

일반논문-11-16-1-01

## 고효율 비디오 부호화를 위한 적응적 인-루프 필터 방법

정 광 수<sup>a)</sup>, 남 정 학<sup>a)</sup>, 임 웅<sup>a)</sup>, 조 현 호<sup>a)</sup>, 심 동 규<sup>a)‡</sup>, 최 병 두<sup>b)</sup>, 조 대 성<sup>b)</sup>

### Adaptive In-loop Filter Method for High-efficiency Video Coding

Kwang Su Jung<sup>a)</sup>, Jung Hak Nam<sup>a)</sup>, Woong Lim<sup>a)</sup>, Hyun Ho Jo<sup>a)</sup>, Dong Gyu Sim<sup>a)‡</sup>,  
Byeong Doo Choi<sup>b)</sup>, and Dae Sung Cho<sup>b)</sup>

#### 요 약

본 논문에서는 고효율의 비디오 부호화를 위한 적응적인 인-루프 필터 방법을 제안한다. 최근 비디오 부호화 표준화 단체에서는 영상의 부호화 후 복원된 영상과 원본 영상과의 평균 제곱 오차(mean square error) 관점에서 오차를 최소화하는 Wiener 필터기반의 post-filter hint SEI 메시지 방법과 블록 기반의 필터 제어 방법 (block-based adaptive filter control, BAFC)에 대한 연구가 있었다. Post-filter hint SEI 메시지 방법은 후처리 필터로서 프레임간의 예측 오차를 줄이지 못하는 문제점이 있으며, BAFC 방법은 기존 H.264/AVC의 디블록킹 필터와 독립적으로 동작하기 때문에 인코더 및 디코더 영역에서 높은 연산 복잡도를 차지하는 문제점이 있다. 본 논문에서는 기존 H.264/AVC의 디블록킹 필터와 문맥 기반으로 설계한 인-루프 필터를 적응적으로 사용함으로써 복잡도를 낮추고 부호화 효율을 높이는 인-루프 필터 방법(Low-complexity adaptive in-loop filter, LCALF)을 제안한다. 실험결과에서 제안하는 방법은 기존 방법보다 평균적으로 약 1% 정도의 비트 감소를 보이고, 동시에 디코더 영역에서 약 22% 정도의 낮은 연산 복잡도를 보인다.

#### Abstract

In this paper, we propose an adaptive in-loop filter to improve the coding efficiency. Recently, there are post-filter hint SEI and block-based adaptive filter control (BAFC) methods based on the Wiener filter which can minimize the mean square error between the input image and the decoded image in video coding standards. However, since the post-filter hint SEI is applied only to the output image, it cannot reduce the prediction errors of the subsequent frames. Because BAFC is also conducted with a deblocking filter, independently, it has a problem of high computational complexity on the encoder and decoder sides. In this paper, we propose the low-complexity adaptive in-loop filter (LCALF) which has lower computational complexity by using H.264/AVC deblocking filter, adaptively, as well as shows better performance than the conventional method. In the experimental results, the computational complexity of the proposed method is reduced about 22% than the conventional method. Furthermore, the coding efficiency of the proposed method is about 1% better than the BAFC.

Keyword: H.264/AVC, in-loop filter, Wiener filter

## 1. 서 론

비디오 부호화 표준과 관련하여 ISO/IEC SC29 WG11 (MPEG)과 ITU-T SG16/Q.6 (VCEG) 으로 구성된 Joint video team (JVT) 의 주도하에 H.264/MPEG-4 Part 10 Advance Video Coding (H.264/AVC) 표준이 성공적으로

a) 광운대학교 컴퓨터공학과

Dept. of Computer Engineering, Kwangwoon University

b) 삼성전자㈜ DMC연구소

Digital Media & Communication R&D Center, Samsung Electronics Co., Ltd.

‡ 교신저자 : 심동규 (dgsim@kw.ac.kr)

※ 이 논문은 삼성전자㈜의 지원을 통하여 이루어졌음.

· 접수일(2010년6월16일), 수정일(1차:2010년11월23일, 2차:2011년1월6일), 게재확정일(2011년1월21일)

이루어졌다<sup>[1]</sup>. H.264/AVC는 다수의 효율적인 부호화 알고리즘을 적용함으로써 고효율의 부호화 성능을 달성하였다. 그리고 최근에는 새로운 부호화 기술들을 바탕으로 MPEG과 VCEG으로 구성된 비디오 부호화 연합팀 (Joint Collaborative Team on Video Coding Standard Development, JCT)의 주도하에 High efficiency video coding (HEVC)라 명명된 차세대 비디오 부호화 표준화가 진행되고 있다<sup>[2][3]</sup>. 표준화를 위해 기고된 기술들 중에서 주목받는 것에는 Motion Vector Competition (MVC), Adaptive Interpolation Filter (AIF), Adaptive Block Partitioning (ABP), Mode-Dependent Directional Transform (MDDT), Adaptive Quantization Matrix Selection (AQMS), Post-filter hint SEI 메시지 (post-filter) 그리고 Block-based Adaptive Filter Control (BAFC) 등이 있으며, Key Technology Area (KTA) 참조 소프트웨어에 구현이 되어있다<sup>[4-10]</sup>. 여러 부호화 기술 중 BAFC 방법은 H.264/AVC의 더블록킹 필터가 적용된 복원 영상과 원본 영상과의 평균 제곱 오차(Mean square error)를 최소화하는 Wiener 필터를 루프 내에 적용함으로써 주관적 화질뿐만 아니라 객관적 화질을 향상시키며, 에러 전파를 방지하는 효과가 있다. 하지만 BAFC 방법은 더블록킹 필터와 독립적으로 동작하며, 더블록킹 필터와 같이 높은 연산량을 필요로 하기 때문에 디코더의 연산 복잡도 관점에서 높은 비중을 차지하는 문제점이 있다. 그리고 기존 H.264/AVC에서는 고화질의 비디오 부호화를 위해 후처리 필터에 대한 정보를 추가로 전송하여, 부호화 효율을 높이는 post-filter가 적용되어 있다. Post-filter 방법은 BAFC 방법과 유사하게 원본 영상과 복원 영상과의 평균 제곱 오차를 최소화하는 Wiener 필터를 사용하기 때문에 높은 효율로 부호화 할 수 있다. 하지만 BAFC 방법과 다르게 루프 내에서 필터를 적용하는 것이 아니라 출력 영상에 대해서만 필터링하기 때문에 프레임간의 예측 오차를 줄이지 못하는 문제점이 있다.

