

논문 2008-45SP-4-5

# 가변블록 기반 저복잡도 H.264/AVC 디블록킹 필터

## ( Low-Complexity H.264/AVC Deblocking Filter based on Variable Block Sizes )

신 승 호\*, 도 남 금\*\*, 김 태 용\*

( Seung-Ho Shin, Nam-Keum Doh, and Tae-Yong Kim )

## 요 약

H.264/AVC는 기존 압축기술 대비 가변 블록 움직임 보상, 복수 참조 영상, 1/4 화소 움직임 벡터 정확도 및 인-루프 디블록킹 필터 (In-loop Deblocking Filter) 등을 지원하고 있다. 이런 부호화 기술은 압축 효율 향상의 주된 기능인 반면, 동시에 높은 복잡도의 요인으로 작용하고 있다. 저사양, 저비트율의 단말기에서 H.264 부호화 기술의 실제 응용 확대를 위해서는 속도 향상 개선이 필수적이다. 동영상의 주관적 화질을 상당부분 개선할 수 있는 디블록킹 필터 (Deblocking Filter)는 현재 복잡도와 높은 계산량으로 인하여 저사양 단말기에서는 제한적으로 사용되고 있다. 본 논문에서는 실시간 저비트율의 디지털 동영상 압축 시 발생하는 블록킹 현상을 효율적으로 제거하는 디블록킹 필터의 성능개선 방법을 제안한다. 본 논문에서 제안하는 디블록킹 필터링 방법은 움직임 보상에서의 가변블록 정보를 이용하여 영상의 공간적 상관관계를 추출하고, 그 특성에 맞게 4가지 필터모드(Filter Mode)로 분리하여 분리된 영역에 적응형 필터 구조를 취한다. 적용된 모드별 필터링은 블록킹 현상을 제거함은 물론 과도한 블러링 현상(Blurring Effects)을 방지하고 영상내의 세밀한 영상 성분들과 블록 경계간의 실제 에지를 보호함과 동시에 기존 방법 대비 30~40%의 성능향상의 개선을 이루었다.

## Abstract

H.264/AVC supports variable block motion compensation, multiple reference images, 1/4-pixel motion vector accuracy, and in-loop deblocking filter, compared with the existing compression technologies. While these coding technologies are major functions of compression rate improvement, they lead to high complexity at the same time. For the H.264 video coding technology to be actually applied on low-end / low-bit rates terminals more extensively, it is essential to improve the coding speed. Currently the deblocking filter that can improve the moving picture's subjective image quality to a certain degree is used on low-end terminals to a limited extent due to computational complexity. In this paper, a performance improvement method of the deblocking filter that efficiently reduces the blocking artifacts occurred during the compression of low-bit rates digital motion pictures is suggested. In the method proposed in this paper, the image's spatial correlational characteristics are extracted by using the variable block information of motion compensation; the filtering is divided into 4 modes according to the characteristics, and adaptive filtering is executed in the divided regions. The proposed deblocking method reduces the blocking artifacts, prevents excessive blurring effects, and improves the performance about 30~40% compared with the existing method.

**Keywords :** H.264, AVC, deblocking filter, loop filter, variable blocks

\* 정회원, \*\* 학생회원, 중앙대학교 영상공학과  
(GSAIM, Chung-Ang University)

※ 본 연구는 ITRC(Information Technology Research Center)와 서울시 산학연 협력사업, 2008년도 중앙대학교 우수연구자 연구비 지원으로 수행되었음.

접수일자: 2007년11월5일, 수정완료일: 2008년4월29일

## I. 서 론

H.264/AVC<sup>[1]</sup>는 새로운 부호화 기술을 사용함으로써, 기존의 H.263v2 (H.263+)<sup>[2]</sup> 또는 MPEG-4 Simple Profile<sup>[3]</sup>과 비교하여 동일한 화질 대비 최대 50%의 비

트를 압축이 가능하다. H.264의 두드러진 특징으로는 지역 특성의 적용성, 오류 강인성, 네트워크 친화성, 정수단위 변환방식을 사용하고 있으며, 16x16~4x4의 다양한 블록 크기를 지원함으로써 기존 16x16 블록 크기만 지원하는 경우에 비해 15% 이상 비트율 감축이 가능하다. 움직임 예측 단위는 1/4 화소 공간 단위를 이용하여 정수 화소 공간 단위를 이용하는 경우에 비해 20% 이상 비트율 감축이 가능하며, 인-루프 디블록킹 필터(In-loop Deblocking Filter)를 적용함으로써 블록 예측 에러로 인한 오류를 제거한다. 이외에 다중 참조 프레임 지원이라든지 엔트로피 코딩에서 CABAC(Content-based Adaptive Binary Arithmetic Coding)을 이용함으로써 기존 압축방법 대비 많은 부분 압축효율을 향상시켰다<sup>[1]</sup>. 이런 특징으로 인해 저비트율의 이동방송 환경에서는 H.264 코덱(codec) 선택이 당연시 되고 있으며, 앞으로 이동방송뿐만이 아닌 HDTV와 같은 다른 매체에서의 활용도 증가할 것으로 예상된다.

하지만 이런 부호화 기술은 압축 효율 향상의 주된 기능이기도 하지만, 이로 인해 기존 방식보다 복잡도가 더욱 증가하게 되어 연산량의 증가 또한 피할 수 없게 되었다. 이를 해결하기 위해서는 복잡도를 줄이면서 화질 저하를 막을 수 있는 성능 개선이 필요하다.

