
적응적 프레임 레벨 H.264 비트율 제어

박상현*

Adaptive Frame Level Rate Control for H.264

Sang-Hyun Park*

요 약

본 논문에서는 저전송률 환경에서 영상의 품질을 향상 시키고 전체 영상의 품질 변화를 감소시키는 프레임 레벨 비트율 제어 알고리즘을 제안한다. 제안된 알고리즘에서는 한 GOP에 할당된 비트량을 버퍼 상태와 전송률뿐만 아니라 각 프레임의 특성에 맞게 적절하게 분배함으로써 전체 영상의 품질을 향상시킨다. H.264 비디오 압축 표준은 다양한 압축 모드 및 최적화 방법을 사용하여 압축률을 향상 시키지만 복잡한 인코더 구조는 정확한 영상 압축 제어를 어렵게 한다. 본 논문에서는 먼저 압축된 영상의 특성을 분석하여 저전송률에 맞는 예측 모델을 제시하고, 이를 이용하여 목표 왜곡 값 설정하고 목표 왜곡 값에 맞는 비트량을 할당한다. 제안하는 알고리즘과 기존 알고리즘 간의 비교 실험은 제안하는 알고리즘이 PSNR 성능에서 기존의 알고리즘 보다 우수함을 보여준다.

ABSTRACT

This paper propose a new frame level rate control algorithm for improving video quality and decreasing quality variation of an entire video sequence in a very low bit rate environment. In the proposed scheme, the allocated bits to a GOP are distributed to each frame properly according to the frame characteristics as well as the buffer status and the channel bandwidth. The H.264 standard uses various coding modes and optimization methods to improve the compression performance, which makes it difficult to control the generated traffic accurately. In this paper, proper prediction models for low bit rate environments are first proposed, and a target distortion is determined using the models. According to the target distortion, the bit budget is allocated to each frame. It is shown by experimental results that the new algorithm can generate the PSNR performance better than that of the existing rate control algorithm.

키워드

H.264, 프레임 레벨, 비트율 제어, 영상 압축

* 순천대학교 정보통신공학부 멀티미디어공학전공

접수일자 2009. 05. 28

심사완료일자 2009. 07. 01

I. 서 론

비디오 코딩에서 비트율 제어는 MPEG, H.263, H.264와 같은 다양한 표준에서 다양한 응용에 맞게 연구가 진행되고 있다 [1]. 비디오 인코더를 구현하기 위해서는 매크로블록 모드, 양자화 파라미터 (QP: Quantization Parameter), 프레임 타입, 프레임 레이트와 같은 코딩 파라미터들이 결정되어야 한다. 이러한 코딩 파라미터들을 어떻게 결정하는지는 비디오 표준에 명시되어 있지 않지만 코딩 파라미터의 설정은 비디오 압축에서 성능에 많은 영향을 주게 된다. 특히 비트율 제어에서 QP 값의 설정은 결과 영상의 비트율을 목표 비트율에 맞추게 하거나 버퍼의 오버플로우와 언더플로우를 방지하는데 중요한 역할을 수행한다. QP 값의 설정은 크게는 프레임 단위에서 작게는 매크로블럭 단위로 설정이 될 수 있다. 하지만 H.264 비디오 표준의 경우 프레임 레벨에서 QP를 설정하게 되면 비트율 제어가 부정확하게 되는 단점이 있다 [2-4].

H.264 비디오 표준은 이전의 표준들에 비해 아주 뛰어난 압축 효율 때문에 최근에 많은 관심을 받고 있다 [5]. H.264 비디오 표준은 다양한 영상 압축 기법들이 사용되었다. 특히 RDO (Rate Distortion Optimization)라고 불리는 비트율-왜곡 (R-D: Rate-Distortion) 최적화 움직임 예측과 압축 모드 선택은 비디오 데이터의 압축율을 크게 향상시킨다. 또한 다양한 화면내 예측과 화면간 예측 모드 및 복수개의 참조 영상 사용 등으로 압축율을 극대화한다 [6, 7]. 반면 이러한 특징들은 H.264에서 비트율 제어를 어렵게 한다.