본 논문에서는 Wiener 필터를 이용하여, 부호화 효율을 높이면서 낮은 연산 복잡도를 가지는 적응적인 인-루프 필터 (Low-Complexity Adaptive in-Loop Filter, LCALF) 방법을 제안한다. 제안하는 방법은 부호화 루프 내에서 동작하기 때문에 부호화 오차뿐만 아니라 프레임 간의 예측 오

차를 줄일 수 있다. 또한, 제안하는 LCALF는 BAFC 방법과 다르게 기존 H.264/AVC의 더블록킹 필터와 결합된 형태로 동작하기 때문에 BAFC 방법과 비교하여 디코더 영역에서 더 낮은 연산 복잡도를 가지며, 문맥 기반의 다수의 필터를 적응적으로 적용함으로써 높은 효율로 부호화 한다.

논문의 구성은 다음과 같다. II장에서는 기존 post-filter와 BAFC 방법에 대해 분석하고 III장에서는 II장에서 분석한 내용을 바탕으로 본 논문에서 제안하는 알고리즘을 자세히 설명한다. IV장에서는 실험을 통해 제안하는 방법과 기존 방법과의 부호화 효율 및 연산 복잡도를 비교하며, 마지막으로 V장에서는 결론 및 향후 연구 방향을 기술한다.

## II. 기존의 후처리 및 인-루프 필터 알고리즘

기존 H.264/AVC의 더블록킹 필터는 부호화후 복원된 영상 내에서 블록 경계 간에 발생하는 오차를 줄임으로써, 프레임간의 에러 전파를 방지하여 주관적 화질과 객관적 화질을 모두 향상시킨다<sup>[11][12][13]</sup>. 하지만, 블록 경계에서만 필터가 적용되기 때문에 블록 내부에 남아있는 양자화 오차를 줄이지 못하는 문제점이 있다.

### 1. Post-filter hint SEI 메시지 방법

기존 H.264/AVC에는 고화질의 비디오 부호화를 위해 후처리 필터에 대한 정보를 추가로 전송하여, 부호화 효율을 높이는 post-filter가 적용되어 있다<sup>[8]</sup>. post-filter 방법은 원본 영상과 복원 영상과의 평균 제곱 오차를 최소화하는 Wiener 필터를 사용하여 복원된 영상의 화질을 향상시키는 효과가 있다. post-filter 방법에서는 최적의 Wiener 필터를 얻기 위해 복원된 영상신호 ( $s'$ )의 자기상관함수 ( $r_{s's'}$ )와 복원된 영상신호 ( $s'$ )와 원본 영상신호 ( $s$ )와의 상호상관함수 ( $r_{s's}$ )가 계산된다. 식 (1)은 자기상관함수 ( $r_{s's'}$ )를 식 (2)는 상호상관함수 ( $r_{s's}$ )를 나타낸다.

$$r_{s's'}(k, l) = \frac{1}{w \cdot h} \sum_m \sum_n^{w-h-1} s'(m, n) \cdot s'(m+k, n+l) \quad (1)$$

$$r_{s's}(k,l) = \frac{1}{w \cdot h} \sum_m^{w-1} \sum_n^{h-1} s'(m,n) \cdot s(m+k,n+l) \quad (2)$$

위 두 함수에 대한 식으로부터 자기상관행렬 ( $R_{s's'}$ ) 과 상호상관행렬 ( $R_{s's}$ ) 이 각각 유도되며, 최적의 Wiener 필터 계수  $c$ 는 식 (3)과 같이 계산된다.

$$c = R_{s's'}^{-1} \cdot R_{s's} \quad (3)$$

이와 같이 계산된 최적의 Wiener 필터 계수는 'Post-filter Hint'라 불리는 SEI 메시지 형태로 디코더에 전송된다. 그림 1은 post-filter의 인코더 블록도를 나타낸 것이며, 인코더에서 계산된 필터 정보가 디코더로 전송되는 것을 확인할 수 있다.

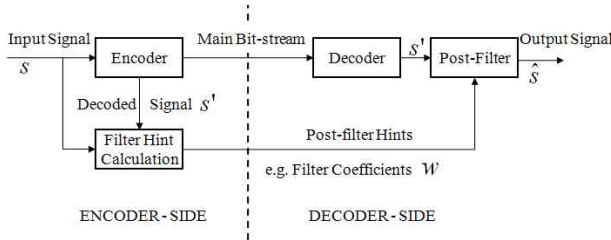


그림 1. post-filter 인코더의 시스템 블록도  
Fig. 1. Block diagram of the post-filter encoder.

최적의 Wiener 필터에 대한 부가적인 정보를 전송함으로써 복원된 영상의 화질을 향상시키는 post-filter 방법은 부호화 루프 내에서 필터를 적용하는 것이 아니라 출력 영상에 대해서만 필터링하기 때문에 프레임간의 예측 오차를 줄이지 못하는 문제점이 있다.

## 2. BAFC 방법

BAFC 방법은 루프필터로서 더블클릭 필터 결과에 다시 Wiener 필터를 적용하여 프레임간의 예측 오차뿐만 아니라 부호화 오차를 줄이는 효과가 있다<sup>[10]</sup>. BAFC 방법에서는 최적의 Wiener 필터를 얻기 위해 다음 식 (4)에 나타난 목적함수 ( $E$ ) 의 오차를 최소화하는 계수 ( $h$ ) 를 계산한다.

$$E = \sum_x \sum_y \left( I_{x,y} - \sum_i \sum_j h_{i,j} R_{x+i,y+j} \right)^2 \quad (4)$$

식에서  $I_{x,y}$  와  $R_{x,y}$  는 각각 입력 영상과 복원된 영상에서  $(x,y)$  에 위치한 화소 값을 나타낸다. 그리고  $h_{i,j}$  는 이웃한 위치,  $(x+i,y+j)$  에서의 필터 계수를 나타내며, 이는 식 (5)에 나타난 연립 방정식의 해를 구하는 것으로 계산된다. 즉,  $h_{i,j}$  는 post-filter 방법에서와 같이 입력 영상과 복원된 영상의 평균 제곱 오차를 최소화하는 필터 계수가 된다.

$$\frac{\partial E}{\partial h_{k,l}} = 0 \quad (5)$$

$$\sum_x \sum_y \left( I_{x,y} - \sum_i \sum_j h_{i,j} R_{x+i,y+j} \right) R_{x+k,y+l} = 0$$

하지만, 평균 제곱 오차를 최소화하는 Wiener 필터를 복원된 영상에 적용하였을 때, 오차가 줄어드는 부분 이외에도 오히려 오차가 증가하는 부분이 있을 수 있으며, 이는 필터링된 영상이 참조 프레임으로 사용될 때 프레임간에 오차를 전파하는 문제의 원인이 될 수 있다. 이러한 이유 때문에 BAFC 방법에서는 블록단위로 필터적용 여부를 다르게 결정하고 그에 대한 추가정보를 전송하여 프레임간의 오차전파를 방지하고 부호화 효율을 높인다. 또한, BAFC 방법에서는 블록의 크기와 필터 플래그와 같은 필터 정보의 양을 최적으로 조정하기 위해 식 (6)과 같은 비트율-왜곡 최적화기법 (Rate distortion optimization)을 적용한다.

$$J = D + \lambda R \quad (6)$$

$D$  는 입력 영상과 필터링된 영상간의 왜곡 값이고  $R$  은 필터 처리를 위해 발생된 비트의 양이다. 그리고  $\lambda$  는 양자화 파라미터 값에 따라 결정되는 라그랑지안 파라미터 값이다. 그림 2는 BAFC 방법의 인코더 블록도를 나타낸 것이다. 'In-loop Filter with BAFC' 블록을 제외하면 기존 H.264/AVC의 인코더 블록도와 동일한 구조를 갖는다.

