mobile(CIF) 영상의 PSNR 비교(128Kbps ~ 768Kbps, 300 Frames, @15, @30)

알 수 있다. 이는 현재 프레임에 대한 복잡도를 예측하는데 사용되는 MAD 계산량이 비트율 제어 전체 계산량의 약 12% ~ 14% 정도 차지하고 있음을 확인할 수 있다<sup>21)</sup>. 제안한 방법의 많은 계산량 절감은 기존 방법들과는 달리 복잡한 MAD 계산과 이에 따른 계산 과정을 따르지 않고, 기존 영상들의 통계정보를 이용하고 실시간적으로 발생하는 비트량

을 다시 통계정보로 활용함으로써 별도의 계산량이 필요하지 않았기 때문이다.

또한 기존 방식 대비 PSNR의 성능 개선은 영상의 특성에 따른 변화를 최근 인접한 3개의 프레임 통계 정보를 활용하여 현재 프레임의 복잡도를 좀 더 정확하게 예측함으로써, 영상의 특성에 관계없이 적응적으로 비트율 제어가 이루어졌기 때문이다.

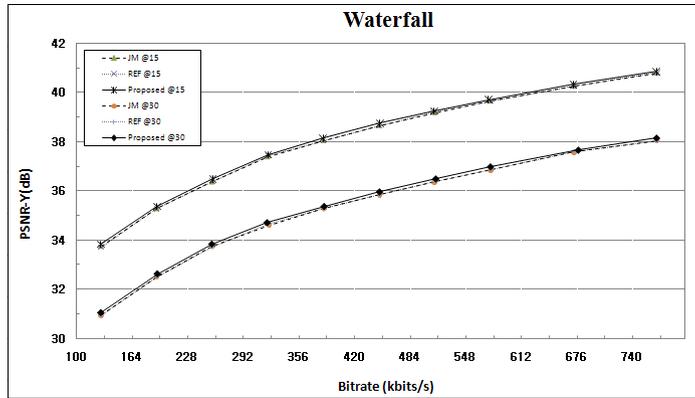


그림 5. Waterfall(CIF) 영상의 PSNR 비교(128Kbps ~ 768Kbps, 260 Frames, @15, @30)

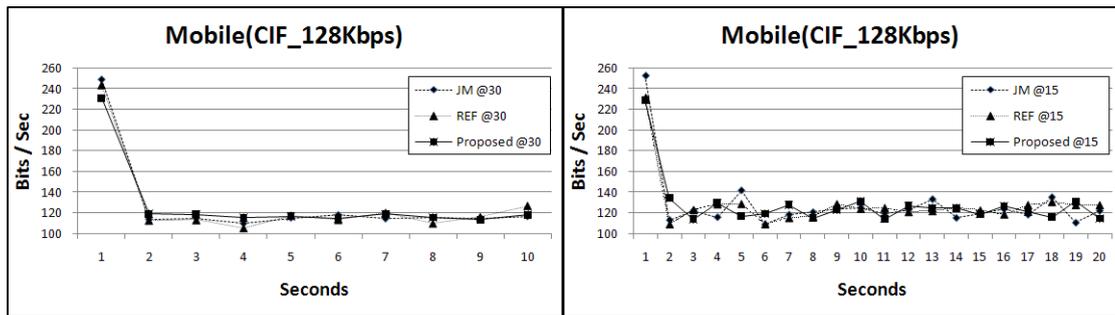


그림 6. Mobile(CIF) 영상의 초당 발생하는 비트량 비교(128Kbps, 300 Frames, @15, @30)

그림 6은 Mobile CIF 영상을 프레임율 @15와 @30에서의 초당 발생하는 비트량을 비교하여 나타난 것으로, I 프레임이 포함된 1초 이후에 발생하는 초당 부호화 비트량이 목표 비트량에 근접하게 발생하고 있음을 확인할 수 있다. 본 논문에서 사용된 모든 실험 영상들에서도 같은 결과를 얻을 수 있었다.

### V. 결 론

본 논문에서는 H.264/AVC 부호기에서 부호화된 비트 스트림을 고정된 채널로 전송할 경우, 기존의 비트율 제어 방법을 소개하고 문제점을 제시하였다. 기존의 비트율 제어 방법의 큰 문제점은 현재 프레임 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 영상 특성에 적응적으

로 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 방법 대비 전체적으로 0.02 ~ 0.43dB의 PSNR 성능 개선을 가져왔다. 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*, 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.*, 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 ISO/IEC MPEG & ITU-T VCEG (ISOWC JTC1/SC29/MCG11 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 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.
- [21] 김명진, 김경환, 홍민철, "실시간 H.264/AVC를 위한 적응적인 프레임 단위 비트율 제어 기법," *방송공학회논문지*, Vol.13, No.6, pp.804-816, November, 2008.
- [22] 김명진, 주원희, 홍민철, "실시간 H.264/AVC를 위한 적응적인 Unit-level 비트율 제어 기법," *한국통신학회논문지*, Vol.35, No.2, pp.161-171, February, 2010.

김 명 진 (Myoung-Jin Kim)

정회원



2002년 2월 한국방송대학교 컴  
퓨터과학과 이학사

2005년 2월 한국방송대학교 정  
보과학과 이학석사

2010년 8월 숭실대학교 정보통  
신공학과 공학박사

2010년 9월~현재 숭실대학교 박

사후 연구원

<관심분야> 영상통신 및 Rate Control, 동영상 부  
호화, 3D 부호화, 영상 복원 및 Enhancement

홍 민 철 (Min-Cheol Hong)

중신회원

한국통신학회 논문지 제34권 10C호 참조