

HEVC 기술 소개 - Part I: 기본 구성 요소



김재일
한국과학기술원



안상수
한국과학기술원



김대운
한국과학기술원



김문철
한국과학기술원

I. 서론

HEVC는 ISO/IEC 산하 MPEG 과 ITU-T 산하 VCEG 이 공동으로 Joint Collaborative Team on Video Coding (JCT-VC)을 구성하여 표준화한 차세대 비디오 압축 표준이다. MPEG 쪽에서는 ISO/IEC 23008-2 - MPEG-H Part 2 High Efficiency Video Coding으로, VCEG에서는 ITU-T H.265로 불리고 있으며, 공식적인 공통명은 HEVC 명명된다 (간혹 논문 등에서 H.265/HEVC로 표기하나 이는 공식적인 이름이 아님). 2013년 1월 제12차 JCT-VC 제네바 회의에서 HEVC의 기본 구성 요소에 대한 1차 버전 최종 표준

HEVC는 ISO/IEC MPEG과 ITU-T VCEG이 공동으로 구성한 JCT-VC에서 표준화한 차세대 비디오 압축 표준이다.

안이 발표되었다. HEVC는 기존 H.264/MPEG-4 AVC와 비교하여 거의 동등 주관적 화질 대비 약 2배의 압축 부호화 성능을 목표로 개발되었으며, 향후 UHD TV 방송, 3DTV 방송, IPTV, 감시카메라, 디지털 캠코더, 스마트폰 등의 응용분야에서 고화질 비디오 서비스에 빠른 속도로 적용될 가능성이 매우 높은 중요한 비디오 부호화 표준이다. 본고에서는 이러한 HEVC의 기본 구성 요소를 기존 H.264/MPEG-4 AVC 표준과 대비하여 설명한다.

II. HEVC 표준화 현황

2003년 H.264/MPEG-4 Part 10 Advanced Video Coding (AVC) 표준이 제정된 이후, AVC 표준은 DMB, IPTV, 디지털 카메

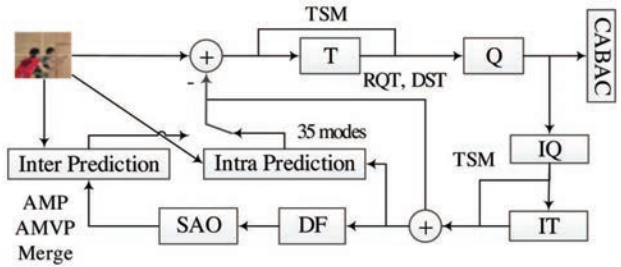
라, 감시시스템 등 다양한 응용 분야에 빠른 속도로 적용되어 산업화 되었다. 이후, ISO/IEC MPEG과 ITU-T VCEG은 서로 AVC 표준 기술보다 부호화 성능이 더 향상된 새로운 차세대 비디오 압축 부호화 기술 표준을 마련하기 위해 요구사항 및 사전 기술 탐색 실험을 수행해 왔다. 그러던 중 2010년 1월에 MPEG과 VCEG는 공동으로 차세대 비디오 압축 부호화 규격 개발을 위한 기술제안요청서를 발표하였고, 2010년 4월 독일 드레스덴에서 개최된 제1차 JCT-VC 회의에서 27개의 기술 제안이 이루어졌으며, 이를 바탕으로 HEVC 표준 개발을 본격적으로 시작하였다. 이후, 많은 기술 기고와 기고 기술들의 표준 채택을 위한 경쟁을 통해 2013년 1월 HEVC 1차 버전 최종 표준안을 완성하였다.

이후, HEVC 1차 버전은 바탕으로, 전문가용 컬러포맷 (4:2:2, 4:4:4) 및 높은 비트 샘플(12 bit 이상)을 지원하는 2차 버전인 HEVC Range Extensions을 표준화 하였으며, 다시점 비디오 부호화를 위한

Multi-View HEVC, 다시점과 깊이맵을 부호화하기 위한 3D HEVC, 그리고 컴퓨터 스크린 영상을 부호화하기 위한 HEVC Screen Content Coding 기술에 대한 표준화를 진행하고 있다. 본고는 2개의 파트로 구성되어 있다. Part I에서는 2D 비디오 부호화를 위한 HEVC 1차 버전에 대한 기술적 개괄을 제공하고, Part II에서는 HEVC 1차 버전에 채택된 부호화 툴의 특징에 대해 상세설명하기로 한다.

III. HEVC 기본 구성 기술

HEVC 비디오 부호화 표준은 기존의 MPEG-2 Video, AVC와 같은 비디오 부호화 표준들과 비슷하게 부호화 블록을 기반으로 다양한 부호화 모드를 선택적으로 사용하여 부호화 하는 하이브리드 비디오 부호화 방식을 채택하고 있다. 하지만 HEVC는 UHD TV와 같은 대형 화면에서부터 스마트 폰과 같은 작은 화면까



〈그림 1〉 HEVC의 부호화 과정

HEVC는 기본적으로 하이브리드 비디오 부호화 방식이며, 다양한 부호화 기술을 도입하여 AVC 대비 동등 주관적 화질 대비 약 50% 정도의 부호화 성능을 향상 하였다.