본 논문에서는 저사양, 저비트율 단말기에서 주관적 화질을 향상시킬 수 있는 디블록킹 필터(Deblocking Filter)의 성능개선 방법에 관하여 제안한다. H.264 규격에서는 블록킹 효과를 줄이기 위해서 디블록킹 필터를 사용하는데 이를 'loop Filter'라고도 부른다. 여기서 블록킹 효과 (Blocking Artifact)라는 것은 매크로블록(Macroblock)간의 양자화 파라미터 (QP : Quantiser Parameter) 값의 차이로 인해 생기는 왜곡 현상으로 저비트율의 동영상에서 영상에 사각형으로 경계가 지면서 깨어져 보이는 현상을 말한다<sup>[4]</sup>. 이런 매크로블록간의 경계부위의 왜곡을 줄이기 위해 디블록킹 필터링을 하게 된다. H.264는 16x16 매크로블록을 4x4 블록까지 세분화해서 처리하므로 4x4 블록간의 경계까지 처리가 가능하다. 하지만 블록킹 효과를 줄이려다가 원치 않게 실제 영상의 에지(edge)가 뭉개지는 원인이 되기도 한다. 또한 계산의 복잡도에 따른 높은 성능의 요구로 인하여 저사양 단말기에서는 적용하지 못하는 경우가 많다. 하지만 이런 단점에도 불구하고 영상에서 블록킹 현상을 제거하는 것만으로도 주관적인 화질을 상당히 높일 수 있기 때문에 디블록킹 필터는 주관적 화질 향상의 가장 핵심적인 기술이라 하겠다. 주관적인 화질

시험을 수행하였을 때 일반적인 의견은 확실히 디블록킹 필터를 사용했을 때와 사용하지 않았을 때 화질의 차이가 뚜렷이 드러난다는 것을 입증한다<sup>[5]</sup>. 이는 부호화의 실제 구현에 있어 고속화 알고리즘을 사용하면 할수록 뚜렷이 나타나는 것으로써, 고속화 알고리즘이 거의 대부분이 이른바 'early-skip' 방식으로 이루어짐에 따라 대 블록간 영상의 차이가 확연히 드러나기 때문이다.

본 논문에서는 움직임 보상(Motion Compensation)<sup>[6]</sup>에서의 가변블록 정보를 이용하여 영상의 공간적 상관관계를 추출하고 디블록킹 필터를 수행함으로써 연산량의 감소를 통해 필터링의 속도를 향상시켰다. 가변블록 정보를 이용한 블록간 경계 분석에 인간의 시각특성과 동영상의 특성을 고려하여 필터모드(Filter Mode)를 4가지로 구분하는 적응형 필터 구조를 취했다.

본 논문의 II장에서는 H.264의 특징적인 부호화 기술인 인-루프 디블록킹 필터와 가변블록 움직임 보상기법에 대해서 소개하고, III장에서는 가변블록 기반의 디블록킹 필터 방법을 제안한다. 제안된 방법은 IV장의 실험을 통해 확인하였으며, 마지막 V장에서는 본 논문의 결론을 맺는다.

## II. H.264/AVC Video Coding

### 1. In-loop Deblocking Filter

블록(block) 기반 부호화 방식에서 블록 에지(edge)들은 일반적으로 내부 화소들 보다 덜 정밀도를 가지고 재현되기 때문에 블록 왜곡 현상이 발생하게 된다. 이를 위하여 H.264/AVC에서는 적응적인 인-루프 디블록킹 필터를 사용하여 블록 왜곡 현상을 제거한다<sup>[7]</sup>.

그림 1과 같이 H.264 디블록킹 필터는 픽처 (picture) 경계의 에지들을 제외하고 모든 4x4 블록 경계의 에지

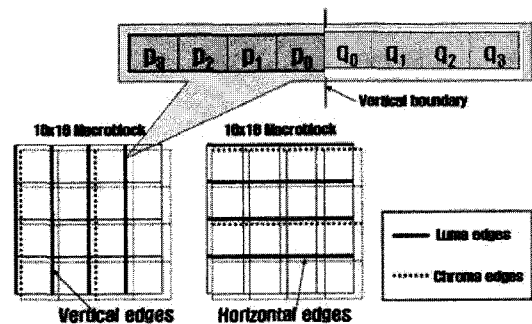


그림 1. 매크로블록에서의 수직/수평 경계 필터링  
Fig. 1. Edge filtering order in a macroblock.

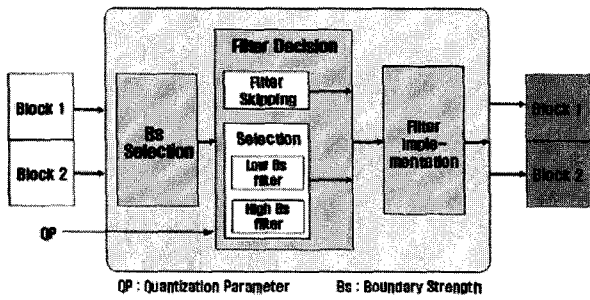


그림 2. 디블록킹 필터 동작절차  
Fig. 2. Deblocking Filter Process.