특히 비트율 제어에서 가장 많이 사용되는 모델 기반 알고리즘을 적용할 때, 현재 영상의 통계치를 이용한 모델 적용을 어렵게 한다. 예를 들어, 비트율 모델의 경우 많은 알고리즘에서 현재 영상의 MAD (Mean Absolute Difference)를 사용하고 있다. 하지만 MAD 값은 RDO 처리 후 움직임 예측과 압축 모드가 결정되면 계산이 가능하다. 또한 RDO는 비트율 모델에 따라 QP 파라미터가 결정되어야만 수행될 수 있다. 이러한 RDO와 현재 영상의 통계치 계산 간의 상호 의존적인 구조 때문에 다른 비디오 표준에 비해 H.264에서의 비트율 제어는 복잡한 형태가 된다.

H.264 표준을 위한 다양한 모델 기반 비트율 제어 알고리즘들이 제안되었다. [8]에서는 [9]에서 제안된 2차 비트율-왜곡 모델을 이용하여 베이직 유닛 (BU: Basic Unit)의 QP 값을 예측한다. 이때 BU는 프레임, 슬라이스 또는 매크로블록이 될 수 있다. 현재의 BU에 대한 MAD 값을 계산할 수 없기 때문에 이전 인코딩된 프레임의 MAD 값을 선형 모델에 적용하여 현재의 MAD 값을 예측한다. 그리고 헤더 데이터의 양은 이전 인코딩된 데이터의 값을 그대로 사용한다. 이러한 접근 방식은 크게 두 가지의 문제점이 있다. 첫 번째는 예측된 MAD 값이 정확하지 않다는 것이다. 특히 장면 전환이나 움직임이 많은 영상의 경우 실제의 MAD와 예측된 값과의 차이는 커진다. 두 번째 문제는 헤더 비트를 예측하기가 어렵다는 것이다. H.264 표준은 다양한 압축 모드를 제공하기 때문에 헤더 정보와 관련된 데이터의 양이 프레임에 따라 크게 변한다. 어떤 경우에는 영상 데이터보다 더 많은 양의 헤더 정보가 발생하기도 한다.

RDO와 비트율 제어 간의 상호 의존적인 구조를 해결하기 위해 예측 모델을 사용할 경우 예측이 정확하지 않으면 압축 효율이 나빠지기 때문에 정확한 예측을 위한 방법들이 필요하게 된다. 본 논문에서는 예측이 어려운 H.264 압축 영상의 특성을 고려하여 GOP (Group of Picture) 구조가 IPPP… 인 실시간 압축에 사용될 수 있는 효과적인 프레임 레벨 비트율 제어 알고리즘을 제안한다. 제안하는 알고리즘에서는 H.264 압축 영상의 특성을 분석하여 정확하게 예측하는 예측 모델을 제시하고, 제시된 모델을 사용하여 전체 영상의 품질을 향상시킬 수 있게 각 프레임에 비트량을 할당한다. 그리고 할당된 비트량은 논문 [10]에서 제안된 매크로블록 레벨 비트율 제어 알고리즘을 사용하여 각 매크로블록에 할당한다. 제안하는 알고리즘은 CBR (Constant Bit Rate) 환경에서 H.264 베이스라인 프로파일을 기준으로 구현되었다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 프레임 레벨 비트율 제어 알고리즘에 대해서 설명한다. 3장에서는 제안하는 프레임 레벨 비트율 제어 알고리즘을 설명하고, 4장에서는 제안하는 알고리즘과 JM 알고리즘과의 성능을 비교한다. 마지막으로 5장에서 결론을 맺는다.