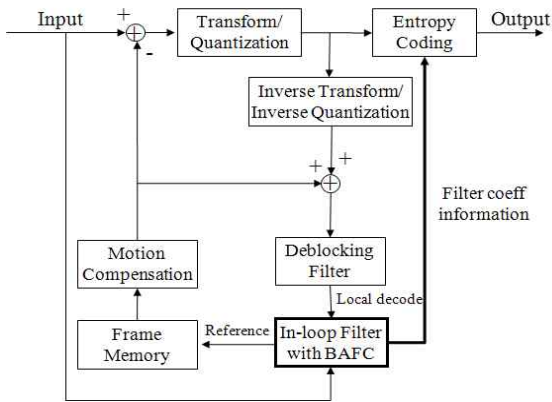


그림 2. BAFC 인코더의 시스템 블록도  
Fig 2. Block diagram of the BAFC encoder

위 그림에서와 같이 BAFC 방법은 기존 H.264/AVC의 디블록킹 필터가 적용된 결과에 다시 필터링을 수행한다. 즉, BAFC 방법은 기존 디블록킹 필터와 독립적으로 수행되기 때문에 디코더 영역에서 높은 연산 복잡도를 갖는 문제점이 있다. 또한, BAFC 방법은 슬라이스마다 하나의 필터를 적용하기 때문에 부호화 오차 각각에 대한 특성을 반영하지 못하는 문제점이 있다. 즉, 부호화 오차마다의 특성에 최적화된 필터를 각각 설계하고 적용한다면 부호화 효율을 향상시킬 수 있을 것이다.

이러한 점에 착안하여 본 논문에서는 기존 H.264/AVC의 디블록킹 필터와 Wiener 필터를 적응적으로 선택하여 디코더 영역의 연산 복잡도를 낮추고, 부호화 문맥정보를 기반으로 다수의 1-D 필터를 설계하여 부호화 효율을 높이는 적응적인 인-루프 필터 방법을 제안한다.

### III. 제안하는 적응적인 인-루프 필터

본 논문에서는 매크로블록 단위의 적응적인 인-루프 필터 방법을 제안한다. 기존 필터링 방법은 H.264/AVC 디블록킹 필터와 독립적으로 동작하고 필터 자체의 연산량이 크기 때문에 디코더 영역에서 필터 부분이 차지하는 연산 비중이 전체의 약 60% 정도에 달한다. 또한, 기존 방법에서는 슬라이스마다 슬라이스 전체에 최적화된 필터를 하나씩

만 적용하고 있기 때문에 슬라이스 내의 모든 특성을 반영하지 못하는 문제점이 있다. 제안하는 알고리즘은 부호화 효율을 높이기 위하여 매크로블록 정보, 부호화된 블록패턴, 움직임 벡터, 정수변환 크기 등과 같은 부호화 문맥정보를 기반으로 부호화 오차와 영상의 특성에 맞게 다수의 필터 설계를 제안하고, 인-루프 필터 부분에 대한 연산 복잡도를 낮추기 위하여 기존 H.264/AVC 디블록킹 필터와 문맥 기반의 인-루프 필터를 적응적으로 선택한다. 제안하는 방법은 기존 방법과 비교하여 디코더 영역에서 더 낮은 연산 복잡도를 보이며, 부호화 실험에서 전체적으로 더 높은 압축 성능을 나타낸다.

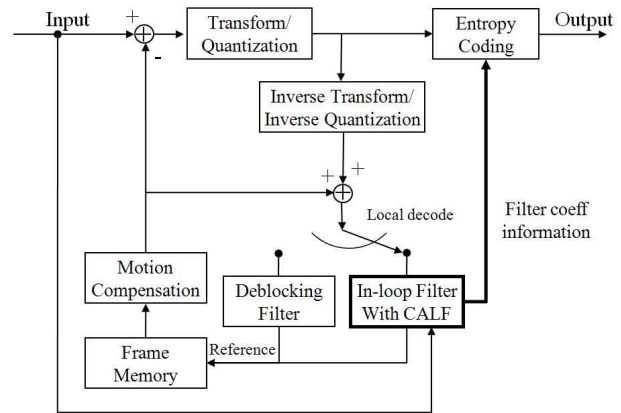


그림 3. 제안하는 LCALF 인코더의 시스템 블록도  
Fig 3. Block diagram of the proposed LCALF encoder

제안하는 저복잡도 고효율의 적응적인 인-루프 필터 (LCALF) 방법의 인코더 시스템 블록도는 그림 3과 같다. 제안하는 방법은 복호화된 영상에 대해서 기존의 H.264/AVC 디블록킹 필터를 사용하는지, 문맥 기반의 적응적 루프 필터(Context-based adaptive in-loop filter: CALF)를 사용하는지 매크로블록 단위로 선택한다. 본 장에서는 CALF를 위한 필터 설계 및 필터 동작 원리는 설명한다.

#### 1. 필터 설계 방법

기존 BAFC 방법은 복원된 영상에 기존 H.264/AVC 디블록킹 필터를 적용하고 그 결과에 다시 블록 기반의 적응적인

인-루프 필터를 적용하기 때문에 디코더 영역에서 연산 복잡도 비중이 높다는 문제점과 슬라이스마다 최적화된 필터를 하나씩만 적용하기 때문에 슬라이스 내에서 발생한 다양한 부호화 오차 특성을 모두 고려하지 못하는 문제점이 있다. 이러한 문제점에 착안하여 제안하는 방법에서는 연산 복잡도를 낮추기 위해 기존 H.264/AVC 디블록킹 필터와 문맥 기반의 인-루프 필터를 적응적으로 사용하는 필터 구조를 사용하며, 슬라이스마다 발생하는 부호화 오차 특성을 고려하기 위해 부호화 문맥정보를 기반으로 다수의 필터를 설계한다. 제안하는 방법의 필터 설계 구조는 그림 4와 같다.