들에 적용될 수 있다. 각 매크로블록에 필터를 적용하기 위하여, 현재 매크로블록의 위와 왼쪽의 필터링된 화소값을 이용하게 되며, 휘도(luma)와 색차(chroma) 성분을 따로 분리해서 처리한다. 필터링은 각 매크로블록 내의 4x4 블록의 수직/수평 경계로 적용된다. 매크로블록 내부의 왼쪽에서 오른쪽 수직방향으로 먼저 필터링이 진행되고, 수평 에지가 위에서 아래로 진행된다. 16x16의 휘도 성분은 4개의 16화소 에지들에 행해지고, 8x8 색차 성분은 2개의 8화소 에지들에 행해진다.

필터링의 처리는 그림 2와 같이 Boundary Strength, Filter Decision, Filter Implementation의 3과정으로 진행된다<sup>[7]</sup>.

■ Boundary Strength : 블록 에지의 필터링의 필요성 및 세기를 선택하는 과정이다. 필터링 선택의 결정은 그림 1의 p와 q블록에 대해 표 1과 같이 경계세기 파라미터(Bs: Boundary Strength Parameter)를 통하여 결정된다.

이 Bs 파라미터를 통하여 동일한 프레임(frame)의 블록 경계에 모두 같은 강도의 필터링을 적용하지 않고, 블록 경계가 매크로블록 경계인지 아닌지, 또는 어느 블록이 화면 내 부호화를 사용하는지의 여부에 따라 어느 정도의 강도로 필터링을 적용할 것인지를 적용적으로 결정한다. 즉, 블록 왜곡 현상이 생기기 쉬운 부분은 강한 필터링을 적용하고, 반대의 경우는 필터링을 최소화함으로써 불필요한 필터링 때문에 발생하는 화질 열화 증상을 막을 수 있다.

■ Filter Decision : 블록 단계에서 경계에 대한 파라미터가 결정되고 나면, 블록 경계 면의 각 화소들을 조사하여 경계 샘플링(sampling)의 필터링을 결정하는 과정이 화소 단계이다. 에지가 이미지의 사물들을 표현하는 실제 에지인지, 양자화 과정에서 생긴 에지인지를

구분하여, 실제 에지에서는 가능한 필터링을 적게 수행하고, 양자화에 의해 생긴 블록 에지에는 강한 필터링을 적용하기 위해 모든 블록 경계의 화소값을 분석한다. 그림 1의 (p<sub>2</sub>, p<sub>1</sub>, p<sub>0</sub>, q<sub>0</sub>, q<sub>1</sub>, q<sub>2</sub>)로부터의 샘플 그룹들은 수식 (1)의 조건이 성립하는 경우에만 필터링을 적용한다.

$$Bs > 0 \text{ and } |p_0 - q_0| < \alpha; |p_1 - p_0| < \beta; |q_1 - q_0| \leq \beta \quad (1)$$

여기서  $\alpha$ ,  $\beta$ 는 규격에 정의된 임계값(threshold)으로 양자화 파라미터 QP (Quantization Parameter)에 따라 그 값이 정해지는데, 비트열 슬라이스 헤더에 포함된 slice\_alpha\_c0\_offset\_div2와 slice\_beta\_offset\_div2라는 두 개의 파라미터에 의해 강도를 조절할 수 있다. QP가 작을 경우, 경계 주위의 매우 작은 변화 외에는 영상의 특성에 기인한 것이므로 보존되어야 하고, 따라서 임계값  $\alpha$ 와  $\beta$ 는 낮다. QP가 큰 경우, 블록 왜곡은  $\alpha$ 와  $\beta$ 가 높을 때 더 심해지므로 더 많은 경계 샘플이 필터링 된다.

■ Filter Implementation : Boundary Strength와 Filter Decision 단계를 거쳐 아래와 같이 필터링이 적용된다. (Bs<4)인 경우, 입력을 p<sub>1</sub>, p<sub>0</sub>, q<sub>0</sub>, q<sub>1</sub>으로 하는 4-tap FIR 필터 (FIR: Finite Impulse Response Filter)에 의해 p<sub>0</sub>', q<sub>0</sub>'를 생성한다. (Bs=4)인 경우는 표 1과 같다.

표 1. (Bs=4)의 경우 필터링 처리  
Table 1. Filter implementation in the case of (Bs=4).

block	rule	input	filter	output
p	If $ p_2 - p_0  < \beta$ and $ p_0 - q_0  < \text{round}(\alpha/4)$ and this is a luma block	p <sub>2</sub> , p <sub>1</sub> , p <sub>0</sub> , q <sub>0</sub> , q <sub>1</sub>	5-tap	p <sub>0</sub> '
		p <sub>2</sub> , p <sub>1</sub> , p <sub>0</sub> , q <sub>0</sub>	4-tap	p <sub>1</sub> '
	else	p <sub>3</sub> , p <sub>2</sub> , p <sub>1</sub> , p <sub>0</sub> , q <sub>0</sub>	5-tap	p <sub>2</sub> '
	else	p <sub>1</sub> , p <sub>0</sub> , q <sub>1</sub>	3-tap	p <sub>0</sub> '
q	If $ q_2 - q_0  < \beta$ and $ p_0 - q_0  < \text{round}(\alpha/4)$ and this is a luma block	q <sub>2</sub> , q <sub>1</sub> , q <sub>0</sub> , p <sub>0</sub> , p <sub>1</sub>	5-tap	q <sub>0</sub> '
		q <sub>2</sub> , q <sub>1</sub> , q <sub>0</sub> , p <sub>0</sub>	4-tap	q <sub>1</sub> '
	else	q <sub>3</sub> , q <sub>2</sub> , q <sub>1</sub> , q <sub>0</sub> , p <sub>0</sub>	5-tap	q <sub>2</sub> '
	else	q <sub>1</sub> , q <sub>0</sub> , p <sub>1</sub>	3-tap	q <sub>0</sub> '