II. 연구 배경

H.264 표준에서 비트율 제어는 크게 GOP 레벨, 프레임 레벨, BU 레벨 비트율 제어로 구성된다. GOP 레벨 비트율 제어는 GOP 단위에 비트를 할당하는 것으로, 일반적으로 CBR 환경에서는 일정하게 할당한다. 프레임 레벨 비트율 제어는 각 프레임에 비트를 할당하는 것으로, 현재 버퍼에 남아 있는 데이터 양과 현재 GOP에서 인코딩 해야 할 프레임의 수 및 GOP에 할당된 비트 중 남은 양 등을 고려해서 결정된다. 마지막으로 BU 레벨 비트율 제어는 한 프레임에 할당된 비트가 결정되면 결과 비트가 할당된 비트 양에 맞게 인코딩 되도록 조절하는 역할을 한다.

JM 11.0 소프트웨어[11]의 경우 BU 레벨 비트율 제어를 사용할 때 사용되는 프레임 레벨 비트율 제어 알고리즘은 간단하다. 첫 I 프레임과 첫 P 프레임은 초기 설정된 QP 값에 의해 압축되고, 두 번째 P 프레임부터는 버퍼 상태를 고려하여 일정하게 비트를 할당한다. 먼저 버퍼 상태를 고려하여 n 번째 프레임에 다음과 같이 비트를 할당한다.

$$\widetilde{T}_n = R/F + \gamma(S_n - V_n) \quad (1)$$

여기서 R 은 전송률이고, F 는 프레임 레이트이다. 그리고 S_n 은 목표 버퍼량이고 V_n 은 실제 버퍼량이다. γ 는 목표 버퍼량과 실제 버퍼량의 차이를 어느 정도 반영 할지를 결정하는 파라미터이다. 목표 버퍼량에 맞게 비트를 할당하기 위해서는 γ 가 큰 값을 가지게 한다. 일반적으로 B 프레임을 사용하지 않으면 0.5의 값을 사용한다. 다음으로 GOP에 할당된 비트량 중 남은 비트량을 고려하여 n 번째 프레임에 다음과 같이 비트를 할당한다.

$$\widehat{T}_n = B_n/N_{p,r} \quad (2)$$

여기서 B_n 은 n 번째 프레임을 압축하기 전에 남아 있는 비트량이고 $N_{p,r}$ 은 아직 압축되지 않은 P 프레임의 개수이다.

최종적으로 n 번째 프레임에 할당되는 비트량은 다음과 같다.

$$T_n = \beta \cdot \widehat{T}_n + (1-\beta) \cdot \widetilde{T}_n \quad (3)$$

여기서 β 는 최종 비트량에 \widehat{T}_n 과 \widetilde{T}_n 의 반영 비율을 조절하는 파라미터이다. 목표 버퍼량에 맞게 비트를 할당하는 것에 중점을 둘 때는 β 를 작은 값으로 설정하게 된다.

JM의 경우 버퍼의 상태와 전송률 및 프레임 레이트만을 고려하고 영상의 품질에 관련된 정보들을 고려하지 않고 있다. 논문 [4]에서는 영상의 품질 변화를 감소시키기 위한 프레임 레벨 비트율 제어를 제안한다. 제안된 알고리즘에서는 먼저 이전 프레임의 왜곡 값과 가장 유사한 왜곡을 발생하는 QP에 해당하는 양자화 스텝값 (Q_{step}) Q_n^* 을 다음과 같이 찾는다.

$$V_n(Q_n) = |\widehat{D}_n(Q_n) - D_{n-1}(Q_{n-1})| \quad (4)$$

$$Q_n^* = \arg \min_{Q_n} V_n(Q_n) \quad (5)$$

여기서 Q_n 은 n 번째 프레임의 Q_{step} 이고 $D_n(q)$ 는 Q_{step} 이 q 일 때 n 번째 프레임의 왜곡 값이고 $\widehat{D}_n(q)$ 는 n 번째 프레임의 예측 왜곡 값이다. 현재 프레임의 왜곡 값은 다음 식과 같이 2차원 모델을 이용하여 예측된다.

