

Deflicker-aware Quantization

*남준우 **양정엽 ***전병우

성균관대학교 정보통신공학부

*deity07@skku.edu **binbak@ece.skku.edu ***bjeon@skku.edu

*Nam, Junwoo **Yang, Jungyoup ***Jeon, Byeungwoo

School of Information and Communication Engineering Sungkyunkwan University

요약

H.264/AVC 표준의 화면내 부호화 방법은 화면내 예측을 통한 예측값과 현재 블록의 화소값의 차이를 변환 계수로 변환하고, 이를 양자화 및 엔트로피 부호화하는 순서로 부호화가 진행된다. 그러나 화면내 부호화 방법은 영상간의 시간적 상관성을 이용하지 않고 비교적 상관도가 낮은 공간적 상관성을 이용하기 때문에 화면간 부호화 방법에서는 발견되지 않는 flicker 효과가 발생하게 된다. 일반적으로 flicker 효과는 양자화 및 역양자화 과정에서 변환 계수값의 차이가 크게 증가하기 때문에 발생한다. 따라서 화면내 부호화 방법을 사용하면 flicker 효과로 인해 주관적인 화질이 크게 열화됨에도 불구하고, 이를 위한 해결책이 많지 않다. 본 논문에서는 flicker 효과를 정의하고 flicker 효과의 발생 이유를 분석한다. 이를 바탕으로 flicker 효과를 효과적으로 감소시킬 수 있는 양자화 방법을 제안한다.

1. 서론

H.264/AVC 표준의 화면내 부호화 방법은 화면내 예측을 통한 예측값과 현재 블록의 화소값의 차이를 변환 계수로 변환하고, 이를 양자화 및 엔트로피 부호화하는 순서로 부호화된다[1]. 화면내 부호화 기법은 부/복호화기의 복잡도가 H.264/AVC 표준 전체 복잡도의 25% 정도 밖에 차지하지 않을 만큼 간단하므로, 부/복호화 시간이 단축되고 구현이 용이하기 때문에 보안 카메라나 모바일 영상 등의 다양한 응용에 널리 사용되고 있다. 그러나 화면내 부호화 방법은 화면간 부호화 방법과 달리 인접한 영상간의 시간적 상관성을 이용하지 않고 비교적 상관도가 낮은 공간적 상관성을 이용하기 때문에 정교한 예측값을 생성하지 못한다. 이에 따라, 예측의 오차값이 증가하게 되어 화면간 부호화 방법보다 비트율이 증가하게 되고, 동일한 이유로 화면간 부호화 방법에서는 발견되지 않는 flicker 효과가 발생하게 된다.

Flicker 효과란 시간적으로 연속하는 화면내 부호화 방법으로 부호화된 동일한 위치의 영상에서 밝기값이나 패턴이 심하게 변화하는 현상을 말한다. Flicker 효과는 영상의 주관적 화질을 크게 열화시키는 원인이 되지만, 현재까지는 이에 대한 해결책이 미비한 실정이다. 현재까지 소개된 flicker 현상을 위한 해결책은 대부분 후처리 필터를 이용하고 있다[2],[3],[4]. 후처리 필터를 이용하는 해결책은 기존의 복호화에 추가적으로 후처리 필터를 수행하여 flicker 현상을 제거하기 때문에 부호화 과정에서 발생하는 flicker 효과의 근본적인 원인을 제거하지 못하고 있다. 뿐만 아니라, 부호화기가 flicker 효과를 제거하여 콘텐츠를 공급하는 것이 아니기 때문에 모든 복호화기에 후처리 필터를 적용해야 하는 새로운 문제가 발생하게 된다.

본 논문에서는 이러한 문제를 해결하기 위해 화면내 부호화 기법에서 flicker 효과가 일어나는 근본적인 원인을 분석하고, 이를 부호화 과정에서 해결하는 새로운 방법을 제안한다.