## 2. 가변블록 크기 움직임보상

H.264의 가변블록 크기 움직임 보상 기술은 기존

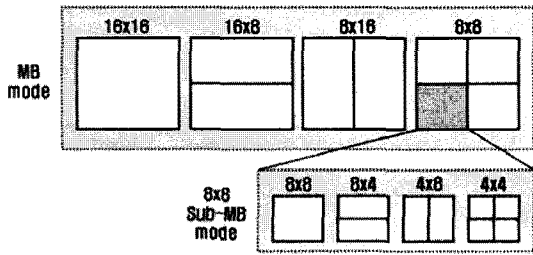


그림 3. 가변블록 크기 움직임보상  
Fig. 3. Variable blocks used for motion compensation.

H.263이나 MPEG-2, MPEG-4 보다 움직임 보상 블록의 크기를 더욱 세밀하게 나눔으로써 영상의 특징이나 영상 내 움직임 특성에 잘 부합하는 기술이다. 즉, MPEG-2에서는 16x16 화소 고정크기 움직임 보상 블록을 사용하고, MPEG-4에서는 16x16, 8x8 화소 두 종류의 움직임 보상 블록을 사용한다<sup>[7]</sup>. 이와 달리 H.264에서는 그림 3과 같이 16x16 화소에서 4x4 화소까지 7 종류의 움직임 보상 블록 크기를 사용하여 움직임을 보상한다.

영상의 평탄한 영역이나 객체의 크기가 큰 경우에는 16x16의 큰 블록으로 움직임 보상이 수행되며, 복잡하고 객체의 크기가 작은 경우는 4x4와 같은 작은 블록으로 움직임 보상이 이뤄진다. 일반적으로 움직임을 위한 블록 크기가 작으면 작을수록 보다 나은 움직임 보상 결과를 얻을 수 있다. 그러나 블록 크기가 작아질수록 더 많은 탐색을 수행하여야 하므로 복잡도가 증가하여 전송해야 할 움직임 벡터 개수도 많아진다. 이러한 문제를 해결하기 위하여 H.264에서는 영상의 특성에 따라 블록의 크기를 선택적으로 적용하는 적응형 움직임 보상 (adaptive motion compensation) 방법을 수행한다<sup>[7]</sup>.

### III. 가변블록 기반 디블록킹 필터

영상의 화면 사이즈(size)가 커질수록 디블록킹 필터가 수행하는 연산의 양은 비례적으로 증가하게 되고, 현재 H.264 표준에서 채택하고 있는 필터링 방식은 이웃 블록의 특성, 참조 픽처(picture)의 특성, I, P, B 부호화에 따라 필터 계수의 선택이 매우 다양하게 되어 있다. 이는 필터 계수 선택에 'if' 명령어를 많이 사용하도록 하여 실제 디블록킹 필터 구현에 있어 'pipelining'에 의한 연산고속화를 거의 기대하지 못하게 만든다<sup>[8]</sup>. 따라서 많은 상용 H.264 코덱(codec)들은 실시간 코딩(coding)을 위해서 디블록킹 필터를 일부러 사용하지

않는 경향이 있는데 이는 시간이 지나면 지날수록 매우 치명적인 화질 열화를 낳는다.

일반적으로 움직임 보상에서의 가변블록은 영상의 미세한 움직임을 나타내는 평탄한 영역이나 큰 물체에서는 16x16, 16x8, 8x16의 블록(MB mode)으로 분할되며, 움직임이 많은 복잡하고 세밀한 영역, 특히 사물의 에지 성분이 많이 존재하는 부분에서는 8x8, 8x4, 4x8, 4x4 블록(Sub-MB mode)으로 분할된다<sup>[9]</sup>. 예를 들어 영상에서 배경과 객체가 분리되는 경계부분은 움직임이 많은 세밀한 부분으로 4x4 블록 모드가 사용될 여지가 많은 부분이라 할 수 있다. 이렇게 분할된 가변블록의 인접 블록들간의 공간적 상관 특성을 디블록킹 필터에 이용하면 필터링의 성능을 향상시킬 수 있다. 또한 인간의 시각시스템(HVS: Human Visual System)은 복잡한 영역보다 평탄하고 단순한 영역의 불연속에 더욱 민감하다. 따라서 단순하고 평탄한 영역에서는 강한 필터링을 적용하여 블록 왜곡 현상을 최대한 줄이며, 반면 복잡하고 세밀한 영역에서는 실제 사물의 에지가 뭉개짐이 없도록 약한 필터링을 적용한다<sup>[10]</sup>.

이에, 본 논문에서 제안하는 가변블록 기반 디블록킹 필터의 특징은 표 2와 같다. 움직임 보상에서 사용되는 가변블록 정보를 활용하여 필터모드를 구분함으로써,

표 2. 제안 디블록킹 필터의 특징  
Table 2. Characteristics of Proposed Deblocking Filter.

