

일반논문-10-15-4-05

고화질 비디오의 부호화 효율성 증대와 연산 복잡도 감소를 위한 디블록킹 필터

정 광 수^{a)}, 남 정 학^{a)}, 조 현 호^{a)}, 심 동 규^{a)†}, 오 승 준^{b)}, 정 세 윤^{c)}, 최 진 수^{c)}

De-blocking Filter for Improvement of Coding Efficiency and Computational Complexity Reduction on High Definition Video Coding

Kwang Su Jung^{a)}, Jung Hak Nam^{a)}, Hyun Ho Jo^{a)}, Dong Gyu Sim^{a)†}, Seoung Jun Oh^{b)}, Sey Yoon Jeong^{c)}, and Jin Soo Choi^{c)}

요 약

본 논문에서는 고화질 비디오의 부호화 효율성 증대를 위한 저 복잡도의 디블록킹 필터를 제안한다. 최근 고해상도 고화질 영상의 사용 증대로 인해 고화질 비디오의 부호화에 적합한 비디오 코덱 알고리즘 연구가 최신 비디오 압축 표준인 H.264/AVC를 기반으로 이루어지고 있다. H.264/AVC의 디블록킹 필터는 낮은 비트율의 비디오 부호화에 적합하도록 설계되었으며, 양자화에 의해 발생하는 블록경계 왜곡을 제거하여 주관적 화질을 향상시킬 뿐만 아니라 부호화 효율을 높인다. 그러나 고화질 비디오의 부호화에서는 부호화 후 복원된 영상 내에 블록화 현상이 약하게 발생하기 때문에 강한 세기의 필터가 적용되는 H.264/AVC의 디블록킹 필터가 적합하지 않다. 또한, 디블록킹 필터는 높은 연산 복잡도 때문에 복호화기에서 높은 비중을 갖는 문제점이 있다. 제안하는 저 복잡도의 디블록킹 필터는 고화질 비디오의 부호화에서 기존의 방법보다 최대 7.3%의 부호화 효율을 높임과 동시에 최대 8.8%의 복호화 연산 복잡도 감소를 보인다.

Abstract

In this paper, we propose a de-blocking filter for improvement of coding efficiency and computational complexity reduction on a high definition video coding. Recently, the H.264/AVC standard-based research for high definition video coding method is under way because the amount of used of high definition videos is on the increase. The H.264/AVC de-blocking filter is designed for low bitrate video coding and it improves not only the subjective quality but also coding efficiency by minimizing the blocking artifact. However, the H.264/AVC de-blocking filter that strong filtering is performed is not suitable in a high definition video coding which occurs relatively low blocking artifact. Also, the conventional de-blocking filter has high computational complexity in decoder side. The computational complexity of the proposed method is reduced about maximum 8.8% than conventional method. Furthermore, the coding efficiency of the proposed method is about maximum 7.3% better than H.264/AVC de-blocking filter.

Keyword: H.264/AVC, de-blocking filter, high definition video

a) 광운대학교 컴퓨터공학과

Dept. of Computer Engineering, Kwangwoon University

b) 광운대학교 전자공학과

Dept. of Electronics Engineering, Kwangwoon University

c) 한국전자통신연구원

ETRI

† 교신저자 : 심동규(dgsim@kw.ac.kr)

※ 본 연구는 지식경제부, 방송통신위원회 및 한국산업기술평가관리원의 산업원천기술개발사업(정보통신)의 일환으로 수행하였음. [KI001932, 차세대 DTV 핵심기술 개발]

·접수일(2010년3월16일), 수정일(2010년6월14일), 게재확정일(2010년6월16일)

I. 서론

최신 비디오 부호화 표준인 H.264/AVC는 ITU-T의 VCEG과 ISO/IEC의 MPEG으로 구성된 Joint video team (JVT)에 의해서 제안된 공동 프로젝트 결과이다^[1]. H.264/AVC는 기존의 비디오 압축 표준보다 화질의 손상 없이 평균적으로 50% 더 높은 압축 성능을 갖는다. H.264/AVC의 주요한 특징으로는 개선된 움직임 보상 기법, 효율적으로 적용된 프레임 내의 예측 기술, 시각적인 화질열화를 최소화하는 방법, 그리고 부호화하는 정보의 특징에 따라 적용되는 가변부호화 기술 등이 있다^{[2][3][4]}. H.264/AVC의 여러 기술 중에서 디블록킹 필터는 블록기반의 부호화 기술에서 부호화 후 복원된 영상 내에 발생하는 블록 경계 왜곡을 최소화함으로써, 주관적인 화질을 향상시키며 동시에 움직임 예측/보상 과정에서 더 정밀한 신호 예측을 가능하게 만들어 부호화 성능을 높게 한다^[5].

최근 비디오 표준화에서는 고해상도 고품질의 영상 사용 증대로 인해 고품질 비디오에 적합한 코덱 개발이 이슈가 되고 있으며, 최신 부호화 표준인 H.264/AVC를 기반으로 한 많은 연구가 진행되고 있다. 그러나 H.264/AVC의 디블록킹 필터는 낮은 비트율의 비디오 부호화에서 발생하는 블록 경계 왜곡 문제를 해결하는데 적합하도록 설계되었기 때문에, 부호화 후 복원된 영상 내에 블록 간 경계 왜곡이 상대적으로 약하게 발생하는 고품질 부호화에는 적합하지 않다^[6]. 또한, H.264/AVC의 디블록킹 필터는 높은 연산 복잡도를 갖기 때문에 복호화기의 전체 연산량에서 높은 비

중을 차지하는 문제점이 있다. 본 논문에서는 고품질 비디오의 부호화 영상 내에 발생하는 블록 경계 왜곡을 제거하기 위해 4×4 블록 경계를 기준으로 각각 최대 한 화소에 대해 고정된 필터를 적용하는 방법을 제안한다. 제안하는 디블록킹 필터는 고품질 부호화 실험에서, 기존 방법과 비교하여 부호화 효율을 높임과 동시에 연산 복잡도를 감소시키는 효과가 있다. 특히, 제안하는 저 복잡도의 디블록킹 필터와 다음 표준화를 위해 선행 작업으로 진행하고 있는 KTA 활동에서 채택한 핵심 기술 중 하나인 적응적 보간 필터(AIF)를 같이 적용하였을 때, 더 높은 부호화 효율을 보인다^[8].

본 논문의 구성은 다음과 같다. II장에서는 기존 H.264/AVC의 디블록킹 필터에 대해서 설명하고 문제점을 분석한다. III장에서는 II장에서 분석한 내용을 바탕으로 제안되는 디블록킹 필터의 특성과 동작과정에 대해 자세히 기술한다. IV장에서는 실험을 통해 제안하는 방법에 대한 성능 평가를 하며, 마지막으로 V장에서는 실험결과 요약 및 향후 연구 방향을 제시하도록 한다.