2. 본론

가. Flicker 효과의 정의와 원인

Flicker 효과란 시간적으로 연속하는 화면내 부호화된 동일한 위치의 영상에서 밝기값이나 패턴이 변화하는 현상을 뜻한다. Flicker 효과가 발생하는 이유는 시간적으로 연속하는 동일한 위치의 영상의 화소값이 부호화 과정에 의해 심하게 변화하기 때문에 발생한다. 이러한 변화는 움직임이 있는 전경보다는 움직임이 없는 배경에서 더욱 확연히 관찰된다. 이는 인간의 시각 계통(Human Visual System)이 전경에서 발생하는 화소값의 변화를 움직임의 일부로 인식하기 때문에, 전경에서 발생하는 화소값의 변화는 flicker 현상으로 인식되지 않는다. 반면에 움직임이 없는 배경에서 발생하는 화소값의 변화는 인간의 시각 계통이 이를 잡음으로 인식하기 때문에 영상의 주관적인 화질을 열화시키는 flicker 효과가 된다. 그림 1은 시간적으로 인접한 영상간의 휘도값 차이를 통해 flicker 효과가 발생하는 예를 보여준다. 그림 1 (a)는 원본 Paris 영상이고, 그림 1 (b)는 화면내 부호화 방법으로 부호화된 Paris 영상이다. 그림 1 (a)에서는 두 사람의 움직임만이 영상에 나타날 뿐 배경의 차이는 존재하지 않는다. 그러나 그림 1 (b)에서는 동일한 영상이 지속되는 배경이 변화되는 것을 확인할 수 있다. 즉, 원본 영상에서 움직임이 없는 배경 부분이 부호화 과정을 거치면서 복원된 영상에서 잡음이 발생하는 현상이 flicker 효과이다.

Flicker 효과가 발생하는 원인은 양자화와 역양자화 과정이다. 시간적으로 인접한 동일한 위치의 화소값이 거의 동일함에도 불구하고, 이러한 화소값의 작은 차이가 양자화와 역양자화 과정을 거치면서 크게 증가하기 때문이다. 즉, 원본 영상에서는 flicker 효과로 인식되지 않는 화소값의 작은 차이가 양자화와 역양자화 과정을 거치면서 flicker 효과로 인식되는 큰 차이로 증가하는 것이다.



(a) 원본 Paris 영상 (CIF)



(b) 화면내 부호화 된 Paris 영상 (CIF)

그림 1. Flicker 효과가 발생하는 예

$$e_t(n) < TH_{Flicker} < \hat{e}_t(n) \quad (1)$$

식 1에서와 같이, 시간적으로 연속하는 n 번째 원본 영상의 화소값 차이 $e_t(n)$ 가 양자화와 역양자화 과정을 통해 flicker 효과로 인식되는 임계치 $TH_{Flicker}$ 이상의 화소값 차이 $\hat{e}_t(n)$ 로 증가하는 것이다. 일반적으로 양자화 과정은 N 개의 화소값을 포함하는 블록 단위의 변환 계수에 대해 수행되므로 식 (1)은 식 (2)와 같이 고려되어야 한다.

$$\sum_{n=0}^{N-1} T[e_t(n)] < TH_{Flicker} < \sum_{n=0}^{N-1} T[\hat{e}_t(n)] \quad (2)$$

그림 2 (a)는 식 (2)가 발생하는 예를 보여준다. 시간적으로 인접한 동일한 위치의 블록에 속한 n 번째 계수 $T[f_{t-1}(n)]$ 와 $T[f_t(n)]$ 이 양자화 이전에는 서로 유사한 값이므로 $T[f_{t-1}(n)]$ 와 $T[f_t(n)]$ 의 차이인 $T[e_t(n)]$ 가 $TH_{Flicker}$ 이하이지만, 양자화와 역양자화 과정을 거친 이후의 계수 $Q^{-1}[Q[T[f_{t-1}(n)]]]$ 와 $Q^{-1}[Q[T[f_t(n)]]]$ 이 서로 다른 레벨에 위치하게 되므로 그 차이인 $T[\hat{e}_t(n)]$ 가 $TH_{Flicker}$ 이상으로 증가하기 때문에 flicker 효과가 발생하게 된다. 이때, 양자화 파라미터 값이 커지게 되면 각 레벨 사이의 간격이 더욱 증가하기 때문에 flicker 효과도 더욱 증가하게 된다.