Variable Block-based Deblocking Filter 특징
1) 움직임보상에서의 가변블록 크기 정보를 활용하여 경계 블록 간 공간 특성 및 움직임 특성 분석
2) 분석된 블록 특성에 맞게 인간의 시각시스템에 기반하여 4개의 필터모드로 구분하고, 블록단위로 적응적 필터링 수행

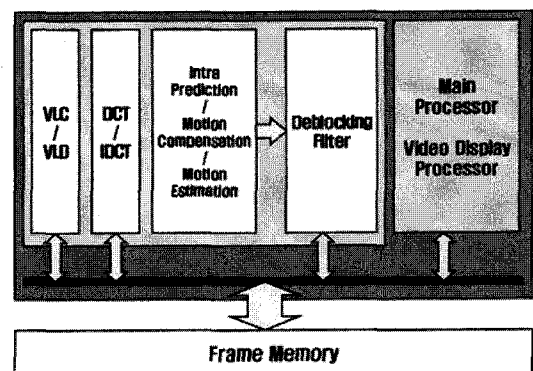


그림 4. H.264 코덱 적용 구성도  
Fig. 4. Modified H.264 codec structures.

기존 더블록킹 필터 방법에서 많은 연산을 필요로 했던 Bs 파라미터 선택 및 경계 화소 분석 단계의 처리량을 상당 부분 줄일 수 있게 되고, 블록 왜곡 현상까지 효율적으로 감소시킬 수 있다.

그림 4는 제안된 방법을 적용하였을 때의 H.264 코덱의 구성도이다. 움직임 보상의 가변블록 정보를 이용하여야 하기 때문에 더블록킹 필터 모듈은 움직임 보상 프로세스(process) 뒤에 위치하게 된다.

필터링은 우선 수직방향에서 시작하여 다음에 수평 방향으로 진행된다. 수직방향 필터링으로 바뀐 화소값은 그대로 수평방향의 필터링에 반영되어 적용된다. 16x16 매크로블록의 4x4 블록단위로 필터링이 수행되는 기존 방법과 달리, 움직임 보상에서 구분된 가변블록 단위로 필터링이 진행된다.

블록 경계간의 필터모드(Filter Mode) 결정은 표 4와 같다. 인간의 시각 특성에 기반하여 블록간 경계를 'flat', 'simple', 'normal', 'complex'의 4가지 영역으로 구분하고, 그에 따라 서로 다른 필터모드가 적용된다. 필터모드가 4일 경우는 영상의 단순한 영역으로 가장 강한 필터링이 적용되며, 필터모드가 1일 경우는 영상의 복잡한 영역으로 가장 약한 필터링이 적용된다.

그림 5, 6, 7은 16x16 가변블록의 수평방향 필터링에 있어 해당 블록의 필터모드를 결정하는 예를 나타낸 그림이다.

그림 5는 16x16 블록의 인접블록이 16x16 블록이거나 8x16 블록인 경우이다. 이 경우, 경계면을 둘러싼 영

상을 'flat region'으로 가정하고 필터모드 4를 부여하여 가장 강한 필터링을 적용한다. 블록 경계뿐 아니라 블록 내부의 왜곡 현상도 없애야 하기 때문에, 16x16 매크로블록을 기준으로 4개의 휘도(luma) 성분과 2개의 색차(chroma) 성분의 모든 4x4 블록 에지를 필터링 한다. 필터링 적용은 기존 방법과 동일하다.

그림 6은 16x16 블록의 인접블록이 16x8 블록이거나

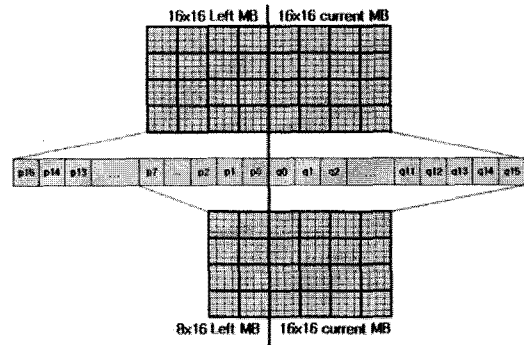


그림 5. 16x16 블록에서의 필터모드 4의 선택(예)  
Fig. 5. Filter Mode (4) decision of 16x16 variable blocks.

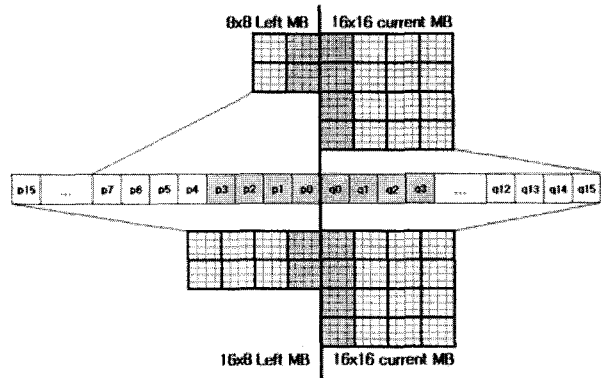


그림 6. 16x16 블록에서의 필터모드 3의 선택(예)  
Fig. 6. Filter Mode (3) decision of 16x16 variable blocks.