II. 기존 H.264/AVC의 디블록킹 필터

MPEG-1, MPEG-2, H.263 등의 비디오 부호화 표준에서는 부호화 후 복원된 영상에 대해 아무런 처리 없이 영상 내에 왜곡이 남아 있는 상태로 참조프레임 메모리에 저장하고, 움직임 보상 과정 중 왜곡이 포함된 영상을 참조하기

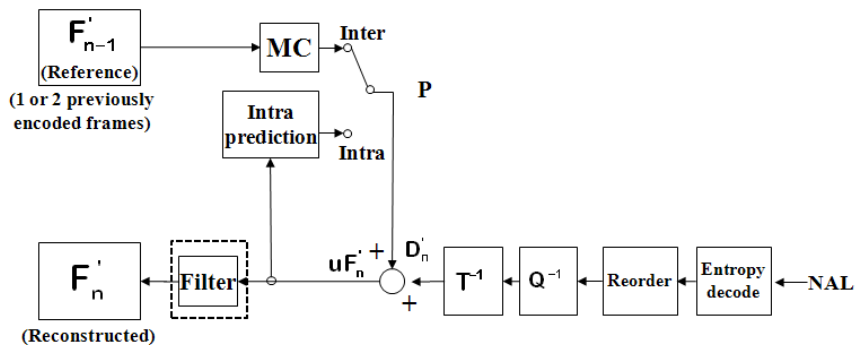


그림 1. H.264/AVC 복호화기의 시스템 블록도
Fig 1. Block diagram of the H.264/AVC decoder

때문에 부호화 된 영상에 화질 열화가 누적되는 문제점이 있다.

H.264/AVC에서는 프레임 메모리에 복원된 영상을 저장하기 전에 디블록킹 필터를 적용함으로써 블록 경계 왜곡을 최소화하여 주관적 화질뿐만 아니라, 객관적 화질을 향상시킨다. 그림 1은 H.264/AVC 복호화기의 시스템 블록도를 나타낸 것이다. 복호화기에 NAL 단위로 전송된 비트 스트림 데이터는 엔트로피 디코딩과 재정렬 과정을 차례로 거친 후 역양자화와 역변환을 통해 차분 신호로 복호화된다. 복호화된 차분 신호는 예측 신호와 가산되어 원 영상으로 복원되며, 그림에서 점선으로 표시된 디블록킹 필터를 거친 후 블록 경계 왜곡이 최소화된 뒤 참조프레임 메모리에 저장된다. H.264/AVC의 디블록킹 필터는 크게 블록 경계마다 필터 처리 강도를 결정하는 경계 세기 결정 단계와 결정된 경계 세기 값에 따라 필터 처리를 수행하는 필터 적용 단계로 나누어 볼 수 있다⁹⁾.

1. 경계 세기 결정 단계

H.264/AVC의 디블록킹 필터는 영상 블록 경계마다 필터 처리 강도를 계산하고, 그 값에 따라 적응적인 필터 처리를 수행한다. 그 이유는 복잡하지 않은 영상 내에서 필터를

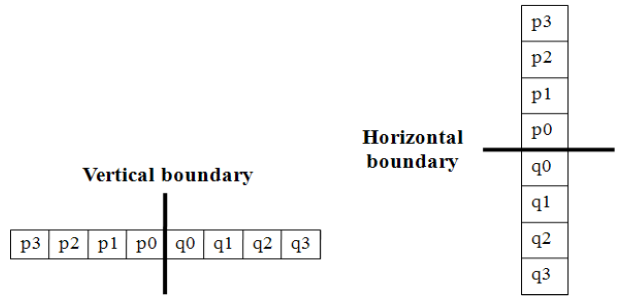


그림 2. 가로, 세로 경계에 인접한 샘플
Fig 2. Samples adjacent to vertical and horizontal boundaries

적용한 결과와 그렇지 않은 결과와의 차이가 거의 없거나, 또는 복잡한 영상 내에서 블록 경계 왜곡이 강한 부분과 그렇지 않은 부분에 동일한 수준의 필터를 적용하게 된다면 불필요한 연산 또는 의도하지 않았던 화질 열화가 발생할 수 있기 때문이다.

H.264/AVC의 디블록킹 필터는 가로, 세로 방향으로 4×4 블록 경계마다 경계 세기 값을 결정하고, 그 값에 따라 필터 처리 강도를 다르게 적용함으로써, 블록 경계 왜곡을 최소화한다. 각각의 매크로 블록마다 128개의 경계 세기 값이 계산되며, 경계 세기 값은 0부터 4까지의 레벨을 가진다. 경계 세기 값이 클수록 강한 필터 처리가 이루어지며, 경계 세기 값의 결정은 블록 경계를 중심으로 인접한

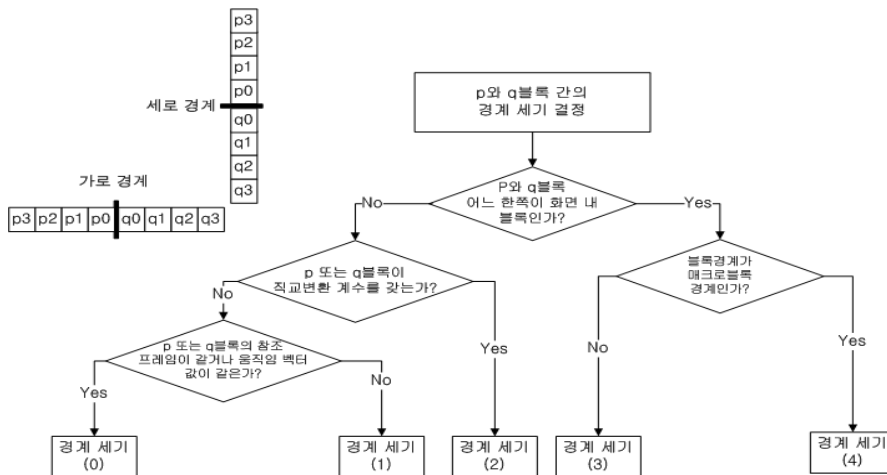


그림 3. 기존의 블록 경계 세기 값 결정 방법
Fig 3. Conventional decision rule of the block boundary strength

두 블록의 특성에 의해 결정된다. 그림 2는 인접한 두 개의 4×4 블록 경계에서 가로, 세로 방향의 샘플 픽셀들을 나타낸 것이고, 그림 3은 기존 H.264/AVC의 디블록킹 필터에서 블록 경계 세기 값을 결정하는 방법을 나타낸 것이다. 그림 3과 같은 순서도에 따라 경계 세기 값이 결정되며, 정해진 경계 세기 값에 따라서 블록 경계마다 적절한 필터 처리가 이루어진다.

표 1은 HD 영상 “Bigships, City, Crew, ShuttleStart, Jets”에 대해서 High 프로파일 인코더를 이용하여 부호화했을 때 발생하는 경계 세기 값의 통계를 구한 것이다. 모든 프레임을 인트라로 부호화하는 I only의 경우에는 경계 세기 값이 3과 4만 계산되기 때문에 영상의 특성에 상관없이 기존 값을 필터링 된 값으로 대체하는 강한 필터링이 적용된다. 그리고 IPPP와 IBBP 부호화 구조에서는 경계 세기 값 중 0과 2가 차지하는 비중이 전체의 90%정도이며 경계 세기 값 1, 3, 4의 비중이 매우 작음을 알 수 있다.

표 1. HD 영상에 대한 경계 세기 값 통계
Table 1. Boundary strength distribution for HD sequences

| Coding Structure | BS = 0 | BS = 1 | BS = 2 | BS = 3 | BS = 4 |
|------------------|--------|--------|--------|--------|--------|
| I only (8Mb/s) | 0 | 0 | 0 | 75.33 | 24.67 |
| IPPP (8Mb/s) | 55.41 | 4.84 | 32.09 | 5.42 | 2.24 |
| IBBP (4Mb/s) | 73.09 | 4.36 | 20.99 | 1.07 | 0.48 |

따라서 상대적으로 블록 오차가 작게 발생하는 고품질 영상에서는 I only 부호화 구조일 때 복원된 화소 값을 필터링 된 값으로 대체하는 강한 필터링 방법보다 영상의 특성을 반영할 수 있도록 양자화 값에 따라 계산된 임계값을 기준으로 필터링된 값을 클리핑할 필요가 있다. 그리고 IPPP와 IBBP 부호화 구조에서는 경계 세기 값 발생 빈도가 0과 2에 집중되고 1, 3, 4는 거의 나타나지 않기 때문에 1, 2, 3, 4를 하나의 대푯값으로 통일하여 동일한 부호화 성능을 얻을 수 있다면 연산 복잡도 감소를 이룰 수 있을 것이다.