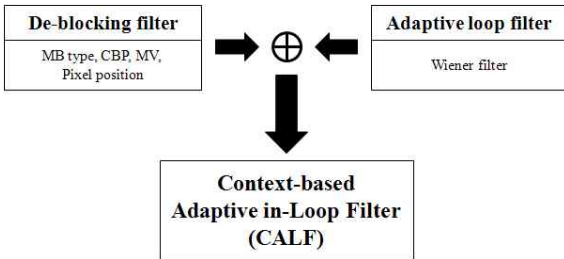


그림 4. 제안하는 필터 설계 방법  
Fig. 4. The proposed filter design method

위 그림과 같이 제안하는 방법에서는 매크로블록 정보, 부호화된 블록패턴, 움직임 벡터, 정수변환 크기 등과 같은 부호화 문맥정보를 이용하여 복원된 영상에서 발생하는 부호화 오차 및 블록킹 오차의 특성을 분류하고 분류된 영역들에 대해서 각각 Wiener filter를 기반으로 다수의 필터를 설계한다. 이는 제안하는 CALF 방법이 사용되는 영역에 대해서는 디블록킹 필터가 적용되지 않기 때문에 CALF의 필터 설계 시에 디블록킹 필터에 대한 고려가 필요하다. CALF 방법에서는 최적의 필터를 얻기 위해 다음 식 (7)에 나타난 목적함수 ( $E_{ver}$ ,  $E_{hor}$ )의 오차를 최소화하는 계수 ( $h, v$ )를 계산한다.

$$\begin{aligned}
 E_{hor} &= \sum_x \sum_y \left( I_{x,y} - \sum_i h_i R_{x+i,y} \right)^2 \\
 E_{ver} &= \sum_x \sum_y \left( I_{x,y} - \sum_j v_j R_{x,y+j} \right)^2
 \end{aligned} \tag{7}$$

위 수식에서  $I_{x,y}$ 와  $R_{x,y}$ 는 각각 입력 영상과 복원된 영상에서 화소 값을 나타낸다. 그리고  $h_i$ 와  $v_j$ 는 각각 이웃한

위치  $(x+i, y)$ 와  $(x, y+j)$ 에서의 필터 계수를 나타내며, 이는 식 (8)에 나타난 연립 방정식의 해를 구하는 것으로 각각 계산된다.  $h_i$ 와  $v_j$ 는 각각 입력 영상과 복원된 영상의 평균제곱오차를 최소화하는 수평 방향 그리고 수직 방향의 필터 계수가 된다.

$$\begin{aligned}
 \frac{\partial E}{\partial h_k} &= 0, \sum_x \sum_y \left( I_{x,y} - \sum_i h_i R_{x+i,y} \right) R_{x+k,y} = 0 \\
 \frac{\partial E}{\partial v_l} &= 0, \sum_x \sum_y \left( I_{x,y} - \sum_j v_j R_{x,y+j} \right) R_{x,y+l} = 0
 \end{aligned} \tag{8}$$

표 1은 CALF에 대한 필터 설계 내용을 나타낸 것이다. CALF는 크게 수평 방향 필터와 수직 방향 필터 집합으로 구성되며 서로 독립적으로 동작한다. 두개의 필터 집합은 매크로블록 정보, 부호화된 블록패턴, 움직임 벡터, 정수변환 크기 등과 같은 문맥정보가 고려된 경계 세기 (Boundary strength, BS) 값과 화소의 위치 정보를 기반으로 설계된 세개의 필터로 구성된다. 필터-(1)은 BS 값이 (0, 1)이고, 화소 위치는 (p0, q0, p1, q1)이다. 필터-(2)는 BS 값이 (2, 3, 4)이고, 화소 위치는 (p0, q0)이며, 필터-(3)은 같은 BS 조건에서 (p1, q1) 위치의 화소에 적용된다. 각 필터는 프레임 기반의 RDO를 통하여 필터의 설계 유무가 결정될 수 있다. 즉, 항상 6개의 필터가 생성되는 것이 아니고, 일부의 필터만이 사용될 수 있다. 블록 단위의 on/off는 디블록킹 필터와 제안하는 문맥 기반 필터와의 RDO를 통하여 결정된다. 필터 단위에서 off인 필터링 위치에서 대해서는 블록 단위의 필터가 on이 되어도 필터링이 수행되지 않는다.

표 1. 제안하는 방법의 필터 설계 내용  
Table 1. Filter design description of the proposed method

블록 경계 세기 (BS)	문맥 기반 적응적 루프 필터 (CALF)				필터링 과정		
	화소 위치	필터 정보					
		수평 필터	수직 필터				
0, 1	(p0, q0) (p1, q1)	Filter-(1)	Filter-(1)'	필터 단위 on / off	블록 단위 on / off		
2, 3, 4	(p0, q0) (p1, q1)	Filter-(2) Filter-(3)	Filter-(2)' Filter-(3)'				

BS 값은 H.264/AVC 디블록킹 필터에서 필터링 강도를

나타내는 파라미터로서 0부터 4까지의 범위를 가지며 값이 클수록 필터링 강도가 높다. 그림 5는 HD급 영상 여섯 개 사용하여 네 개(22, 27, 32, 37)의 QP에 대한 평균 BS 발생 빈도를 보여준다. 제안하는 방법에서는 필터 설계를 위해 BS 값을 (0, 1) 그리고 (2, 3, 4) 두 그룹으로 분류하였으며, 이는 필터링 강도(약한 필터와 강한 필터)와 BS 값의 발생 빈도를 고려한 것이다. 즉, 상대적으로 약한 필터에 해당하는 BS(1, 2)와 필터를 취하지 않는 BS(0)을 하나의 그룹으

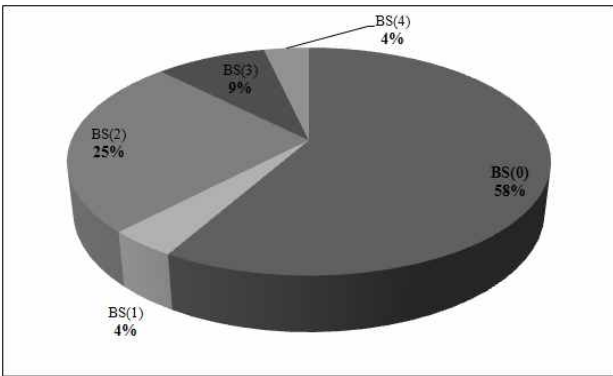


그림 5. BS 값의 발생 빈도  
Fig 5. Statistics of BS value

로 묶을 경우 발생 빈도가 비대칭적으로 분류될 수 있기 때문에 제안하는 방법과 같이 BS 값을 분류하였다.