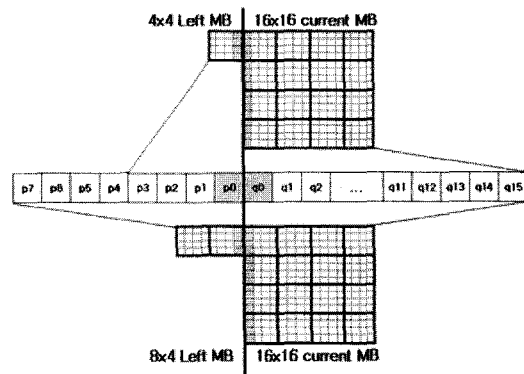


그림 7. 16x16 블록에서의 필터모드 1의 선택(예)  
Fig. 7. Filter Mode (1) decision of 16x16 variable blocks.

표 3. 필터모드의 결정  
Table 3. Decision of Filter Mode.

Region	p, q 화면 내 부호화 관계	Filter Mode	Filter Strength
flat	p and q are MB mode and boundary is a 16-pixel boundary	4	Strongest
simple	p and q are MB mode and boundary is a 8-pixel boundary	3	Strong
normal	p and/or q is sub-MB mode and boundary is a 8-pixel boundary	2	normal
complex	p and/or q is sub-MB mode and boundary is a 4-pixel boundary	1	weak
-	otherwise	0	no filtering

(p: adjacent block, q: current block)

8x8 블록인 경우이다. 이 경우는 경계면을 둘러싼 영상을 'simple region'으로 가정하여 필터모드 3을 부여한다. 해당 블록 경계면의  $p_0 \sim p_3, q_0 \sim q_3$ 의 8개의 화소들에 대하여 필터링을 적용한다. 블록 경계를 중심으로  $p_3, p_2, p_1, p_0, q_0, q_1, q_2, q_3$ 의 8화소를 필터링 대상으로 정하여 (1/16, 1/16, 1/8, 1/8, 1/4, 1/8, 1/8, 1/16, 1/16)의 가중치를 갖는 1차원 9탭 평탄화 식을 적용한다.

그림 7은 16x16 블록의 인접블록이 8x4 블록이거나 4x4 블록인 경우이다. 이 경우는 영상 내 사물의 실제 에지가 존재하는 복잡한 'complex region'으로 가정하고 필터모드 1을 부여하여 가장 약한 필터링을 적용한다. 해당 블록 경계면을 중심으로  $p_0$  와  $q_0$ 의 2개의 화소들에 대해서만 필터링을 적용한다. 가장 세밀하고 주의 깊게 필터링이 되어야 하기 때문에 다음과 같은 평탄화 식을 적용하여  $p_0', q_0'$ 의 값을 구한다.

$$d = (3p_1 - 8p_0 + 8q_0 - 3q_1) / 16. \quad (2)$$

$$d' = \text{sign}(d)\text{Max}[0, |d| - \text{Max}(0, 2(|d| - QP))]. \quad (3)$$

$$p_0' = p_0 + d'. \quad (4)$$

$$q_0' = q_0 + d'. \quad (5)$$

특이한 점은 필터링 블록이 MB mode인 16x16, 8x16 블록이며 인접블록이 Sub-MB mode인 8x8 또는 4x8 블록인 경우 필터 모드 3의 경우에 비해 좀 더 세밀한 'normal region'으로 가정하여 필터모드 2를 부여한다는 점이다. 이 경우는 블록 경계간 왜곡 효과와 실제 사물의 에지가 공존한다고 가정하여 해당 블록 경계면을 중심으로  $p_0, p_1, q_0, q_1$ 의 4개의 화소 범위로 필터링을 적용한다. 블록 경계값 차의 1/5을  $p_1$ 에 1배를 더하고,  $p_0$ 에 2배를 더한 평탄화 식을 적용한다. 마찬가지로  $q_0$ 에 2배,  $q_1$ 에 1배를 빼준다.

$$d = (p_0 - q_0) / 5. \quad (6)$$

$$p_1' = p_1 + \text{sign}(d) \times |d|. \quad (7)$$

$$p_0' = p_0 + 2 * \text{sign}(d) \times |d|. \quad (8)$$

$$q_0' = q_0 - 2 * \text{sign}(d) \times |d|. \quad (9)$$

$$q_1' = q_1 - \text{sign}(d) \times |d|. \quad (10)$$

이런 방식으로 7개의 가변블록들에 대한 인접블록을 조사하여 적용적으로 필터모드를 결정하고, 블록단위로 동일한 필터링 식을 적용한다. 수직방향의 필터링도 같은 과정으로 진행된다. 수평 및 수직 방향의 인접블록

표 4. 가변블록에서의 수직/수평 경계 필터모드

Table 4. Filter Mode adjacent to edge boundaries in variable blocks

block		16x16	16x8	8x16	8x8	8x4	4x8	4x4
16x16	H	4	4	3	3	2	1	1
	V	4	3	4	2	2	2	2
16x8	H	3	4	3	3	2	1	1
	V	3	3	3	2	2	2	2
8x16	H	3	3	3	2	2	1	1
	V	4	3	4	2	2	2	2
8x8	H	2	2	2	2	2	1	1
	V	2	2	2	2	1	2	1
8x4	H	2	2	2	2	2	1	1
	V	1	1	1	1	1	1	1
4x8	H	1	1	1	1	1	1	1
	V	2	2	2	2	1	2	1
4x4	H	1	1	1	1	1	1	1
	V	1	1	1	1	1	1	1