2. 필터 적용단계

H.264/AVC의 디블록킹 필터는 블록 경계 간 필터 처리

를 수행하기 위해 서로 다른 특성이 있는 다수의 FIR 필터를 채택하고 있다. 필터 선택은 블록 경계마다 결정된 경계 세기 값과 경계면을 기준으로 양쪽에 위치한 화소 값들을 바탕으로 결정된다. 그리고 선택된 필터의 적용 여부는 양자화 파라미터 값이 고려된 임계값을 통해 결정된다.

경계 세기 값에 따른 필터의 종류와 적용방법을 보면 다음과 같다. 먼저, 경계 세기 값이 1, 2, 또는 3일 경우에는 블록 경계를 기준으로 각각 최대 두 개의 화소에 대해 서로 다른 두 개의 4 탭 필터가 적용될 수 있다. 필터가 적용된 결과 값은 강한 필터 처리를 방지하기 위해 양자화 파라미터와 경계 세기 값이 고려된 범위로 클리핑되며, 그 결과 값은 이후 원 화소 값에 가산 또는 감산된다. 수식 (1)은 경계 세기 값이 1, 2 또는 3일 경우에 수행되는 필터 처리를 나타낸 것이다.

$$O_{x,y} = R_{x,y} \pm Clip(-t_c, t_c, \Delta) \quad (\because \Delta = F[R_{x,y}]) \quad (1)$$

위 식에서 $R_{x,y}$ 는 부호화 후 복원된 것으로 필터를 적용하기 전의 화소 값을 나타내고, $O_{x,y}$ 는 필터가 적용된 최종 결과 값으로서 $R_{x,y}$ 와 필터가 적용된 Δ 값이 클리핑되어 가산 또는 감산된 것이다. 또한, $F[\]$ 는 필터 함수를 나타낸다.

경계 세기 값이 4일 경우에는 블록 경계를 기준으로 각각 최대 세 개의 화소에 대해 서로 다른 다섯 개의 필터가 적용될 수 있다. 필터가 적용된 결과 값은 클리핑 처리가 이루어지지 않고, 최종 결과 값으로 대체된다. 즉, 경계 세기 값이 4일 경우에는 필터링 결과 값이 그대로 대체되기 때문에 강한 필터 처리가 이루어질 수 있는 문제점이 있다. 수식 (2)는 경계 세기 값이 4일 경우에 수행되는 필터 처리를 나타낸 것이다.

$$O_{x,y} = F[R_{x,y}] \quad (2)$$

위와 같이 결정된 필터는 그림 4와 같은 순서에 의해 블록단위로 적용될 수 있다. 그림 4에서는 휘도와 색차 정보에 대해 경계의 필터 적용 순서를 각각 16×16, 8×8 블록단위로 수행하는 것을 나타낸 것이다. 두 블록 모두 4×4 블록단위로 경계선을 따라 필터가 적용된다. 휘도 블록에

서는 먼저 a, b, c, d 의 수직방향에 대해 그리고 e, f, g, h 의 수평방향에 대해 필터 처리가 반복 수행되며, 색차 블록에서도 이와 동일하게 수직 그리고 수평방향 순서로 필터가 적용된다.

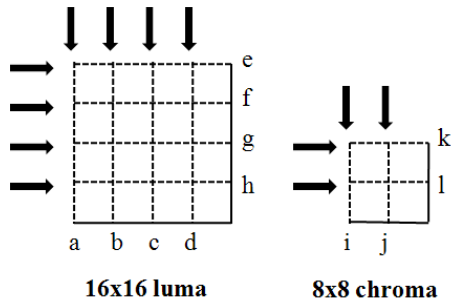


그림 4. 매크로 블록에서 에지의 필터링 순서
Fig 4. Edge filtering order in a macroblock

표 2는 HD 영상을 이용한 다양한 부호화 실험에서 H.264/AVC 디블록킹 필터가 적용되는 화소 중 그림 2에서와 같이 블록 내부 위치 (p1, p2, q1, q2)에 필터링이 적용되는 비율과 필터링 후 나빠지는 화소의 비율을 각각 계산하여 나타낸 것이다.

표 2. 블록 내부 영역에 대한 H.264/AVC 디블록킹 필터의 성능 평가
Table 2. Performance evaluation of H.264/AVC deblocking filter for inner block.

| Coding Structure | | Filtering process ratio (p1/p2/q1/q2) | Disadvantage |
|------------------|--------|---------------------------------------|--------------|
| High Profile | I only | 53.35 | 49.45 |
| | IPPP | 49.33 | 59.02 |
| | IBBP | 46.23 | 62.91 |
| Total Average | | 49.64 | 57.13 |

기존 H.264/AVC의 디블록킹 필터를 이용한 필터링 결과를 보면 전체 필터링 영역 중 블록 내부 영역(p1, p2, q1, q2)이 차지하는 비중이 약 50% 정도임을 알 수 있다. 그리고 블록 내부 영역 중 필터링 후 오차가 더 커지는 부분이 작게는 50%에서 크게는 62% 정도까지 차지함을 확인 할 수 있다. 즉, 기존 H.264/AVC 디블록킹 필터는

HD 영상에 대해서 전체 필터링 처리 중 50%를 블록 내부에 적용하고 있음에도 불구하고 그에 맞는 필터 효과를 얻지 못하고 있다.

이와 같이, 상대적으로 높은 양자화 파라미터에 맞게 설계되어 강한 필터의 특성이 있는 H.264/AVC의 디블록킹 필터는 블록 경계 왜곡이 약하게 발생하는 고화질 비디오 부호화에 적합하지 않다. 이러한 문제점에 착안하여, 본 논문에서는 고화질 비디오에 적합하고 동시에 부호화 성능을 높일 수 있으며, 필터 처리 연산 복잡도를 줄일 수 있는 디블록킹 필터를 제안한다.

III. 제안하는 디블록킹 필터

본 논문에서는 상대적으로 높은 양자화 파라미터에 적합한 기존의 H.264/AVC 디블록킹 필터를 고화질 비디오 부호화에 적합하도록 새롭게 설계하여, 기존의 방법보다 높은 부호화 효율을 보임과 동시에 연산 복잡도를 감소시킬 수 있는 방법을 제안한다. 제안하는 방법은 기존 H.264/AVC의 디블록킹 필터와 같이 가로, 세로 방향으로 4x4 블록 경계마다 경계 세기 값을 결정하고, 그 값에 따라 필터 처리 강도를 다르게 적용하는 적응적인 필터링을 기본으로 한다. 또한, 제안하는 방법은 기존의 방법에서와 같이 경계 세기 값을 결정하는 단계와 필터를 적용하는 단계로 나누어 볼 수 있다. 그러나 각각의 단계에서 적용되는 제안된 방법의 세부적인 알고리즘은 기존의 방법과 차이점을 갖는다.