$$\widehat{D}_n(q) = b_1 \cdot q + b_2 \cdot q^2 \quad (6)$$

여기서 b_1 과 b_2 는 모델 파라미터이다. Q_{step} 이 결정되면 식 (7)과 같이 2차원 비트율 모델을 이용하여 한 프레임에 할당할 목표 비트량 T_n 을 결정한다.

$$\begin{aligned} \widehat{R}_{texture,n} &= (a_1 \cdot q^{-1} + a_2 \cdot q^{-2}) \cdot \widehat{M}_n \\ \widehat{R}_{header,n} &= R_{header,n-1} \\ T_n &= \widehat{R}_{texture,n} + \widehat{R}_{header,n} \end{aligned} \quad (7)$$

여기서 $\widehat{R}_{texture,n}$ 은 예측된 텍스쳐 비트량이고 $\widehat{R}_{header,n}$ 와 $R_{header,n}$ 는 각각 예측 헤더 비트량과 실제 헤더 비트량이다. 그리고 a_1 과 a_2 는 모델 파라미터이고, \widehat{M}_n 은 예측된 n 번째 프레임의 MAD 값으로 다음과

같이 예측된다.

$$\widehat{M}_n = c_1 \cdot M_{n-1} + c_2 \quad (8)$$

여기서 M_n 은 n 번째 프레임의 실제 MAD 값이고, c_1 과 c_2 는 모델 파라미터이다.

논문 [4]의 알고리즘은 프레임 레벨에서 QP를 결정하고 프레임 내의 모든 매크로블록은 결정된 QP를 이용하여 압축된다. 이 경우 저전송률 환경에서는 헤더 비트에 대한 예측이 정확하지 않기 때문에 목표 비트율과 결과 비트율이 큰 차이가 발생하여 정확한 비트율 제어가 이루어지지 않게 된다.

III. 제안하는 알고리즘

제안하는 알고리즘은 각 프레임의 특성을 분석하고 적절하게 비트량을 할당함으로써 전체 영상의 품질을 향상시킬 수 있는 프레임 레벨 비트율 제어 알고리즘이다. 제안하는 알고리즘의 전체 블록 다이어그램은 그림 1과 같다. 제안하는 프레임 레벨 비트율 제어 알고리즘은 할당된 비트량에 맞게 결과 비트를 발생시키는 BU 레벨 비트율 제어 알고리즘과 함께 사용된다.

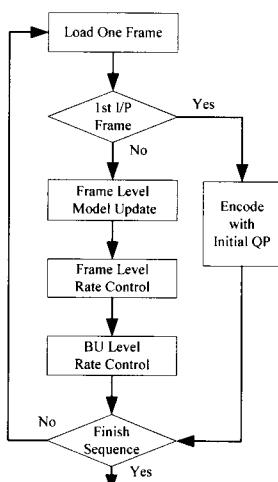


그림 1. 제안하는 알고리즘의 블록 다이어그램.
Fig. 1. Block diagram of the proposed algorithm.

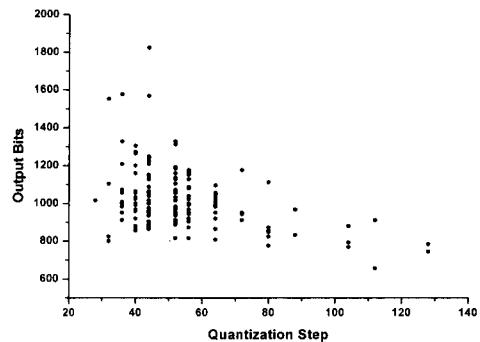


그림 2. 양자화 스텝과 결과 비트량과의 관계.
Fig. 2. Relations between quantization step and output bits.