나. Flicker-aware Quantization

진술한 바와 같이, 양자화를 거친 변환 계수의 차이가 양자화 이전의 변환 계수의 차이보다 크게 증가하게 되어 flicker 효과가 발생하고 또한 양자화 파라미터 값이 증가함에 따라 그 효과 역시 더욱 증가하게

된다. 본 논문에서 그림 2 (b)에서와 같이 flicker 효과가 발생하는 블록에 대해 양자화된 변환 계수를 시간적으로 인접한 변환 계수와의 차이가 최소화되도록 조절함으로써 역양자화 이후의 flicker 효과를 감소시키는 양자화 방법을 제안한다. 양자화 과정에서 flicker 효과를 감소시키는 방법은 다음과 같다.

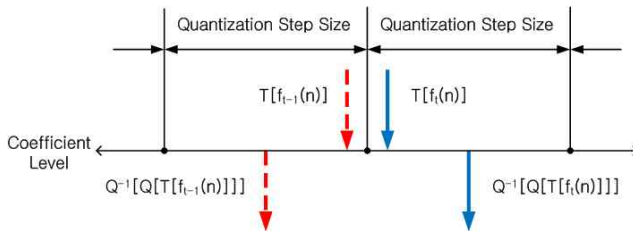
우선 현재 부호화될 블록이 flicker 효과가 발생하는 배경 부분인지 여부를 구분한다. Flicker 효과가 발생하는 배경인 경우에는 본 논문에서 제안하는 양자화 방법을 사용하였고, 그렇지 않은 경우에는 기존의 양자화 방법을 사용하였다. Flicker 효과가 발생하는 배경 부분인지 여부를 구분하는 방법은 식 (3)과 같다.

$$\sum_{n=0}^{N-1} e_t(n) = \sum_{n=0}^{N-1} |f_{t-1}(n) - f_t(n)| < TH_{Motion} \quad (3)$$

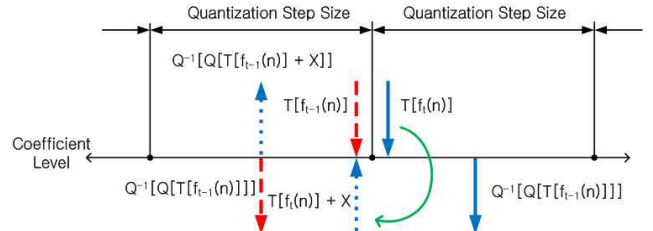
시간상으로 t 번째에 위치한 블록의 n 번째 화소값 $f_t(n)$ 과 시간상으로 $t-1$ 번째에 위치한 영상의 n 번째 화소값 $f_{t-1}(n)$ 의 차이를 해당 블록에 속한 N 개의 화소값에 대해 합하여 배경 부분을 구분할 수 있는 임계값 TH_{Motion} 보다 낮은 경우에만 제안하는 양자화 방법을 적용한다. 식 (3)을 이용하여 현재 블록이 flicker 효과 감소의 대상이 되는 배경으로 선택되면 식 (4)의 $T[\hat{e}_t(n)]$ 가 $TH_{Flicker}$ 이하가 되도록 양자화 계수값을 조절한다.

$$\sum_{n=0}^{N-1} T[\hat{e}_t(n)] = \sum_{n=0}^{N-1} |T[\hat{f}_t(n)] - T[\hat{f}_{t-1}(n)]| < TH_{Flicker} \quad (4)$$

양자화 과정은 선형연산이므로 $T[\hat{e}_t(n)]$ 은 $T[\hat{f}_t(n) - \hat{f}_{t-1}(n)]$ 및 $T[\hat{f}_t(n)] - T[\hat{f}_{t-1}(n)]$ 으로 동일하게 표현될 수 있다.