(H: horizontal filtering, V: vertical filtering)

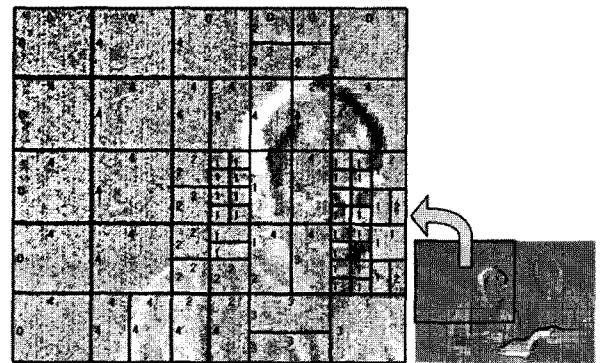


그림 8. 가변블록 경계의 필터모드 적용 예

Fig. 8. Example of filter mode selection that used variable blocks.

에 대한 필터모드의 관계는 표 4와 같다.

그림 8은 가변블록 경계간 필터모드 선택의 적용 예를 나타낸 그림이다.

기존 방식은 매크로블록 내 모든 4x4 블록 경계면 화소들의 'line by line' 분석을 통하여 필터링의 적용여부를 판단하기 때문에 많은 계산량을 필요로 하지만, 본 논문에서 제안하는 방법을 사용하면 가변 블록 단위로 필터링의 적용여부를 판단함으로써 계산량을 감소시킬 수 있다.

#### IV. 실험 및 평가

본 논문에서 제안된 디블록킹 필터의 성능을 측정하기 위하여 H.264 표준화 그룹에서 권고하고 있는 JM

(Jonit Model)<sup>[11]</sup> version 10 부호기를 이용하여 실험하였다. 화질에 미치는 영향을 평가하기 위해, H.264 표준화 그룹에서는 비트율의 차이는 %, PSNR의 차이는 차분값으로 비교하는 것을 권고하기 때문에 수식 (11)과 같이  $\Delta PSNR$ 을 이용하여 결과를 측정하였다. 또한 디블록킹 필터 계산량 감소의 성능개선 측정을 위하여 수식 (12)를 이용하여 구하였다.

$$\Delta PSNR(dB) = PSNR_R - PSNR_P \quad (11)$$

$$\Delta COMP(\%) = (COMP_R - COMP_P) / COMP_R \times 100 \quad (12)$$

(R: 기존방법 디블록킹 필터, P: 제안방법 디블록킹 필터)

실험에 사용된 시퀀스(sequence)는 다양한 특성을 갖는 영상으로 구성하였고, 이동방송과 같은 저비트율 환경에 많이 쓰이는 형식인 QCIF로 휘도와 색차 성분은 4:2:0으로 샘플링 하였다. 영상 내 부호화되는 I, P 프레임에 양자화 파라미터 QP의 단계별 변화를 주어 실험하였다.

실험 영상들의 PSNR의 차분값과 계산감소량을 나타낸 표 5의 실험 결과를 보면, 평균적인 PSNR 감소는 0.1~0.25(dB) 정도로 객관적 화질에 변화가 거의 없었으며, 전체 평균 수행시간은 기존 방식대비 30~40(%) 감소를 보였다. 필터링 적용 판별을 위한 'if'와 'for' 문의 수행 횟수가 약 25(%) 줄어든 것이 전체 필터링 수행시간 감소에 영향을 준 것으로 판단된다.

그림 9는 실험 영상 중 하나인 'foreman'의 화질 평가 비교 및 구체적 결과를 나타낸 그래프이다. 주관적

표 5. 실험 영상의 평균 결과 값  
Table 5. Average Results of the Experimental Sequences.

Sequence (100 frames)		QP				avg.
		22	26	30	34	
container	R1	-0.21	-0.19	-0.26	-0.29	-0.24
	R2	38.83	32.99	27.86	25.16	31.21
stefan	R1	-0.23	-0.21	-0.19	-0.24	-0.22
	R2	41.24	39.28	38.31	37.25	39.02
mobile	R1	-0.17	-0.15	-0.13	-0.17	-0.16
	R2	35.83	37.91	36.74	40.57	37.76
news	R1	-0.13	-0.14	-0.15	-0.14	-0.14
	R2	37.36	38.17	36.71	34.52	36.69
foreman	R1	-0.19	-0.11	-0.19	-0.22	-0.18
	R2	37.84	33.45	35.37	32.21	34.71

(R1:  $\Delta PSNR$ (dB), R2: 계산감소량(%))



(a) 기존방법 적용 (b) 제안방법 적용

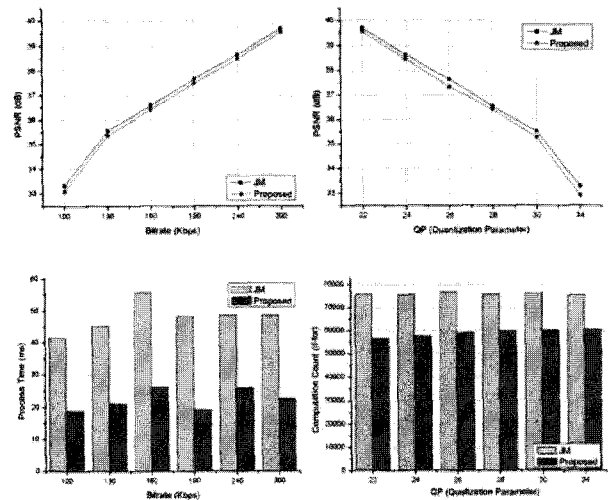


그림 9. 실험 영상 'foreman'의 실험 결과 (QP = 28)  
Fig. 9. Experimental results of the 'foreman' (QP = 28).