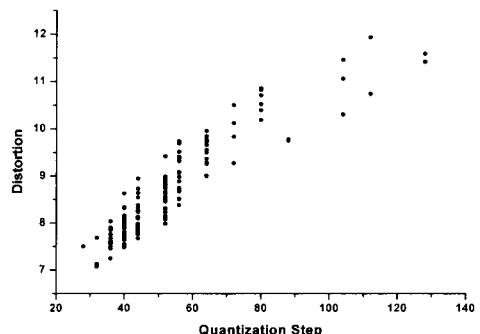


그림 3. 양자화 스텝과 왜곡의 관계.
Fig. 3. Relations between quantization step and distortion.

제안하는 알고리즘에서는 먼저 현재 프레임의 특성을 예측 모델을 이용하여 예측한다. 그림 2와 3은 Foreman 시퀀스를 32 Kbps로 JM 소프트웨어를 이용하여 압축했을 때 각각 Q_{step} 에 따른 결과 비트량과 왜곡 값을 보여준다. 여기서 왜곡값은 MSE (mean square error)를 사용한다. 일반적으로 결과 비트량과 Q_{step} 의 관계는 식 (7)에 의해 모델링되지만 저전송률로 압축된 H.264 영상은 더 이상 식 (7)로 모델링되지 않는 것을 알 수 있다. 반면 왜곡의 경우 Q_{step} 과 왜곡값이 비례하는 것을 알 수 있다. 따라서 제안하는 알고리즘에서는 비트율 모델을 고려하지 않고 왜곡값만 고려한다.

그리고 왜곡 모델은 비례 관계를 모델링하기 위해 다음과 같이 일차 모델을 사용한다.

$$\hat{D}_n(q) = d_1 \cdot q + d_2. \quad (9)$$

제안하는 알고리즘에서는 목표 왜곡 값을 결정하고 목표 왜곡 값과 예측된 현재 프레임의 왜곡 값의 차이를 이용하여 프레임에 비트를 할당한다. 이를 위해 목표 왜곡 값과 σ 를 다음과 같이 정의한다.

$$D_{target} = \frac{\sum_{i=1}^N D_{n-i}}{N}, \quad (10)$$

$$\sigma = \alpha \cdot \frac{\hat{D}_n - D_{target}}{D_{target}}. \quad (11)$$

여기서 N 과 α 는 상수로 논문에서는 각각 20과 4를 사용한다. 그림 4는 Foreman 영상의 왜곡값과 목표 왜곡값의 변화를 보여주고, 그림 5는 왜곡값에 따른 σ 의 변화를 보여준다. 예측된 왜곡값이 목표 왜곡값보다 클 때는 σ 값이 그 차이에 비례하는 양의 값을 가지고, 예측된 왜곡값이 작을 때는 차이에 비례한 음의 값을 가진다. 비트 할당은 σ 를 반영하여 다음과 같이 수행 한다.

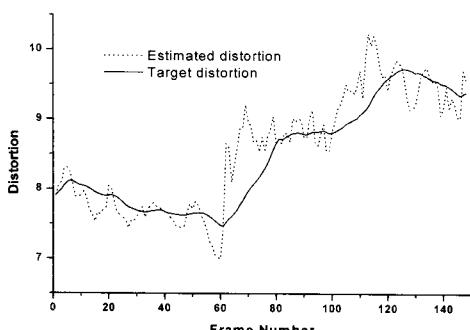


그림 4. 예측 왜곡값과 목표 왜곡 값의 관계.
Fig. 4. Relations between estimated distortion and target distortion.