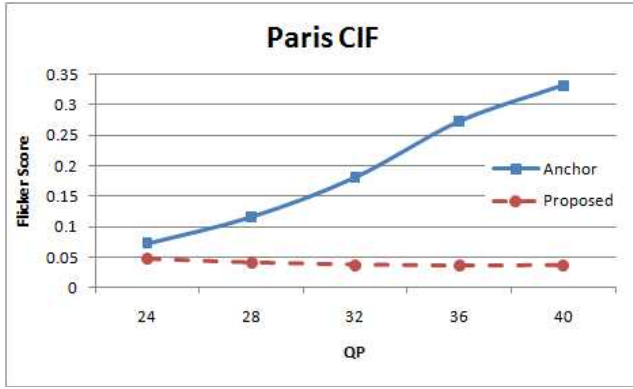


(a) Flicker 효과 발생의 원인

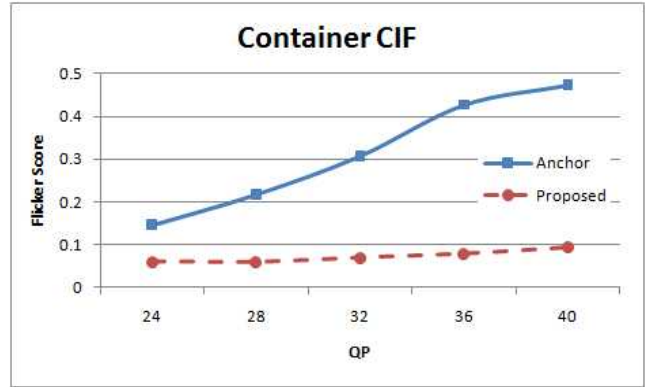


(b) Flicker 효과 감소를 위한 양자화 방법

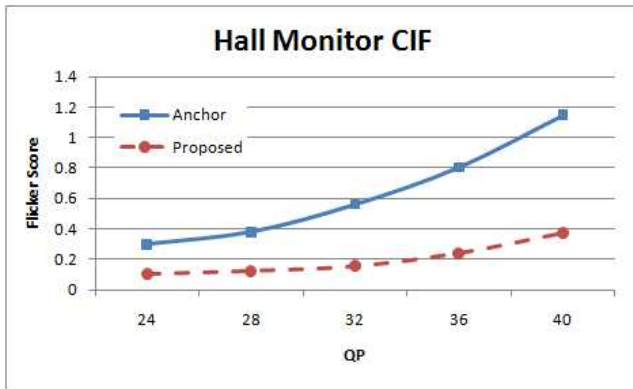
그림 2. Flicker 효과가 발생하는 양자화 과정의 예



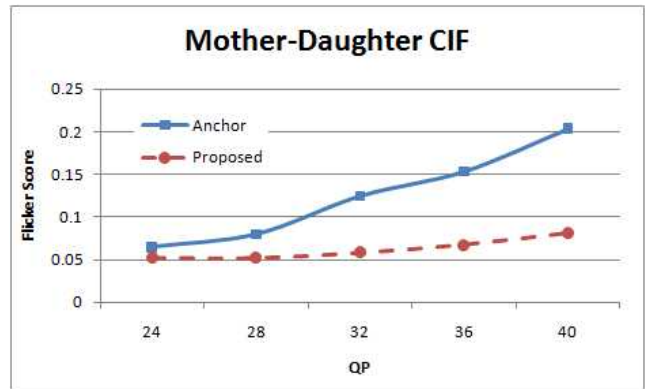
(a) Paris 영상의 Flicker Score (CIF)



(b) Container 영상의 Flicker Score (CIF)



(c) Hall Monitor 영상의 Flicker Score (CIF)



(d) Mother-Daughter 영상의 Flicker Score (CIF)

그림 3. Flicker-aware Quantization의 객관적 성능 평가 (Flicker Score 그래프)

양자화 계수의 조정 방법은 다음과 같다. 우선 이미 복원되어 있는 시간상으로 $t-1$ 번째 위치한 n 번째의 화소값 $\hat{f}_{t-1}(n)$ 을 시간상으로 $t-1$ 번째 위치한 n 번째의 변환 계수 $T[\hat{f}_{t-1}(n)]$ 로 변환한다. 이러한 이유는 양자화와 역양자화 과정이 변환 계수에 대해 수행되기 때문이다. 즉, 양자화 과정을 통해 현재 영상과 이전 영상을 비교하여 flicker 효과가 최소화되도록 조정하기 위해서는 변환 계수에 대해 조정이 수행되어야 하기 때문에 이미 복원된 이전 영상의 화소값 $\hat{f}_{t-1}(n)$ 을 다시 변환 계수로 변환해야 한다.