1. 경계 세기 결정 단계

고화질 비디오의 부호화에서는 블록 경계마다 발생하는 왜곡이 상대적으로 약하기 때문에 강한 필터 특성이 있는 기존 H.264/AVC의 디블록킹 필터가 적합하지 않은 문제점이 있다. 따라서 본 논문에서는 기존 필터가 갖는 강한 필터 특성을 고화질 비디오 부호화에 적합하도록 재설계하는 과정에서 필터링 강도에 대한 레벨의 수를 줄이는 방법을 제안한다.

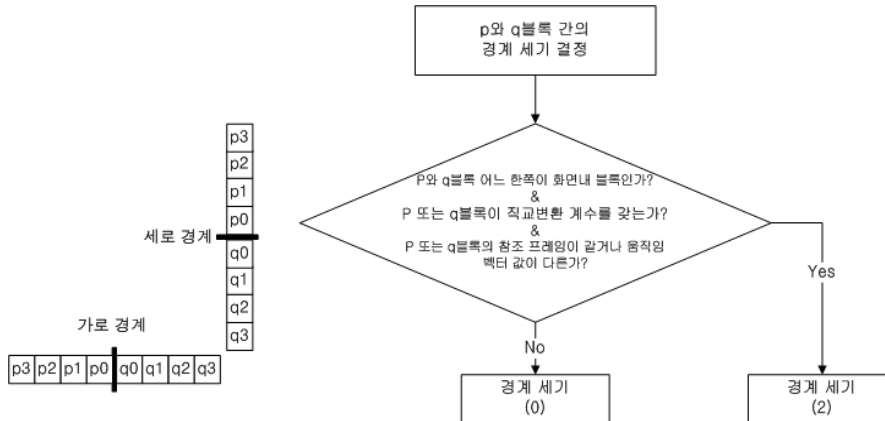


그림 5. 제안된 블록 경계 세기 값 결정 방법
 Fig 5. Proposed decision rule of the block boundary strength

기존 H.264/AVC의 더블록킹 필터의 경우 경계 세기 값을 0부터 4까지, 5레벨로 나누어 계산하는 반면에 제안하는 방법에서는 2레벨로 나누어 계산한다. 그리고 영상에 대한 통계적인 실험에서 1부터 4까지의 경계 세기 값 중, 0과 2가 가장 많은 분포를 나타내는 결과를 바탕으로 제안하는 방법에서는 1부터 4까지의 레벨을 하나의 레벨로 보완하기 위해 2를 대표 값으로 선택하여, 전체 경계 세기 값을 0과 2로 제안하였다. 경계 세기 값 중 0이 결정되기 위한 필요조건은 기존의 방법과 동일하며, 2가 결정되기 위한 필요조건은 기존 더블록킹 필터의 1부터 4까지의 조건 중에서 하나라도 만족하면 된다. 그림 5는 제안된 방법에서 필터 경계 세기 값을 결정하는 방법에 대한 순서도를 나타낸 것이다.

경계 세기 값 계산에 대한 연산 복잡도 비교에서 제안하는 필터 방법은 기존 방법 보다 더 적은 횟수의 분기 연산을 사용한다. 기존 방법에서는 인접한 두 개의 4x4 블록 중 적어도 어느 한쪽이 화면 내 블록 모드일 경우 두 블록 경계가 매크로블록 경계인지를 판단하는 분기 연산이 필요하다. 하지만, 제안하는 방법에서는 인접한 두 개의 4x4 블록 경계가 매크로블록 경계인지 아닌지를 판단할 필요가 없다. 즉, 제안하는 방법은 기존 방법보다 화면 내 슬라이스와 화면 간 슬라이스에서 매크로블록마다 각각 50%, 25% 정도 더 적은 분기 연산이 사용된다. 표 3은 기존 방법과 제안하는 방법에서 경계 세기를 결정할 때 사용되는 분기 연산의

횟수를 화면 내 슬라이스와 화면 간 슬라이스에 따라 비교한 것이다.

표 3. 경계 세기 결정을 위해 사용되는 분기 연산 횟수 비교
 Table 3. Comparison number of branch operations for a decision of boundary strength in a macroblock

| Filtering algorithms | I-slice | P-slice |
|---|-----------|------------|
| H.264/AVC Deblocking filter | 64 branch | 128 branch |
| Proposed deblocking filter | 32 branch | 96 branch |
| Number of branch operations reduction (%) | 50% | 25% |

2. 필터 적용단계

기존의 더블록킹 필터에서는 블록 경계를 기준으로 양쪽의 최대 세 개 화소에 대해 필터링이 수행될 수 있으며, 경계 세기 값에 따라서 필터링된 값의 클리핑 여부가 다르게 결정된다. 4x4 블록마다 경계를 기준으로 양쪽의 최대 세 개 화소에 대한 필터가 적용되는 기존의 방법은 블록 왜곡이 상대적으로 약하게 발생하는 고품질 비디오의 부호화에 적합하지 않다. 또한, 기존의 방법은 경계 세기 값이 4일 경우 필터링된 결과 값이 클리핑 처리되지 않고 최종 결과 값으로 사용되기 때문에 강한 필터링이 될 수 있는 문제점을 갖는다.

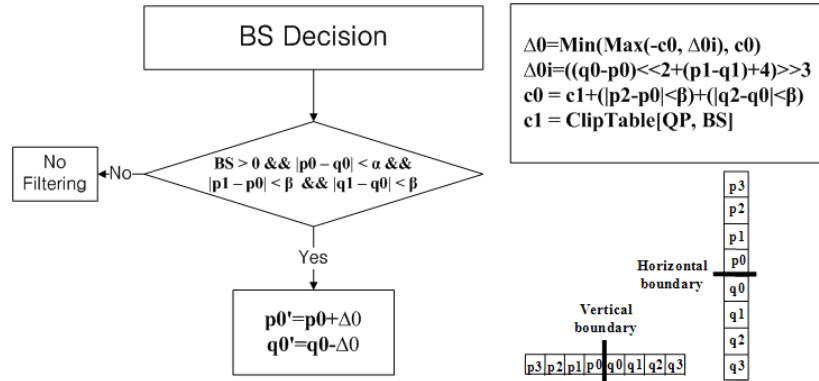


그림 6. 경계 세기 값과 블록의 화소 값을 이용한 필터처리 결정 방법
 Fig. 6. Decision rule of filtering process using the boundary strength and pixel values

제안하는 방법은 고화질 비디오의 부호화에 적합하며, 기존 디블록킹 필터의 강한 필터 특성을 줄일 수 있도록 설계하였다. 자세하게는 필터가 적용되는 블록 내의 화소 값을 블록경계의 양쪽 값으로 제한하였고, 모든 필터링 결과에 대해 클리핑 처리를 수행하도록 설계하였다. 또한, 적용되는 필터의 수를 하나로 제한함으로써, 기존 디블록킹 필터의 복잡한 구조를 간소화시켰다. 제안하는 디블록킹 필터는 그림 6과 같이 필터 경계 세기 값과 블록 내의 화소 값을 통해서 적응적으로 필터 사용을 결정한다.

제안하는 방법에서는 필터 적용 여부를 결정하기 위해 경계 세기 결정 단계에서 계산된 블록 경계 세기 값과 블록 내 화소들의 차분 값을 이용한다. 제안하는 방법에서의 필터 적용 조건은 다음과 같다.