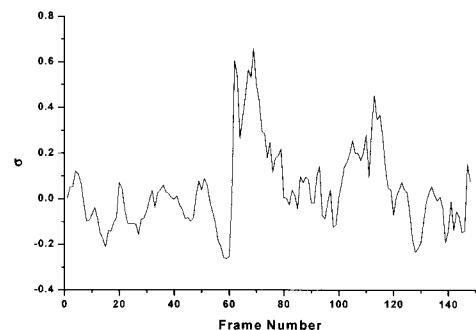


그림 5. σ 값의 변화.
Fig. 5. Variation of the value of σ .

$$\hat{T}_{d,n} = (1 + \sigma)B_n/N_{p,r}. \quad (12)$$

σ 에 의해 너무 많은 비트가 할당되거나 음수의 비트가 할당되는 것을 막기 위하여 σ 의 범위는 다음과 같이 제한된다.

$$\sigma = \begin{cases} 1, & \text{if } \sigma > 1 \\ \sigma, & \text{if } -1 < \sigma < 1 \\ -1, & \text{otherwise.} \end{cases} \quad (13)$$

최종적으로 프레임에 할당되는 비트량은 버퍼 상태를 고려한 \tilde{T}_n 를 함께 고려하여 다음과 같이 결정된다.

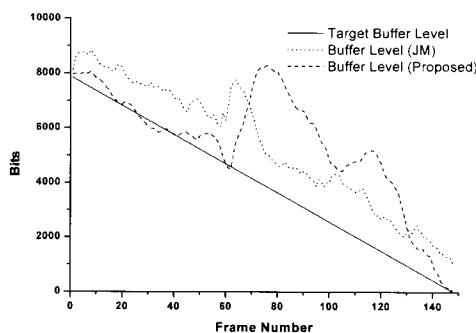
$$T_n = \beta \cdot \hat{T}_{d,n} + (1 - \beta) \cdot \tilde{T}_n. \quad (14)$$

각 프레임에 할당된 비트는 최종적으로 논문 [10]에 제안된 매크로블록 레벨 비트율 제어 방법에 의해 각 매크로 블록에 할당된다.

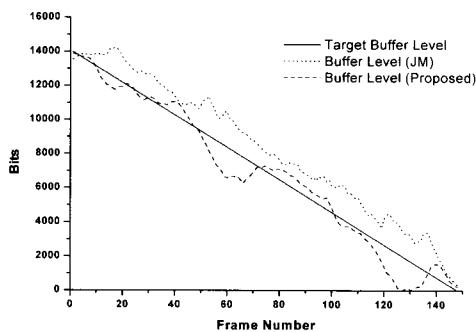
IV. 실험 결과

제안하는 알고리즘의 성능을 분석하기 위하여 JM 11.0에 사용된 비트율 제어 알고리즘 [8]과 성능을 비교하였다. 각 알고리즘은 H.264 베이스라인 프로파일을 기반으로 구현되었다.

본 실험에서는 첫 프레임만 I-프레임으로 코딩하고 나머지 프레임은 P-프레임 타입으로 코딩하였다. 그리고 BU의 단위는 매크로블록으로 설정하였다. 실험에 사용된 영상 시퀀스는 Foreman, Mother & Daughter (MD), Akiyo, Carphone, Salesman 시퀀스이다. 사용된 시퀀스 모두 QCIF 영상 포맷 (176×144)이고 프레임 레이트는 30fps이다.



(a)



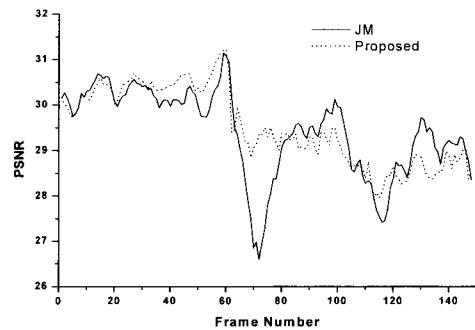
(b)

그림 6. 목표 버퍼 양과 실제 버퍼 양 비교.
(a) Foreman, (b) Akiyo.