$$\begin{aligned}
 E &= \sum_{n=0}^{N-1} T[\hat{e}_t(n)] \\
 &= \sum_{n=0}^{N-1} |T[\hat{f}_t(n)] - T[\hat{f}_{t-1}(n)]| \\
 &= \sum_{n=0}^{N-1} |Q^{-1}[Q[T[f_t(n)]] + X] - T[\hat{f}_{t-1}(n)]|
 \end{aligned} \tag{5}$$

이후, 식 (5)와 같이, 시간상으로 t 번째 위치한 n 번째의 변환 계수에 양자화를 수행한 레벨값 $Q[T[f_t(n)]]$ 에 flicker 효과 조정값 X 를 이용하여 flicker 효과 E 가 최소화되도록 레벨값을 조정한다. Flicker 효과 조정값은 $-1, 0, +1$ 을 사용하고, 해당 블록에 포함된 모든 변환 계수에 대해 동일한 과정을 수행한다. 식 (5)에서 $[Q^{-1}[Q[T[f_t(n)]]]$ 와 $T[\hat{f}_{t-1}(n)]$ 는 시간적으로 다른 위치에 있고 공간적으로 동일한 위치에 있는 변환 계수이므로 해당 블록에 동일한 영향을 줄 수 있는 동일한 상태의 연산 가능한 값이 된다. 즉, 서로 연산이 가능해진 두 개의 값인

시간적으로 $t-1$ 번째 위치한 n 번째 변환 계수와 시간적으로 t 번째 위치한 n 번째 변환 계수 N 개를 flicker 효과 E 가 최소화되도록 조정함으로써 부호화된 영상을 전송하면 추가적인 정보나 복호화기 측에서의 추가적인 후처리 과정없이 flicker 효과를 효과적으로 제거할 수 있다.

3. 실험 결과

가. 실험 조건

제안방법의 객관적인 성능을 평가하기 위해 참고문헌 [2]에서 제시된 FS (Flicker Score)를 사용하였다. FS 수치가 낮을수록 flicker 효과

표 1. 실험조건

H.264/AVC JM 15.1	
Profile	Baseline
Entropy	CAVLC
QP	24, 28, 32, 36, 40
블록크기	4x4
RDO	Used
실험영상	Container, Hall Monitor, Paris, Mother-Daughter
영상크기	CIF
프레임 수	300
영상내 예측 모드	Not used



(a) 화면내 부호화 된 Paris 영상 (CIF)



(b) 제안된 양자화 방법을 사용하여 화면내 부호화 된 Paris 영상 (CIF)

그림 4. Flicker-aware Quantization의 주관적 성능 평가

가 감소하는 것을 의미한다. 보다 상세한 실험조건은 표 1과 같다. H.264/AVC 표준의 JM(Joint Model) 15.1의 화면내 부호화 방법으로 CIF 크기의 Container, Hall Monitor, Paris, Mother-Daughter 총 4개의 영상을 양자화 파라미터 24, 28, 32, 36, 40의 5개 값을 사용하여 부호화하였다. 엔트로피 부호화 방법은 CAVLC를 사용하였고 총 300개의 프레임을 부호화하였다. 마지막으로 양자화 과정에 의한 flicker 효과만을 고려하기 위하여 화면내 예측 부호화 방법은 사용하지 않았다.