- 가) 경계 세기 값 (BS) > 0
- 나) $|p0 - q0| < \alpha$ and $|p1 - p0| < \beta$ and $|q1 - q0| < \beta$

첫 번째 조건, '가'에서는 블록 경계 세기 값이 0보다 큰 지 아닌지를 판단한다. 경계 세기 값이 0인 경우에는 필터링을 하지 않고, 0보다 큰 2인 경우만 두 번째 조건, '나'에 따라서 필터 적용 여부를 결정한다. 두 번째 조건에서는 블록 내의 네 개의 화소 값 (p1, p0, q0, 그리고 q1)과 양자화 파라미터 값이 고려된 두 개의 임계 값 (α 그리고 β)을 이용하여 필터 처리 여부를 결정한다. α 와 β 는 통계적인 실험을 통해 계산된 값으로 테이블로 정의되어 있으며, 블록 내의

각 화소에 대해 부호화 에러가 있는지 없는지를 판단하는 기준 값으로 사용된다^[10]. 이와 같이, 제안하는 방법에서는 비디오내의 4x4 블록경계 마다 위의 두 가지 조건을 모두 만족할 경우에만 필터를 적용한다.

자세한 필터링 방법은 블록 경계를 기준으로 양쪽 최대한 화소에 대해 필터를 적용하는 것이다. 각 화소 값에 적용

표 4. 기존 H.264/AVC의 디블록킹 필터특성
 Table 4. Filter property of H.264/AVC de-blocking filter

| BS Type | H.264/AVC de-blocking filter | | |
|---------|------------------------------|------------------------|--|
| | Pixel position | Clipping | Filter-tap coefficients |
| 0 | No filtering | | |
| 1 | (p0, q0) | Yes | {1/8, -1/2, 1/2, -1/8} |
| | (p1) | | {1/2, -1, 1/4, 1/4} |
| | (q1) | | {1/4, 1/4, -1, 1/2} |
| 2 | (p0, q0) | | {1/8, -1/2, 1/2, -1/8} |
| | (p1) | | {1/2, -1, 1/4, 1/4} |
| 3 | (q1) | | {1/4, 1/4, -1, 1/2} |
| | (p0, q0) | {1/8, -1/2, 1/2, -1/8} | |
| 4 | (p1) | No | {1/8, 1/4, 1/4, 1/4, 1/8} or {2/4, 1/4, 1/4} |
| | (q1) | | {1/4, 1/4, 1/4, 1/4} |
| | (p2) | | {1/4, 1/4, 1/4, 1/4} |
| | (q2) | | {2/8, 3/8, 1/8, 1/8, 1/8} |
| | (q2) | | {1/8, 1/8, 1/8, 3/8, 2/8} |

하는 필터는 {1/8, -1/2, 1/2, -1/8}의 고정된 계수를 갖는 4 탭 필터이며, 필터링된 값은 양자화 파라미터와 경계 세기 값이 고려된 범위 안에서 클리핑 된다. 표 4와 표 5는 각각 기존 H.264/AVC의 더블록킹 필터와 본 논문에서 제안하는 더블록킹 필터의 특성을 경계 세기 값에 따라 나타낸 것이다.

표 5. 제안된 방법의 필터특성
Table 5. Filter property of proposed method

| BS Type | Proposed de-blocking filter | | |
|---------|-----------------------------|----------|-------------------------|
| | Pixel position | Clipping | Filter-tap coefficients |
| 0 | No filtering | | |
| 2 | (p0, q0) | Yes | {1/8, -1/2, 1/2, -1/8} |

두 가지 방법 모두 경계 세기 값이 0일 경우에는 필터를 적용하지 않는다. 하지만, 경계 세기 값이 0보다 큰 경우에는 제안된 방법과 기존의 방법 사이에서 명백한 차이를 보인다. 기존의 방법과 제안된 방법 사이의 주요한 차이는 다음과 같이 크게 세 가지로 정리 해 볼 수 있다.

가) 블록 경계를 기준으로 필터가 적용되는 화소의 수

나) 필터가 적용된 값의 클리핑 처리 여부
다) 필터 처리를 위한 FIR 필터의 종류

위와 같은 특성을 갖는 제안된 더블록킹 필터는 고화질 비디오의 부호화 실험 결과에서 기존의 방법보다 높은 효율의 부호화 성능을 보이며, 연산 복잡도를 감소시킨다.

필터 적용에 대한 연산 복잡도 비교에서 제안하는 필터 방법은 기존 방법 보다 더 적은 횟수의 덧셈 연산과 시프트 연산을 사용한다. 기존 방법에서는 블록 경계를 기준으로 최대 세 개의 화소까지 필터가 적용되지만 제안한 방법에서는 블록 경계를 기준으로 최대 한 개의 화소에 대해 필터가 적용되기 때문에 제안한 방법이 기존 방법보다 필터 처리에 대한 연산 복잡도가 낮다. 표 6은 기존 방법과 제안하는 방법의 필터링 수행에 사용되는 덧셈과 시프트 연산 횟수를 매크로블록 단위로 비교하여 나타낸 것이다.

기존 방법에서는 화면 내 매크로블록마다 덧셈 연산이 960번, 시프트 연산이 640번 수행되고 화면 간 매크로블록 마다 덧셈 연산이 768번, 시프트 연산이 640번 수행된다. 그리고 제안하는 방법에서는 화면 내 그리고 화면 간 매크로블록에서 덧셈 연산이 384번, 시프트 연산이 256번 수행됨을 확인 할 수 있다. 즉, 제안하는 방법의 필터링 수행에

표 6. 필터 적용을 위해 사용되는 덧셈과 시프트 연산 횟수 비교
Table 6. Comparison number of addition and shift operations for filtering process in a macroblock

| Filtering Algorithms | BS Type | Pixel position | Filter coefficients | ADD | SHIFT | Number of operations for a MB | |
|--|---------|----------------|---|--------|-------|-------------------------------|----------------------|
| | | | | | | Intra MB | Inter MB |
| H.264/AVC deblocking filter | 1,2,3 | (p0, q0) | {1/8, -1/2, 1/2, -1/8} | 3 | 2 | ADD=960 SHIFT=640 | ADD=768 SHIFT=640 |
| | | (p1, q1) | {1/2, -1, 1/4, 1/4}, {1/4, 1/4, -1, 1/2} | 3 | 3 | | |
| | 4 | (p0, q0) | {1/8, 1/4, 1/4, 1/4, 1/8} or {2/4, 1/4, 1/4} | 4 or 2 | 2 | | |
| | | (p1, q1) | {1/4, 1/4, 1/4, 1/4} | 3 | 1 | | |
| | | (p2, q2) | {2/8, 3/8, 1/8, 1/8, 1/8}, {1/8, 1/8, 1/8, 3/8, 2/8} | 5 | 2 | | |
| Proposed filter | 2 | (p0, q0) | {1/8, -1/2, 1/2, -1/8} | 3 | 2 | ADD=384 SHIFT=256 | ADD=384 SHIFT=256 |
| Number of ADD and SHIFT operations reduction | | | | ADD | | 60% | 50% |
| | | | | SHIFT | | 60% | 60% |

사용하는 덧셈 연산과 시프트 연산 사용 횟수는 기존 방법과 비교하여 약 50~60% 더 적음을 확인 할 수 있다. 하지만, 위와 같은 결과는 모든 화소에 필터가 적용되었을 때를 가정한 것이고 실제적으로 더 적은 수의 화소에 대해 필터가 적용되기 때문에 연산 복잡도 측정 결과에서는 예상치보다 낮은 시간 감소율을 보인다.