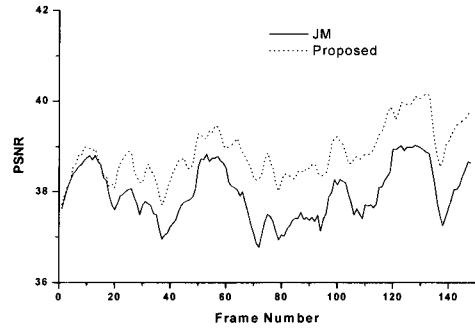
Fig. 6. Comparison of target buffer fullness and actual buffer fullness. (a) Foreman, (b) Akiyo.

그림 6은 Foreman 시퀀스와 Akiyo 시퀀스에 대해서 JM 알고리즘과 제안하는 알고리즘을 적용했을 때 목표 버퍼량과 실제 버퍼량을 비교한 것이다. JM 알고리즘의 경우 실제 버퍼량이 목표 버퍼량보다 큰 값을 전체 GOP 동안 유지하고 있는 것을 볼 수 있다. 이것은 매크로블록 레벨 비트율 제어가 정확하지 않아서 목표 비트량보다

많은 결과 비트량이 만들어지기 때문이다. 그리고 단순히 현재의 버퍼량과 전송률만을 인자로 프레임에 비트를 할당하기 때문에 영상의 특성을 반영하지 못하고 있다. 제안하는 알고리즘의 경우 목표 왜곡값이 이전 프레임들의 평균값으로 설정되는데, 이것은 전체 영상의 왜곡값의 변화를 감소시키는 효과가 있다. Foreman 시퀀스의 경우 62 프레임부터 영상의 움직임이 증가하기 때문에 왜곡값의 변화를 줄이기 위해서는 많은 비트가 할당되어야 한다. 제안하는 알고리즘의 경우 62 프레임부터 버퍼량이 증가하는 것은 왜곡값의 변화를 줄이기 위해 프레임에 많은 비트량을 할당하기 때문이다. 하지만 제안하는 알고리즘의 경우 GOP의 마지막 프레임을 압축했을 때 버퍼량이 목표 버퍼량과 거의 일치하는 것을 볼 수 있다. Akiyo 시퀀스의 경우 영상의 특성에 따라 버퍼량이 목표 버퍼량을 중심으로 변화하고 있는 것을 볼 수 있다.



(a)



(b)

그림 7. PSNR 값 비교. (a) Foreman, (b) Akiyo.
Fig. 7. Comparison of PSNR. (a) Foreman, (b) Akiyo.

그림 7은 Foreman 시퀀스와 Akiyo 시퀀스에 대해서 PSNR 값을 비교한 것이다. 제안하는 알고리즘은 목표 왜곡값을 중심으로 왜곡값을 일정하게 유지하는 효과가 있다. 따라서 움직임이 적은 영상의 경우 비트량을 적게 할당하고 움직임이 큰 영상의 경우 비트량을 많이 할당하게 된다. 이를 통해 영상의 품질 변화를 감소뿐만 아니라 전체 PSNR 값도 향상시킬 수 있다. 그림 7은 제안하는 알고리즘이 JM 알고리즘에 비해 영상 품질을 향상시키는 것을 보여준다.

표 1. PSNR 비교
Table 1. Comparison of PSNR

| 영상 | 알고리즘 | 전송률 | PSNR | STD |
|----------|----------|--------|-------|------|
| Foreman | Proposed | 32kbps | 29.52 | 0.85 |
| | JM 11.0 | 32kbps | 29.41 | 0.96 |
| Akiyo | Proposed | 32kbps | 38.85 | 0.56 |
| | JM 11.0 | 32kbps | 37.96 | 0.60 |
| Salesman | Proposed | 32kbps | 32.62 | 0.73 |
| | JM 11.0 | 32kbps | 32.18 | 0.91 |
| Carphone | Proposed | 32kbps | 32.00 | 0.60 |
| | JM 11.0 | 32kbps | 31.65 | 0.82 |
| MD | Proposed | 32kbps | 35.58 | 1.38 |
| | JM 11.0 | 32kbps | 35.29 | 1.62 |