제안하는 CALF 필터는 BS 값이 (0, 1)인 경계에서 화소의 위치에 상관없이 필터링 강도가 약한 필터를 설계하고, BS 값이 (2, 3, 4)인 경계에서 블록 경계를 기준으로 첫 번째 화소와 두 번째 화소에 대해 필터링 강도가 서로 다른 필터를 설계한다. 제안하는 필터는 모두 계수 값이 대칭 형태인 1-D 필터이며, 필터 탭 크기는 (5, 7, 9), (7, 9, 11), (9, 11, 13) 그룹으로 나누어 설계하여 각 그룹에 포함된 세 개의 탭 중 최적의 필터 탭을 결정한다. 예를 들어, (5, 7, 9) 그룹에서 필터 탭을 결정하는 경우, 5, 7, 9 탭 크기의 필터를 모두 설계한 후, 이를 적용하여 비트율-왜곡 관점에서 최적인 필터 탭을 결정하게 된다. 필터 탭 크기를 세 개씩 그룹화한 것은 기존의 BAFC 방법과 공평한 복잡도를 유지하기 위함이다. 보다 정교한 필터 설계를 위해 기존 BAFC 방법에서와 같이 필터링 효과가 있는 부분을 이용하여 세 단계의 필터 재설계 과정을 수행하였다.

## 2. 필터 적용 방법

제안한 CALF 방법에서는 슬라이스 단위로 수평 방향과

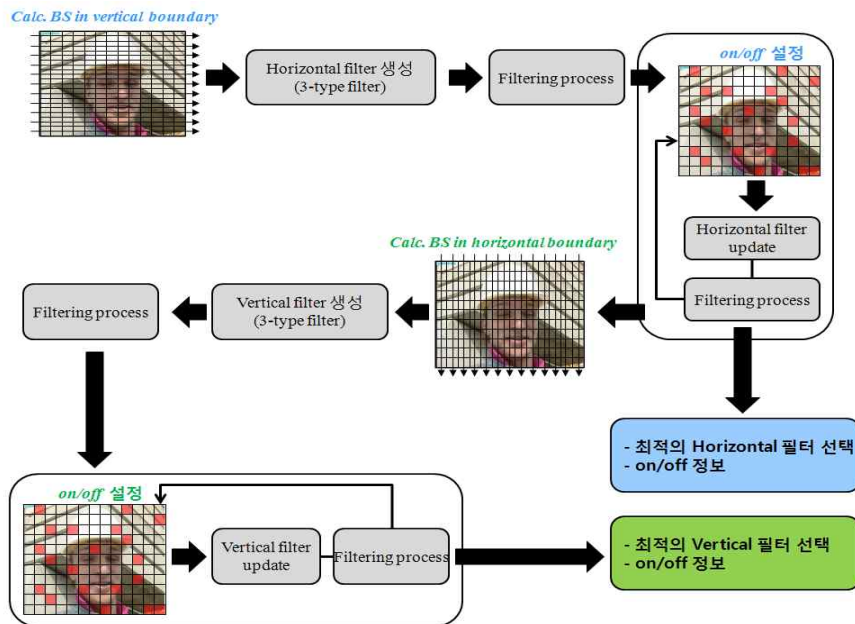


그림 6. CALF의 동작 순서도  
Fig. 6. Flow chart of the CALF



수직 방향, 두 단계로 나누어 필터링을 진행한다. 두 단계의 필터는 서로 독립적으로 동작하며 자세한 필터링 순서는 그림 6과 같다.

먼저, 영상의 수직 방향 경계에 대한 오차를 최소화하기 위해 수평 방향 필터를 설계 및 적용한다. 수평 방향 필터를 적용한 후 오차가 줄어든 부분 이외에도 반대로 오차가 증가한 부분이 있을 수 있기 때문에 부호화 오차 전파를 방지하기 위하여 블록단위로 필터적용 여부를 결정한다. CALF 방법에서는 필터적용 여부를 결정하기 위해 식 (9)를 이용한다.

$$f_n = \begin{cases} 1 & \left( \sum_{x,y \in B_n} \{F_{x,y} - I_{x,y}\}^2 < \sum_{x,y \in B_n} \{R_{x,y} - I_{x,y}\}^2 \right) \\ 0 & \left( \sum_{x,y \in B_n} \{F_{x,y} - I_{x,y}\}^2 \geq \sum_{x,y \in B_n} \{R_{x,y} - I_{x,y}\}^2 \right) \end{cases} \quad (9)$$

$f_n$ 은 n번째 블록( $B_n$ )에 대한 필터 적용 여부를 나타내는 플래그를 의미한다. 여기서  $F_{x,y}$ 은 필터가 적용된 영상이고,  $I_{x,y}$ 는 입력 영상,  $R_{x,y}$ 는 복원된 영상이다. 또한, CALF 방법에서는 블록의 크기와 필터 플래그와 같은 필터 정보의 양을 최적으로 조정하기 위해 BAFC 방법에서와 같이 식 (6)과 같이 비트율-왜곡 최적화 기법을 적용한다.

수평 방향 필터에 의해 필터 적용 여부가 결정되면, 필터 정교화를 위해 필터가 적용된 블록들을 이용하여 수평 방향 필터를 재설계한다. 그런 후 다시 생성된 수평 방향 필터를 이용하여 복원된 영상에 필터링을 한 후 블록 단위로

필터 적용 여부를 다시 결정한다. 이와 같은 필터 재설계 과정을 거쳐 최종적인 수평 방향 필터와 블록 단위 필터 적용 여부 정보가 결정된다. 그 다음 CALF는 수평 방향 필터가 적용된 영상에 대해 수평 방향 경계에 대한 오차를 최소화하기 위한 수직 방향 필터를 설계 및 적용한다. 수직 방향 필터의 설계 및 적용에 대한 세부 과정은 수평 방향 필터와 동일하게 이루어진다. 이와 같이 제안하는 CALF 방법에서는 슬라이스 단위로 최대 6개의 적응적인 필터가 사용될 수 있다. 또한, 제안하는 방법에서는 부호화 효율을 높이고 복잡도를 낮추기 위해 블록 단위 필터 적용 여부 결정이외에도 필터 단위의 필터 사용 유무를 결정할 수도 있다. 블록 단위의 on/off 정보는 기존의 BAFC 방법과 유사하게 8x8, 16x16, 32x32, 64x64 블록 중에 RDO 관점에서 최적인 필터 적용 블록 크기를 결정하게 된다. 이진 값인 on/off 정보는 CABAC 를 이용하여 압축하고 이를 디코더로 전송한다.