IV. 실험 결과 및 토의

본 논문에서 제안하는 디블록킹 필터의 부호화 효율과 연산 복잡도 감소를 측정하기 위해 표준화 단체에서 권고하는 실험 조건에서 HD급의 표준 실험영상 4개에 대해 실험하였으며, 기존 H.264/AVC의 디블록킹 필터에 의한 실험 결과와 비교하여 제안된 방법이 부호화 효율과 연산 복잡도 관점에서 보다 좋은 성능을 보임을 확인하였다^[11]. 특히, 제안하는 저 복잡도의 디블록킹 필터는 다음 표준화를 위해 선행 작업으로 진행하고 있는 KTA 활동에서 채택한 핵심 기술 중 하나인 적응적 보간 필터와 같이 적용하였을 때, 보다 높은 부호화 효율을 보인다. 제안하는 방법에 대한 성능 측정을 위해 부호화 효율과 연산 복잡도에 대해 실험을 별도로 진행하였다.

1. 부호화 효율 실험 조건 및 결과

제안하는 방법의 부호화 성능 확인을 위해 KTA 참조 소프트웨어 1.9 버전에 알고리즘을 구현하였다^[12]. 또한, 부호화 성능 실험은 비디오 부호화 표준 단체에서 권고하고 있는 조건에 맞추어 진행하였다. 부호화 성능에 대한 평가는 BD_PSNR (Bjontegaard Delta Peak Signal-to-Noise Ratio)^[13] 모델을 사용하였다. BD_PSNR 모델은 현재 VCEG에서 채택하고 있으며 영상처리 분야에서 널리 사용되고 있는 부호화 성능 측정도구이다. 자세한 실험 조건은 표 7과 같다.

모든 실험 영상에서 각각의 프로파일마다 적용 가능한 예측 구조를 바탕으로 부호화 모드를 설정하고 제시된 시나리오에 따라 실험을 진행하였다. 첫 번째 시나리오는 “KTA | KTA + Proposed (P)”로 KTA에 기존의 H.264/AVC 디블록킹 필터를 사용한 것과 KTA에 제안하는 디블록킹 필터를 적용한 것과의 성능 차이를 비교함으로써 본 논문에서 제안하는 방법의 부호화 효율을 확인 할 수 있다. 그리고 두 번째 시나리오, “KTA + AIF | KTA + AIF + P”를 통해서는 KTA에서 채택한 핵심 알고리즘인 적응적 보간 필터와 제안하는 방법을 같이 적용하였을 때, 제안된 방법이 기존의 방법보다 더 높은 부호화 효율을 나타내는

표 7. 부호화 성능 측정을 위한 실험 조건
Table 7. Test conditions for coding performance evaluation

| Test conditions | Description | |
|------------------------------|---|--------------------|
| | Baseline | High |
| Profile | Baseline | High |
| GOP Structure | I only, IPPP | I only, IPPP, IBBP |
| Transform 8x8 Mode | off | on |
| Entropy Coding Method | CAVLC | CABAC |
| Rate-Distortion Optimization | on | |
| Sequences | 720p60 (1280×720) - BigShips, City, Crew, ShuttleStart | |
| Quantization Parameters | QPISlice = 22, 27, 32, 37 QPPSlice = 23, 28, 33, 38 | |
| Test scenarios | 1. KTA KTA + Proposed (P) 2. KTA + AIF KTA + AIF + P | |

것을 확인 할 수 있다.

먼저, 첫 번째 시나리오에 대한 실험 결과를 보면 표 8과 같다. Baseline과 High 프로파일 모두에서 GOP 구조가 I only일 때, 제안하는 방법이 기존 방법보다 높은 부호화 효율을 나타내는 것을 확인 할 수 있다. GOP 구조가 I only일 때, Baseline 프로파일에서는 평균적으로 약 5.4% 정도의 BD_Bitrate 성능 향상이 있으며, High 프로파일에서는 평균적으로 약 3.5 % 정도의 성능 향상을 보인다. 그리고 “ShuttleStart”영상에서 최대 7.3 % 에 가까운 BD_Bitrate 성능 향상이 있다. 하지만, GOP 구조가 IPPP와 IBBP일 때, Baseline 프로파일과 High 프로파일 모두에서는 제안하는 방법은 기존 방법과 성능이 같거나 약간의 비트 증가가 있음을 확인 할 수 있다. IPPP와 IBBP의 GOP 구조에서 제안한 방법의 성능 효율이 거의 없는 문제는 KTA에 채택된 주요 핵심 알고리즘 중 하나인 적응적인 보간 필터를 같이 적용하였을 때 보완 될 수 있다. 적응적인 보간 필터는 참조 영상과 부호화 영상간의 정수 화소 값을 기반으로 생성되

기 때문에 강한 필터 특성으로 인해 고화질 비디오의 정수 화소 값을 왜곡 시키는 기존의 방법은 오히려 부호화 효율을 감소시킬 여지가 있다. 즉, 제안하는 방법은 기존의 방법보다 적응적인 보간 필터와 잘 맞으며, 보다 높은 부호화 성능 향상을 얻을 수 있다. 자세한 내용은 두 번째 시나리오에 대한 실험 결과를 통해 확인 할 수 있으며, 표 9와 같다.

두 번째 시나리오에 대한 실험 결과에서, 전체적으로 제안하는 방법이 기존의 더블록킹 필터보다 높은 부호화 효율을 나타내는 것을 확인 할 수 있다. GOP 구조가 IPPP일 때, Baseline 프로파일에서는 평균적으로 약 2.8 % 정도의 성능 향상이 있고, High 프로파일에서는 평균적으로 약 1.8 % 정도의 성능 향상이 있다. 그리고 GOP 구조가 IBBP일 때, 제안한 방법이 기존 방법보다 평균적으로 약 1.4% 정도의 비트 감소를 보인다.

표 10은 제안한 방법과 적응적 보간 필터와의 상관관계를 확인하기 위한 실험 결과이다. 제안한 방법에 적응적 보간 필터를 같이 적용할 때와 적용하지 않을 때의 필터 효율

표 8. “KTA | KTA + Proposed”에 대한 실험 결과
Table 8. Experimental result of “KTA | KTA + Proposed”

| Profile | Baseline | | | | High | | | | | |
|------------------|----------|------------|---------|------------|---------|------------|---------|------------|---------|------------|
| | I only | | IPPP | | I only | | IPPP | | IBBP | |
| Coding Structure | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate |
| Bigships | 0.27 | -4.83 | -0.02 | 0.86 | 0.15 | -3.16 | -0.04 | 1.29 | -0.02 | 0.75 |
| City | 0.24 | -3.31 | 0 | -0.04 | 0.14 | -2.07 | 0 | 0.19 | -0.02 | 0.48 |
| Crew | 0.26 | -6.28 | 0.01 | -0.46 | 0.17 | -4.48 | 0 | -0.13 | 0.01 | -0.33 |
| ShuttleStart | 0.3 | -7.35 | 0 | -0.13 | 0.16 | -4.44 | -0.02 | 0.63 | 0 | 0.11 |
| Average | 0.27 | -5.45 | 0 | 0.06 | 0.16 | -3.54 | -0.02 | 0.49 | -0.01 | 0.25 |