표 1은 다섯 가지 영상 시퀀스에 대해서 제안하는 알고리즘과 JM 알고리즘을 적용했을 때의 성능을 보여준다. 여기서 STD는 PSNR 값의 표준편차를 의미한다. 제안하는 알고리즘이 영상 시퀀스에 상관없이 PSNR을 향상시키는 것을 알 수 있다. 그리고 PSNR 값의 변화도 제안하는 방식이 JM 방식보다 우수한 것을 알 수 있다.

V. 결론

본 논문에서는 H.264 비트율 제어에서 프레임 레벨 비트율 제어 알고리즘을 제안하였다. JM 알고리즘에서 사용하는 프레임 레벨 비트율 제어는 버퍼 상태와 전송률 및 프레임 레이트만 고려하기 때문에 영상의 특성을 고려한 비트량 할당이 불가능하다. 제안하는 알고리즘은 이전 프레임의 평균 왜곡값을 목표 왜곡값으로 설정

하고 현재 프레임의 예측 왜곡값과의 차이에 비례하여 비트량을 할당함으로써 전체 영상의 영상 품질 변화를 감소시켰다. 또한 움직임이 적은 프레임을 압축할 때 적은 비트량을 할당하고, 남은 비트를 움직임이 많이 프레임에 할당함으로써 전체 영상 품질도 향상시키고 있다.

많은 프레임 레벨 비트율 제어 알고리즘이 제안되었으나, 하지만 기존의 알고리즘에서 사용되는 2차원 비트율 모델이 저전송률 H.264 영상 압축에는 잘 적용되지 않는다. 제안된 알고리즘은 저전송률에서도 비교적 예측이 가능한 왜곡 모델만을 사용하여 효과적으로 프레임 레벨 비트율 제어를 수행하고 있다. 따라서 제안된 알고리즘은 저전송률 환경에서 압축률이 좋은 H.264 알고리즘을 사용하는 실시간 영상 통신에 효과적으로 사용될 것으로 기대된다.

참고문헌

- [1] Z. Chen and K.N. Ngan, "Recent advances in rate control for video coding," *Signal Process.: Image Commun.*, vol 22, pp. 19-38, Jan. 2007.
- [2] H. Song and C.-C. J. Kuo, "Rate control for low-bit-rate video via variable-encoding frame rates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 4, pp. 512 - 521, Apr. 2001.
- [3] Y. Liu, Z. G. Li, and Y. C. Soh, "A Novel Rate Control Scheme for Low Delay Video Communication of H.264/AVC Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 1, pp. 68-78, Jan. 2007.
- [4] Z. Chen and K. N. Ngan, "Distortion variation minimization in real-time video coding," *Signal Process.: Image Commun.*, vol 21, pp. 273-279, Apr. 2006.
- [5] 윤홍준, 서영호, 최현준, 김동욱, "H.264/A VC의 고속 인트라 예측 방법," *한국해양정보통신학회논문지*, vol. 10, no. 7, pp. 1172-1179, 2006년 7월.
- [6] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 688-703, Jul. 2003.

- [7] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 7, pp. 1-19, Jul. 2003.
- [8] 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, Apr. 2006.
- [9] T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 246 - 250, Feb. 1997.
- [10] 박상현, "적응적 베이직 유닛 레벨 H.264 비트율 제어," 한국해양정보통신학회논문지, vol. 13, no. 2, pp. 355-361, 2009년 2월.
- [11] Z. Li, F. Pan, and K. P. Lim, Adaptive Basic Unit Layer Rate Control for JVT Doc. JVT-G012-r1, Thailand, Mar. 2003.

저자소개

박상현(Sang-Hyun Park)

한국해양정보통신학회 논문지
제13권 제2호 참조