기존의 방법에서는 H.264/AVC의 디블록킹 필터를 수행한 이후에 독립적으로 BAFC 필터를 적용하였다. 제안하는 방법에서는 문맥정보를 기반으로 다수의 고효율 필터를 설계를 위해 디블록킹 필터의 필터링 강도 파라미터인 BS 값을 이용하였다. 그림 7은 LCALF 시스템에서 CALF 방법과 기존 디블록킹 필터를 블록 단위로 RDO하여 최적의 필터를 선택하는 과정을 보여준다.

이와 같이 문맥정보 기반의 다수의 필터와 기존 디블록킹 필터를 적응적으로 사용하는 방법은 실험결과에서 기존

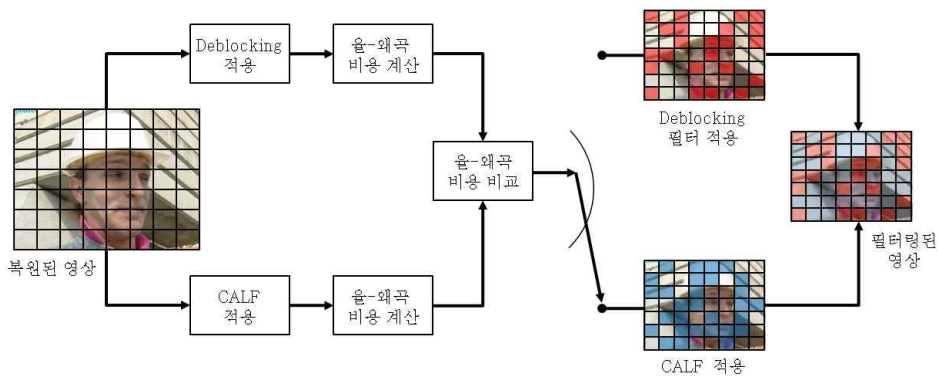


그림 7. LCALF 시스템의 블록단위 RDO 구조  
Fig. 7. RDO structure of LCALF system

BAFC 방법과 비교하여 더 낮은 연산 복잡도를 보이는 동시에 상대적으로 더 높은 부호화 효율을 나타낸다.

#### IV. 실험 결과 및 토의

본 논문에서 제안하는 저복잡도의 적응적인 인-루프 필터(LCALF)의 부호화 효율과 연산 복잡도 감소를 측정하기 위해 표준화 단체에서 권고하는 조건에 맞추어 실험을 수행하였으며, 제안한 방법이 기존 BAFC 방법보다 부호화 효율과 연산 복잡도 관점에서 더 좋은 성능을 보임을 확인하였다<sup>[14]</sup>.

표 2. 부호화 성능 측정을 위한 실험 조건  
Table 2. Test condition for coding performance evaluation

Test conditions	Description
Profile	High
GOP Structure	IPPP
Frame nums.	50 frames
Rate-Distortion Optimization	ON
8×8 Transform	ON
Entropy Coding Method	CABAC
Sequences	720p60 (1280×720) (BigShips, Crew, Jets, Raven)
Quantization Parameters	QPISlice = 22, 27, 32, 37 QPPSlice = 23, 28, 33, 38

본 논문에서 수행한 실험 조건은 표 3과 같고, 부호화 실험에 대한 성능 평가를 위해 Bjontegaard Delta Peak Signal-to-Noise Ratio(BD\_PSNR)<sup>[15]</sup> 모델을 사용하였다. BD\_PSNR 모델은 비디오 부호화 표준 단체에서 채택하고 있으며 비디오 압축 분야에서 사용되고 있는 부호화 성능 측정도구이다. 제안한 방법은 KTA 참조 소프트웨어 2.2 버전을 이용하여 구현하였으며, KTA 기술인 BAFC 방법의 부호화기와 비교하였다<sup>[16]</sup>.

제안하는 방법은 슬라이스마다 최적의 필터 탭 크기를 선택하여 적용한다. 필터 탭 크기가 클수록 필터 성능은 좋지만, 필터 계수에 대한 부호화 비트가 증가하기 때문에 부호화 오차와 발생 비트를 모두 고려하여 최적의 필터를 선택해야 한다. 제안한 방법에 대한 성능 실험에서는 (5, 7, 9), (7, 9, 11) 그리고 (9, 11, 13)과 같이 필터 집합을 세 그룹으로 나누어 기존 방법과 성능을 비교하였다. 표 3은 제안한 LCALF와 기존 방법인 BAFC 방법에 대해 전체부호화 비트 중 필터에 대한 추가전송 비트의 비중을 비교한 것이다.

BAFC 방법의 경우에는 필터에 대한 추가전송 비트의 비중이 전체부호화 비트의 2.5% 정도를 차지하며, 그 중 필터 계수 정보가 0.6% 이고 필터적용 여부에 대한 정보가 1.9% 임을 알 수 있다. 반면, 제안하는 CALF에 대한 추가전송 비트의 전체 비중은 기존 방법보다 작게는 1.5배, 크게는 2배 정도 더 높게 나타난다. 구체적으로 필터계수 정보와 필터적용 여부에 대한 정보 비교에서도 제안한 방법이 작게는 1.5배, 크게는 2배 정도 더 높음을 확인할 수 있다. 제

표 3. BAFC와 LCALF에 대한 추가전송 비트율 비교  
Table 3. Additional bitrate comparison between BAFC and LCALF

	기존의 적응적 보간 필터 (BAFC)			제안하는 LCALF (5, 7, 9 탭)			제안하는 LCALF (7, 9, 11 탭)			제안하는 LCALF (9, 11, 13 탭)		
	Coeff.	on/off	Total	Coeff.	on/off	Total	Coeff.	on/off	Total	Coeff.	on/off	Total
Bigships	0.54	1.59	2.12	0.91	2.74	3.66	1.05	2.9	3.94	1.2	2.9	4.1
Crew	0.54	2.16	2.7	0.81	3.01	3.82	0.95	3.14	4.09	1.09	3.25	4.34
Jets	0.8	2.52	3.32	1.21	5.23	6.45	1.38	5.55	6.93	1.49	5.66	7.15
Raven	0.7	1.91	2.61	1.07	3.59	4.66	1.19	3.46	4.65	1.33	3.53	4.85
평균(%)	0.62	1.87	2.49	0.95	3.35	4.30	1.08	3.47	4.55	1.20	3.54	4.74



안한 LCALF에 대한 세 가지 실험만을 비교해보면, 필터 그룹의 탭 크기가 증가할수록 필터계수 정보와 필터적용 여부에 대한 정보의 비중이 모두 증가하는 것을 알 수 있다. 필터계수 정보에 대한 비트 증가는 필터 탭 크기가 증가하기 때문이며, 필터적용 여부에 대한 비트 증가는 필터 탭 크기가 증가할수록 필터 성능이 좋아짐으로써 필터적용 슬라이스와 블록단위 필터링을 수행하는 슬라이스가 증가하기 때문이다.