표 9. “KTA + AIF | KTA + AIF + Proposed”에 대한 실험 결과
Table 9. Experimental result of “KTA + AIF | KTA + AIF + Proposed”

| Profile | Baseline | | | | High | | | | | |
|----------------------|----------|------------|---------|------------|---------|------------|---------|------------|---------|------------|
| | I only | | IPPP | | I only | | IPPP | | IBBP | |
| Prediction Structure | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate | BD_PSNR | BD_Bitrate |
| Bigships | 0.27 | -4.83 | 0.06 | -1.93 | 0.15 | -3.16 | 0.03 | -0.96 | 0.03 | -0.94 |
| City | 0.24 | -3.31 | 0.06 | -1.92 | 0.14 | -2.07 | 0.03 | -1.03 | 0.01 | -0.39 |
| Crew | 0.26 | -6.28 | 0.07 | -2.36 | 0.17 | -4.48 | 0.04 | -1.6 | 0.03 | -1.35 |
| ShuttleStart | 0.3 | -7.35 | 0.15 | -5.33 | 0.16 | -4.44 | 0.11 | -3.86 | 0.08 | -2.85 |
| Average | 0.27 | -5.45 | 0.09 | -2.88 | 0.16 | -3.54 | 0.05 | -1.87 | 0.04 | -1.38 |

표 10. "IPPP" 부호화 구조에서 제안한 방법의 필터 성능 평가
Table 10. Performance evaluation of the proposed filter for IPPP coding structure

| Coding Structure | | Filtering process ratio | Advantage ratio |
|-------------------|----------|-------------------------|-----------------|
| High profile IPPP | No AIF | 6.21 | 78.52 |
| | With AIF | 9.40 | 78.67 |

을 실험한 것이다. 제안한 방법과 적응적 보간 필터를 같이 적용하는 경우 그렇지 않은 경우보다 필터 적용 비율이 약 3%정도 증가함을 알 수 있고, 동일한 필터 효율을 나타낸다. 즉, 적응적 보간 필터를 같이 적용함으로써 제안한 필터의 효율은 동일하면서 적용 비율이 상대적으로 증가하므로 부호화 성능이 향상됨을 알 수 있다. 그림 7은 "ShuttleStart" 영상에 대한 제안한 방법과 기존 방법의 주관적 화질을 비교한 것이다. (a)는 기존 방법이 적용된 영상이고 (b)는 제안한 방법이 적용된 영상이다. 전체적으로 (b)의 영상에서 선명한 에지를 확인할 수 있으며, 빨간색 네모 영역에서 보다 뚜렷한 차이를 볼 수 있다.

두 개의 시나리오에 대한 실험 결과를 통해서, 제안하는 방법이 I only GOP 구조의 경우 기존의 방법보다 높은 부호화 효율을 나타냄을 알 수 있고, IPPP와 IBBP의 경우 적응적 보간 필터를 같이 적용하였을 때, 기존 보다 높은 부호화 향상을 보임을 확인 할 수 있다.

2. 연산 복잡도 실험 조건 및 결과

제안하는 방법과 기존 방법과의 복호화기 연산 복잡도를 비교하기 위해, 부호화 효율 실험에서와 같이 KTA 참조 소프트웨어 1.9 버전에 알고리즘을 구현하였으며, 실험 조건은 부호화 효율 실험 조건에서 Baseline 프로파일, CAVLC의 경우에 대해서만 진행하였다. 실험 결과에서 복호화기 연산 복잡도는 디블록킹 필터 자체에 대해서만 측정하여 비교한 것이며, 연산 복잡도 측정을 위해서 수식 (3)을 이용하였다. 식에서 R_t 는 기존의 디블록킹 필터에 대한 평균 필터링 시간이고 P_t 는 제안된 방법에 대한 평균 필터링 시간을 측정한 것이다. 수식의 결과는 기존 방법 대비 제안된 방법의 평균적인 필터 수행 시간에 대한 감소율을 나타낸다.

$$\text{Average Time Saving}(\%) = \frac{R_t - P_t}{R_t} \times 100 \quad (3)$$

연산 복잡도에 대한 실험은 각 영상마다 비트스트림을 생성하고, 복호화기를 통해 복호화하면서 디블록킹 필터 자체에 대해 평균적으로 소모되는 시간을 측정하는 방법으로 진행하였다. 그리고 실험 오차를 줄이기 위해 각 영상마다

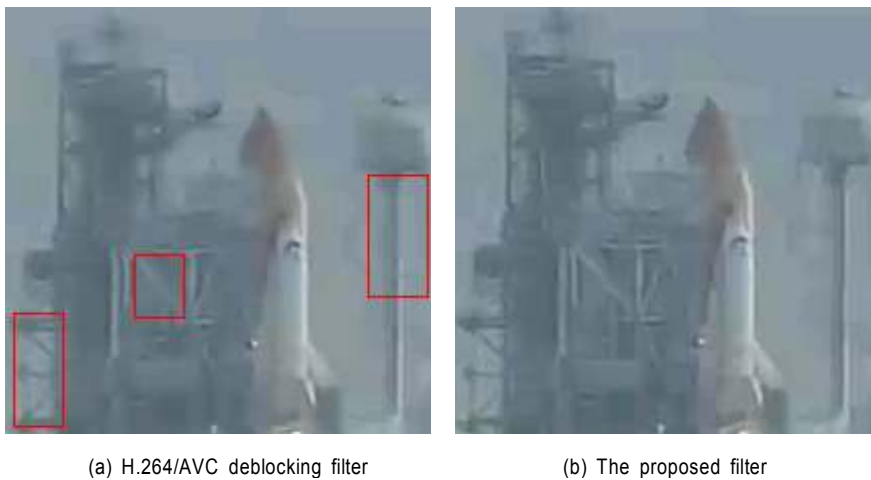


그림 7. 기존 방법과 제안한 방법의 주관적 화질 비교 (ShuttleStart, 30번째 프레임, 양자화 파라미터=37)

Fig 7. Subjective quality comparison between H.264/AVC deblocking filter and the proposed filter (ShuttleStart, frame 30 encoded, QP=37)

표 11. 연산 복잡도에 대한 실험 결과

Table 11. Experimental result for computational complexity

| GOP structure | I only | | | IPPP | | | IBBP | | |
|---------------|-------------|-----------|-------|-------------|-----------|-------|-------------|-----------|-------|
| | BS Decision | Filtering | Total | BS Decision | Filtering | Total | BS Decision | Filtering | Total |
| Bigships | 6.21 | 7.58 | 7.42 | 2.2 | 3.9 | 4.02 | 4.31 | 4.27 | 3.76 |
| City | 2.62 | 6.39 | 6.62 | 5.36 | 0.99 | 4.55 | 3.71 | 4.21 | 3.13 |
| Crew | 5.42 | 7.53 | 8.11 | 3.31 | 8.54 | 5.06 | 3.61 | 5.64 | 4.11 |
| ShuttleStart | 2.3 | 9.46 | 8.82 | 4.9 | 6 | 3.86 | 4.21 | 5.35 | 3.14 |
| Average | 4.138 | 7.74 | 7.743 | 3.943 | 4.858 | 4.373 | 3.96 | 4.87 | 3.54 |

다 약 100번씩 실험하였으며, 전체에 대한 평균값을 계산하여 최종 측정값으로 적용하였다. 실험 결과는 표 11과 같다.