나. 실험 결과

그림 3은 제안방법의 객관적인 성능을 보여주는 FS 그래프이다. 그림 3의 그래프에서 Anchor는 화면내 예측 모드를 제거하고 부호화한 영상의 FS 수치를 나타내고, Proposed는 화면내 예측 모드를 제거하고 제안방법을 통해 부호화한 영상의 FS 수치를 나타낸다. 각 영상에서의 FS 수치는 Paris 영상의 경우 평균 82.35%, Hall Monitor 영상의 경우 평균 68.33%, Container 영상의 경우 평균 74.15%, Mother-Daughter 영상의 경우 44.72%의 감소율을 보여 제안방법이 flicker 효과 제거에 매우 효과적임을 확인할 수 있다. 뿐만 아니라, 그림 3의 FS 그래프를 보다 상세히 살펴보면, 기존의 화면내 부호화 방법에서는 낮은 양자화 파라미터 값에서는 FS 수치가 감소하고 높은 양자화 파라미터 값에서는 FS 수치가 증가함을 확인할 수 있다. 그러나 본 논문에서 제안하는 방법은 양자화 파라미터 값이 증가하여도 FS 수치의 증가율이 매우 낮은 것을 확인할 수 있다.

그림 4는 Anchor와 Proposed의 주관적인 성능을 보여준다. Paris CIF 크기의 양자화 파라미터 36의 영상을 기존의 방법을 이용하여 부호화 한 영상의 시간적으로 인접한 영상간의 차이를 나타내는 영상과 제안 방법을 이용하여 부호화 한 영상의 시간적으로 인접한 영상간의 차이를 나타내는 영상이다. Anchor의 경우, 그림 1 (b)에서와 동일하게 배경에서 잡음이 관찰되는 것을 확인할 수 있다. 이와는 반대로 Proposed의 경우, 배경에서 flicker 효과가 거의 제거되었음을 확인할 수 없다.

4. 결론 및 향후 연구방향

본 논문에서는 화면내 부호화 방법에서 발생하는 flicker 효과의 발생 원인에 대해 분석하고, 이를 바탕으로 효과적으로 flicker 효과를 제

거할 수 있는 새로운 양자화 방법을 제안하였다. 제안된 방법은 flicker 효과가 FS 수치 기준 평균 약 66.66% 감소됨을 보임으로써, 양자화 과정에서 발생하는 레벨값의 차이가 flicker 효과를 발생시키는 주요 원인 중 하나임을 보여주는 동시에 제안된 방법이 양자화 및 역양자화 과정에서 발생하는 flicker 효과를 제어하기 위한 효과적인 방법임을 보여준다. 뿐만 아니라, 본 논문을 통해 낮은 비트의 영상에서 flicker 효과가 증가하는 것을 확인하는 동시에 제안방법이 영상의 비트율이 낮아질수록 매우 효과적임을 알 수 있다. 본 논문에서 제시한 방법은 비록 H.264/AVC의 JM 15.1을 이용하여 실험을 하였지만 화면내 예측 모드를 제거한 상태에서 실험을 하였기 때문에 JPEG 및 MPEG 등 화면내 예측 모드를 사용하지 않는 모든 분야에서 응용이 될 수 있다. 향후 실험에서는 flicker 효과를 발생시키는 또 다른 원인인 H.264/AVC 표준의 화면내 예측 모드를 제거하지 않은 상황에서 제안방법을 실험하고 그 성능을 측정한다.

5. 참고문헌

- [1] ISO/IEC MPEG and ITU-T VCEG, *Draft ITU-T Recommendation and Final International Standard of Joint Video Specification(ITU-T Rec. H.264, ISO/IEC 14496-10 AVC)*, 2003
- [2] J. Yang, J. B. Park, and B. Jeon, "Flickering Effect Reduction for H.264/AVC Intra Frames", in *SPIE OpticsEast 2006*, vol. 6391, pp.639104.1 - 639101.9, 2006
- [3] S. Delcorso, C. Miro, and J. Jung, "MNR: A Novel Approach to Correct MPEG Temporal Distortions," in *Consumer Electronics*, Vol. 49, No. 1, pp.229-236, 2003.
- [4] J. G. Apostolopoulos and N. S. Jayant, "Postprocessing for Very Low Bit-rate Video Compression," in *Image Processing*, Vol.8, No. 8, pp. 1125-1129, 1999
- [5] Seong Soo Chun, Jung-Rim Kim, and Sanghoon Sull, "Intra Prediction Mode Selection for Flicker Reduction in H.264/AVC", in *IEEE Trans. on Consumer Electronics 2006*, Vol, 52, No.4 pp. 1303 - 1310, 2006