표 4는 제안하는 LCALF에 대한 성능을 보여준다. 기존 BAFC 방법은 KTA 2.2 소프트웨어 대비 BD\_Bitrate 관점에서 약 13.66%의 비트 감소를 보이며, 제안하는 LCALF는 최소 14.44%에서 최대 14.53% 정도의 비트 감소를 보인다. 즉, 제안하는 LCALF가 기존 BAFC 방법보다 약 1%

정도의 비트 감소 효과가 있음을 알 수 있다.

표 3에서와 같이 제안한 방법이 기존 방법보다 필터에 대한 추가전송 비트의 비중이 더 높은 점을 고려한다면 LCALF의 필터 성능이 기존 BAFC 방법보다 더 좋음을 간접적으로 알 수 있다. 표 5는 추가정보를 제외했을 때 제안한 방법의 부호화 성능을 비교하기 위하여, 제안한 방법의 부호화 성능 결과 중 필터 탭 그룹이 (5, 7, 9)인 방법과 기존 BAFC 방법에 대해서 추가비트를 전송했을 때와 전송하지 않았을 때의 부호화 성능을 비교한 것이다. 결과를 보면, 필터에 대한 추가정보를 전송했을 때는 제안한 방법이 기존 방법보다 약 0.9% 정도 더 높은 부호화 성능을 보이고, 필터에 대한 추가정보를 전송하지 않았을 때는 제안한 방법이 기존 방법보다 약 2% 정도 더 높은 비트 감소율을 보

표 4. 제안된 방법의 부호화 성능  
Table 4. Performance of the proposed method

	기존의 적응적 보간 필터 (BAFC)		제안하는 LCALF (5, 7, 9 탭)		제안하는 LCALF (7, 9, 11 탭)		제안하는 LCALF (9, 11, 13 탭)	
	BD_PSNR (dB)	BD_Bitrate (%)	BD_PSNR (dB)	BD_Bitrate (%)	BD_PSNR (dB)	BD_Bitrate (%)	BD_PSNR (dB)	BD_Bitrate (%)
Bigships	0.27	-9.07	0.27	-9.08	0.27	-9.09	0.27	-9.15
Crew	0.46	-16.64	0.47	-17	0.47	-17.06	0.48	-17.47
Jets	0.34	-8.01	0.43	-10.22	0.45	-10.46	0.44	-10.27
Raven	0.88	-20.9	0.91	-21.45	0.92	-21.36	0.91	-21.23
평균(%)	0.49	-13.66	0.52	-14.44	0.53	-14.49	0.53	-14.53

표 5. 추가전송 비트에 따른 제안한 방법의 부호화 성능  
Table 5. Performance of the proposed method by additional information

	필터에 대한 추가비트 전송				필터에 대한 추가비트 비전송			
	기존의 적응적 보간 필터 (BAFC)		제안하는 LCALF (5, 7, 9 탭)		기존의 적응적 보간 필터 (BAFC)		제안하는 LCALF (5, 7, 9 탭)	
	BD_PSNR (dB)	BD_Bitrate (%)	BD_PSNR (dB)	BD_Bitrate (%)	BD_PSNR (dB)	BD_Bitrate (%)	BD_PSNR (dB)	BD_Bitrate (%)
Bigships	0.27	-9.07	0.27	-9.09	0.34	-11.14	0.37	-12.07
Crew	0.46	-16.64	0.47	-17	0.53	-19.07	0.56	-19.83
Jets	0.34	-8.01	0.43	-10.22	0.44	-10.79	0.64	-15.18
Raven	0.88	-20.9	0.91	-21.45	0.98	-22.92	1.09	-25.01
평균(%)	0.49	-13.66	0.52	-14.44	0.57	-15.98	0.67	-18.02

인다.

제안한 LCALF와 기존 BAFC 방법과의 디코더 연산 복잡도 비교를 위해, 부호화 성능 실험에서와 같이 KTA 소프트웨어를 이용하였다. 연산 복잡도는 디코더 전체에 대한 것과 인-루프 필터링에 대한 것으로 나누어 측정하였다. 연산 복잡도 측정을 위해서 다음 수식 (11)을 이용하였다.

$$\text{Average Time Saving}(\%) = \frac{BAFC_t - LCALF_t}{BAFC_t} \times 100 \quad (11)$$

식에서  $BAFC_t$ 는 기존 방법에 대한 디코더 및 필터링의 평균 시간을 나타내고,  $LCALF_t$ 는 제안하는 방법에 대한 디코더 및 필터링의 평균 시간을 나타낸다. 즉, 수식의 결과 (ATS)는 기존 방법 대비 제안된 방법의 평균적인 필터 수행 시간에 대한 감소율을 나타낸다. ATS를 측정하기 위해 Core2 Q6600 2.4GHz의 CPU, 2GHz 메모리, window XP 32비트OS 환경에서 실험을 진행하였다. 연산 복잡도는 각 영상마다 비트스트림을 생성하고, 복호화기를 통해 복호화 하면서 소모되는 평균시간을 측정한 것이다. 그리고 오차를 줄이기 위해 각 영상마다 약 100번씩 실험하였으며, 전체에 대한 평균값을 계산하여 최종 측정값으로 적용하였다. 실험 결과는 표 6과 같다.

먼저, 인-루프 필터 부분에 대한 결과를 보면 제안하는 LCALF가 기존 BAFC 방법보다 평균적으로 약 35.37%에서 37.6% 정도까지 연산 복잡도 감소가 있음을 확인할 수

있다. 또한, 디코더 전체에 대한 결과에서도 제안한 LCALF가 기존 방법보다 평균적으로 약 21.68%에서 23.15% 정도까지 연산 복잡도 감소를 나타낸다. 표 6의 결과에서 제안한 방법의 필터 탭 크기가 증가할수록 연산 복잡도가 상대적으로 증가함을 확인할 수 있고, 블록단위 필터적용 여부에 대한 추가전송 비트가 작은 Bigships과 Crew 영상의 복호시간 개선이 상대적으로 더 많음을 알 수 있다. Bigships과 Crew의 영상의 경우 블록단위 필터링보다는 상대적으로 슬라이스 전체에 대한 필터링이 상대적으로 더 많기 때문이다. 반대로 블록단위 필터적용 여부에 대한 추가전송 비트가 가장 많은 Jets 영상의 경우 상대적으로 복호시간 개선이 작게 나타남을 확인할 수 있다. 이와 같이 본 논문에서 제안하는 방법은 부호화 문맥정보를 기반으로 다수의 적응적인 필터를 적용함으로써 기존 방법보다 더 높은 부호화 효율을 보이는 동시에 1-D 구조의 필터집합과 기존 디블록킹 필터와의 결합 구조를 통해 기존 BAFC 방법보다 더 낮은 연산 복잡도를 보인다.