I only의 경우 제안된 방법이 기존의 방법보다 디블록킹 필터 전체에 대한 결과에서 평균적으로 약 7.7 % 정도의 시간 감소율을 보이고, IPPP의 경우 약 4.3 %의 시간 감소율을 나타낸다. 그리고 IBBP의 경우 평균적으로 약 3.5%의 시간 감소율을 보이며, 전체 결과 중 “ShuttleStart” 영상에서 최대 8.8 %의 시간 감소율을 보인다. 실험 결과에서 제안된 방법이 IPPP나 IBBP 보다 I only GOP 구조에서 보다 많은 시간 감소율을 보이는 이유는 기존 H.264/AVC 디블록킹 필터의 경우, I only GOP 구조에서 경계 세기 값이 3과 4만 결정되기 때문이다. 다시 말하면, I only GOP 구조에서 기존의 방법은 블록 경계를 기준으로 최대 세 화소에 대해 필터링이 수행되는 경우가 상대적으로 많기 때문에 최대 한 화소에 대해 필터링을 수행하는 제안된 방법과 비교하여 높은 연산 복잡도를 가지게 된다.

V. 결론

본 논문에서는 고화질 비디오의 부호화에 부적합한 기존 H.264/AVC의 디블록킹 필터의 문제점을 분석 및 보완하여 고화질 비디오의 부호화에서 부호화 효율을 높일 수 있는 새로운 필터 방법을 제안하였다. 제안된 방법은 블록 왜곡이 약하게 발생하는 고화질 비디오의 부호화 특성에 적합하도록 설계됐으며, 실험 결과에서 부호화 효율을 높임과 동시에 부호화기 연산 복잡도를 감소시키는 효과를 보

인다. 자세하게는 기존의 디블록킹 필터가 적용된 KTA 1.9와의 비교 실험을 통해서 최대 7.3% 정도의 BD-Bitrate 향상을 얻을 수 있었고, 최대 8.8% 정도의 연산 복잡도 감소를 보였다. 또한, KTA에 채택된 적응적인 보간 필터를 같이 적용하였을 때, 최대 5.3% 정도의 부가적인 성능 향상을 얻을 수 있었다.

참 고 문 헌

- [1] A. Luthra, G. J. Sullivan, and T. Wiegand, “Introduction to the special issue on the H.264/AVC video coding standard,” IEEE Trans. Circuit Syst. Video technol, vol. 13, pp. 557-559, July 2003.
- [2] 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. 13, pp. 560-576, July 2003.
- [3] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, “H.264/AVC in wireless environments,” IEEE Trans. Circuits Syst. Video technol, vol. 13, pp. 657-673, July 2003.
- [4] S. Wenger, “H.264/AVC over IP,” IEEE Trans. Circuits Syst. Video technol, vol. 13, pp. 645-656, July 2003.
- [5] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, “Adaptive deblocking filter,” IEEE Trans. Circuits Syst. Video technol, vol. 13, pp. 614-619, July 2003.
- [6] Y. L. Lee and H. W. Park, “Loop filtering and post-filtering for low-bit rates moving picture coding,” Signal Processing: Image Communication., vol. 16, pp. 871-890, June 2001.
- [7] 정세윤, 최진수, 김동형, 정원식, 문경애, 홍진우, “H.264 이후의 차세대 비디오 부호화 표준화 동향 및 전망,” 전자통신동향분석, 23권, 122-129쪽, 2008년 2월.
- [8] Y. Vatis, B. Edler, I. Wassermann, D. T. Nguyen and J. Ostermann, “Coding of Coefficients of two-dimensional non-separable Adaptive Interpolation Filter,” Visual Communications and Image Processing, Beijing, China, July 2005.

[9] Iain E. G. Richardson, "H.264 and MPEG-4 Video Compression Video Coding for Next-generation Multimedia," John Wiley & Sons, pp. 184-187, 2003.

[10] S.D. Kim, J. Yi, H.M. Kim, and J.B. Ra, "A deblocking filter with two separate modes in block-based video coding," IEEE Trans. Circuits Syst. Video technol, vol. 9, pp. 156-160, Feb. 1999.

[11] TK Tan, G. Sullivan, T. Wedi, "Recommended Simulation Conditions for Coding Efficiency Experiments Revision 3," ITU-T SC16/Q.6, Doc. VCEG-A110, Berlin, Germany, July 2008.

[12] KTA Reference Software Version 1.9, <http://iphome.hhi.de/suehring/ttml/download/KTA/>

[13] G. Bjontgaard, "Calculation of average psnr differences between rd-curves," ITU-T SG16/Q.6 Doc. VCEG-M33, Austin, Texas, USA, April 2001.

저 자 소 개



정 광 수

- 2008년 2월 : 광운대학교 컴퓨터공학과 학사
- 2010년 2월 : 광운대학교 컴퓨터공학과 석사
- 주관심분야 : 비디오 신호처리, 비디오 압축, 영상 화질측정



남 정 학

- 2006년 2월 : 광운대학교 컴퓨터공학과 학사
- 2008년 2월 : 광운대학교 컴퓨터공학과 석사
- 2008년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 박사과정
- 주관심분야 : 비디오 압축, 멀티프로세서



조 현 호

- 2008년 2월 : 광운대학교 컴퓨터공학과 학사
- 2010년 2월 : 광운대학교 컴퓨터공학과 석사
- 2010년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 박사과정
- 주관심분야 : 비디오 신호처리, 비디오 압축



심 동 규

- 1999년 2월 : 서강대학교 전자공학과 공학박사
- 1999년 ~ 2000년 : (주) 현대 전자
- 2000년 ~ 2002년 : (주) 바로 비전
- 2002년 ~ 2005년 : Univ. of Washington
- 2005년 ~ 현재 : 광운대학교 컴퓨터공학과 (부교수)
- 주관심분야 : 영상신호처리, 영상압축, 컴퓨터 비전

저 자 소 개



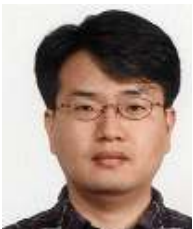
오 승 준

- 1980년 2월 : 서울대학교 전자공학과 졸업(학사)
- 1982년 2월 : 서울대학교 전자공학과 대학원 졸업(석사)
- 1988년 5월 : 미국 Syracuse University 전기/컴퓨터공학과 졸업(박사)
- 1982년 3월 ~ 1992년 8월 : 한국전자통신연구원 근무(멀티미디어연구실 실장)
- 1986년 7월 ~ 1986년 8월 : NSF Supercomputer Center 초청 학생연구원
- 1987년 5월 ~ 1988년 5월 : Northeast Parallel Architecture Center 학생연구원
- 1992년 3월 ~ 1992년 8월 : 충남대학교 컴퓨터공학부 겸임교수
- 1992년 9월 ~ 현재 : 광운대학교 전자공학부 교수 (VIA-Multimedia Center 멀티미디어연구실)
- 2002년 3월 ~ 현재 : SC29-Korea 의장 및 MPEG Forum 부의장
- 관심분야 : 비디오 데이터 처리, 비디오 코덱 시스템, 비디오 및 영상압축, 멀티미디어시스템



정 세 윤

- 1995년 2월 : 인하대학교 전자공학과 학사
- 1997년 2월 : 인하대학교 전자공학과 석사
- 2006년 ~ 현재 : 한국과학기술원 전기및전자공학과 박사과정
- 1997년 ~ 현재 : 한국전자통신연구원 선임연구원
- 관심분야 : HEVC 비디오 부호화 표준, UHDTV/3DTV 비디오 코덱



최 진 수

- 1990년 2월 : 경북대학교 전자공학과 공학사
- 1992년 2월 : 경북대학교 전자공학과 공학석사
- 1996년 2월 : 경북대학교 전자공학과 공학박사
- 1996년 5월 ~ 현재 : 한국전자통신연구원 책임연구원, 실감미디어연구팀장
- 관심분야 : 영상통신, UHDTV방송, 3DTV방송, 데이터방송