화질의 경우도 그림 9의 (a), (b)와 같이 큰 차이가 없음을 확인할 수 있었다.

### V. 결론

H.264/AVC 디블록킹 필터는 계산의 복잡도로 인하여 무선 통신이나 이동 전화 등의 실시간 처리 능력이 제한된 저사양 단말기에서 구현 및 적용에 어려움이 있다. 또한 H.264 부호화 방식의 최적화가 진행될수록 디블록킹 필터의 연산량은 비례적으로 증가하게 된다. 본 논문에서 제안된 가변블록 기반 디블록킹 필터 방법은 움직임 보상에서의 가변블록 정보를 이용하여 디블록킹 필터를 선택적으로 수행함으로써, 기존의 'line by line'의 화소 단위로 이루어지는 방법보다 성능이 향상되었음을 확인할 수 있었다. 다시 말해, 필터링 적용을 위한 영상의 특성을 따로 분석할 필요가 없으며, 영상 특성에 맞게 평탄한 영역은 강한 필터링을 수행하여 블록 현상을 최대한 감소시키고, 복잡영역은 약한 필터링을 적용함으로써 영상의 선명도를 최대한 유지함과 동시에 가변 블록 단위로 필터링을 수행하여 계산 속도를 줄임으로써 전체적으로 화질 저하 없이 필터링의 속도가 향

상됨을 확인할 수 있었다. 실험을 통해 큰 부호화 손실 없이 약 30~40% 정도의 계산 감소가 가능하다는 것을 입증하였다.

따라서 제안된 방법을 사용하면 디블록킹 필터의 복잡도를 감소시켜 DMB와 같은 이동방송 환경에서의 저사양 휴대 단말기에서 디블록킹 필터의 구현이 가능할 것으로 예상된다.

Syst. Video Technol, vol.9. pp.156-160, Feb.1999.  
[11] JVT software JM10.2, May. 2006.

## 참 고 문 헌

- [1] Draft ITU-T Recommendation and Final Draft International Standard of Joint Specification (ITU-T Rec. H.264/ISO/IEC 14496-10 AVC), Mar. 2003.
- [2] Draft ITU-T Recommendation H.263, "Video Coding for low bitrate communication Telecommunication Standardization Sector of International Telecommunication Union," Oct.1995.
- [3] ISO/IEC 14496-2, "Information technology-coding of audio-visual objects. Part 2: Visual," Dec.2001.
- [4] Y.L. Lee and H.W. Park, "Loop filtering and post-filtering for low-bitrates moving picture coding," Signal Processing Image Commun., vol.16, pp.871-890, 2001.
- [5] Peter List, Anthony Jock, Joni Lainema, Gisle Bjontegaard, and Marta Karczewicz, "Adaptive Deblocking Filter," IEEE Trans Circuits. Syst. Video Technol, vol.13, No.7, July.2003.
- [6] Z. Zhou, M.T. Sun, and Y.F. Hsu, "Fast variable block-size motion estimation algorithm based on merge and slit procedures for H.264 / MPEG-4 AVC," International Symposium on Circuits and Systems, Vol.3, pp.725-728, May 2004.
- [7] E.G. Richardson, "H.264 and MPEG-4 Video Compression," John Wiley & Sons, pp.170-187, Sept.2003.
- [8] Yuwen Huang, Towei Chen, "Architecture Design for Deblocking Filter in H.264/AVC," in Proceedings of ICME, pp.693-696, Baltimore, Maryland, USA, July 6-9, 2003.
- [9] A. Ahmad, N. Khan, S. Masud, and M.A. Maud, "Selection of variable block sizes in H.264," IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP '04), Vol.3, pp.173-176, May 2004.
- [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



저 자 소 개



신 승 호(정회원)  
 2000년 인하대학교 공과대학 학사  
 2004년 서강대학교  
 미디어공학과 석사  
 2004년~현재 중앙대학교  
 영상공학과  
 박사과정(수료)

2003년~2005년 AIRCODE DTV기술연구소  
 2005년~현재 TU Media Corp. 방송기술연구팀  
 <주관심분야 : 멀티미디어방송, 영상 및 신호처  
 리, MPEG/H.264, 표준화>



도 남 금(학생회원)  
 1996년 중앙대학교  
 전자계산학과 학사  
 1996년~2001년 (주)동양시스템즈  
 2002년~2006년 (주)코디얼  
 2006년~현재 중앙대학교  
 영상공학과 석사과정

<주관심분야 : 영상 및 신호처리, MPEG/H.264>



김 태 용(정회원)  
 1986년 한양대학교  
 전기공학과 학사  
 1988년 한양대학교  
 전자통신공학과 석사  
 1999년 포항공과대학교  
 컴퓨터공학과 박사

2003년~현재 중앙대학교 영상공학과 교수  
 <주관심분야 : 컴퓨터비전, 영상통신, 인공지능,  
 게임>