## V. 결론

본 논문에서는 낮은 복잡도를 가지면서 고효율의 성능을 보이는 LCALF 방법을 제안하였다. 제안한 방법은 부호화 성능을 높이기 위해 부호화 문맥정보를 기반으로 다수의 적응적인 필터를 설계 및 적용하였으며, 디코더 연산 복잡

표 6. 제안된 방법의 복호화 시간 개선  
Table 6. Decoding time saving of the proposed method

	인-루프 필터에 대한 측정시간 (ATS)			디코더 전체에 대한 측정시간 (ATS)		
	LCALF (5, 7, 9)	LCALF (7, 9, 11)	LCALF (9, 11, 13)	LCALF (5, 7, 9)	LCALF (7, 9, 11)	LCALF (9, 11, 13)
Bigships	39.1	36.97	36.97	22.87	21.58	21.72
Crew	47.1	45.62	45.13	30.61	29.39	29.19
Jets	30.12	27.37	27.83	17.31	15.78	15.78
Raven	34.11	32.32	31.58	21.83	20.4	20.05
평균(%)	37.6075	35.57	35.3775	23.155	21.7875	21.685

도 감소를 위해 기존 디블록킹 필터와 적응적인 선택을 한다. 실험 결과, 제안한 LCALF 방법은 기존 BAFC 방법보다 약 1% 정도 비트 감소율을 보이며, 디코더 전체에 대한 복잡도 관점에서 약 22%의 연산 복잡도 감소를 보였다.

향후에는 제안한 방법의 부호화 성능 향상을 위해 QALF(quadtree-based adaptive loop filter) 방법을 기반으로 필터링의 on/off 정보를 줄이거나 제안하는 방법의 추가적인 필터 계수 정보를 줄이는 연구를 수행할 수 있다. 또한, 경계 세기 값 이외에 추가적으로 이용할 수 있는 부호화 문맥 정보를 이용하여 정밀도가 높은 알고리즘을 개발해야 할 것이다.

### 참 고 문 헌

- [1] Joint Video Team (JVT) of ITU-T VCEG and ISO/IEC MPEG, "Draft ITU-T Recommendation H.264 and ISO/IEC 14496-10 AVC," May 2003.
- [2] ISO/IEC JTC1/SC29/WG11 and ITU-T SG16, "Joint Call for Proposals on Video Compression Technology," N1113&VCEG-AM91, Jan. 2010.
- [3] 정세운, 김휘용, 임성창, 김종호, 이하현, 이진호, 최해철, 조숙희, 최진수, "JCT-VC의 HVC 표준화 동향," 전자통신동향분석, 25권, 8-18쪽, 2010년 4월.
- [4] J. Jung, G. Laroche, "Competition-Based Scheme for Motion Vector Selection and Coding," ITU-T SG16/Q6 Document VCEG-AC06, Klagenfurt, July 2006.
- [5] O. Divorra, P. Yin, and C. Gomila, "Geometry adaptive Block Partitioning on B-frames," ITU-T SG16/Q6, VCEG-AG13, Oct. 2007.
- [6] Y. Ye, M. Karczewicz, "Improved Intra Coding," ITU-T SG16/Q6 Document VCEG-AG11, Shenzhen China, October 2007.
- [7] A. Tanizawa, T. Chujoh, "Adaptive Quantization Matrix Selection," ITU-T SG16/Q6 Document D-266, Geneva, Switzerland, Nov. 2006.
- [8] M. Karczewicz, W. Chien, P. Chen, and Y. Ye, "Post-Filter SEI Message Extensions," ITUT SG16/Q.6 Doc. VCEG-AI34, Berlin, Germany, July 2008.
- [9] T. Chujoh, G. Yasuda, N. Wada, T. Watanabe and T. Yamakage, "Block-based Adaptive Loop Filter," ITU-T SG16/Q.6 Document VCEG-AI18, Berlin, Germany, July 2008.
- [10] T. Watanabe, N. Wada, G. Yasuda, A. Tanizawa, T. Chujoh, and T. Yamakage, "In-loop filter using block-based filter control for video coding," ITU-T SG16/Q.6 Document VCEG-AI18, Berlin, Germany, July 2008.
- [11] T. Wiegand, G. J. Sullivan, G. Bjontegaard, A. Luthra, "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560-576, Jul. 2003.
- [12] I. Richardson, "H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia," John Wiley & Sons, 2003.
- [13] 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.
- [14] TK Tan, G. Sullivan, T. Wedi, "Recommended Simulation Conditions for Coding Efficiency Experiments Revision 3," ITU-T SC16/Q.6, Doc. VCEG-AI10, Berlin, Germany, July 2008.
- [15] G. Bjontgaard, "Calculation of average psnr differences between rd-curves," ITU-T SG16/Q.6 Doc. VCEG-M33, Austin, Texas, USA, April 2001.
- [16] KTA Reference Software Version 2.2, <http://iphome.hhi.de/suehring/tml/download/KTA/>

### 저 자 소 개



#### 정 광 수

- 2008년 2월 : 광운대학교 컴퓨터공학과 학사
- 2010년 2월 : 광운대학교 컴퓨터공학과 석사
- 2010년 3월 ~ 현재 : LG전자 MC연구소 Firmware part 연구원
- 주관심분야 : 비디오 신호처리, 비디오 압축, 영상 화질측정

---

저 자 소 개

---



남 정 학

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



임 응

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



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



최 병 두

- 2007년 2월 : 고려대학교 전자컴퓨터공학과 공학박사
- 2007년 ~ 2008년 : Fraunhofer HHI 연구소 방문연구원
- 2009년 : 고려대학교 BK21 연구교수
- 2009년 ~ 현재 : 삼성전자 DMC 연구소 멀티미디어 연구팀 책임연구원
- 주관심분야 : 영상처리, 영상압축

---

저 자 소 개

---



조 대 성

- 1994년 : 서강대학교 전자공학과 학사
- 1996년 : 서강대학교 전자공학과 석사
- 1996년 ~ 2008년 : 삼성종합기술원 컴퓨팅랩 전문연구원
- 2008년 ~ 현재 : 삼성전자 DMC 연구소 멀티미디어 연구팀 수석연구원
- 주관심분야 : 영상처리, 영상압축