지 다양한 크기 화면에 대해 개선된 압축 성능을 얻기 위해 신호 특성에 적합한 다양한 예측 방법 및 변환 방법을 도입하였다. 특히 기본 부호화 단위 블록의 크기를 최대 64×64 크기로 확장하여 부호화 블록이 재귀적 분할 구조로 분할 되도록 설계하여 다양한 특성의

입력 신호에 대해 부호화 효율이 최대화 되도록 부호화 단위 블록이 적응적으로 분할되어 부호화될 수 있도록 하였다. 변환 블록 또한 신호 특성에 따라 가변 블록 크기로 적용될 수 있도록 하여 압축 성능 향상을 얻었다. 움직임 벡터 부호화 향상을 위한 Advanced Motion Vector Prediction (AMVP), 유사 움직임 벡터 재사용을 위한 움직임 블록 병합(Merge) 및 스킵(SKIP) 예측 구조 등을 사용하여 부호화 향상을 꾀하였다. 또한, 비디오 부호화 과정이 하드웨어 및 소프트웨어 플랫폼에서 병렬화 처리가 가능하도록 설계되어, 기존 AVC 구현시 발생되었던 부호소 화면 보간 및 더블록 필터, CABAC의 병목 현상을 개선하였다. HEVC의 주관적 화질 및 객관적 화질 개선을 위하여 인루프(in-loop) 필터인 디블록킹필터(DF: Deblocking Filter) 이외에 SAO(Sample Adaptive Offset) 필터를 추가적으로 도입하여 부호화된 영상의 복원영상을 원 영상과 최대한 비슷하도록 하는 기법도 채택 되었다. 이러한 기술들로 인하여, HEVC는 AVC 대비 동등 주관적 화질 대비 약 50% 정도의 부호화 성능을 향상 하였다^[1]. 〈그림 1〉은 HEVC 부호화 과정 블록도를 나타낸다. 〈그림 1〉에서 T는 변환 과정, Q는 양자화 과정, IT는 역변환 과

직임 벡터 부호화 향상을 위한 Advanced Motion Vector Prediction (AMVP), 유사 움직임 벡터 재사용을 위한 움직임 블록 병합(Merge) 및 스킵(SKIP) 예측 구조 등을 사용하여 부호화 향상을 꾀하였다. 또한, 비디오 부호화 과정이 하드웨어 및 소프트웨어 플랫폼에서 병렬화 처리가 가능하도록 설계되어, 기존 AVC 구현시 발생되었던 부호소 화면 보간 및 더블록 필터, CABAC의 병목 현상을 개선하였다. HEVC의 주관적 화질 및 객관적 화질 개선을 위하여 인루프(in-loop) 필터인 디블록킹필터(DF: Deblocking Filter) 이외에 SAO(Sample Adaptive Offset) 필터를 추가적으로 도입하여 부호화된 영상의 복원영상을 원 영상과 최대한 비슷하도록 하는 기법도 채택 되었다. 이러한 기술들로 인하여, HEVC는 AVC 대비 동등 주관적 화질 대비 약 50% 정도의 부호화 성능을 향상 하였다^[1]. 〈그림 1〉은 HEVC 부호화 과정 블록도를 나타낸다. 〈그림 1〉에서 T는 변환 과정, Q는 양자화 과정, IT는 역변환 과

정 그리고 IQ는 역양자화 과정을 나타낸다. TSM은 변환과정을 거치지 않고 양자화 과정만은 이용한 부호화 방법인 Transform Skip Mode를 나타낸다. AMP는 Asymmetric Motion Partition의 약자로서 비대칭 움직임 블록 분할을 통한 움직임 예측 방법(틀)을 나타낸다. 본 장에서는 HEVC의 성능 향상을 위해 확장 및 새롭게 제안된 기술에 대해 소개한다.

1. 확장 블록 부호화 구조

HEVC는 다양한 특성의 영상 신호를 효과적으로 예측 부호화하기 위해 기존의 16×16 매크로블록보다 큰 확장된 부호화 단위 유닛인 CTU(Coding Tree Unit)를 가지며, CTU는 신호의 특성에 따라 쿼드트리 구조로 분할되는 유연한 부호화 유닛(CU: Coding Unit) 블록을 가진다. 이 각각의 CU 블록은 재귀적 쿼드트리 구조로 하위 CU로 분할될 수 있다. <그림 2>는 CTU 및 CU 분할 블록의 예와 각 CU 분할 블록 내의 예측 유닛(PU: Prediction Unit) 블록과 변환 유닛(TU: Transform Unit) 분할 블록의 예를 나타낸다^[1]. <그림 2>에서 보듯이 이러한 유연한 CU 분할 구조와 CU 내

의 PU 및 TU의 분할 구조는 HEVC가 향상된 부호화 성능을 발휘하는데 크게 기여하였다.

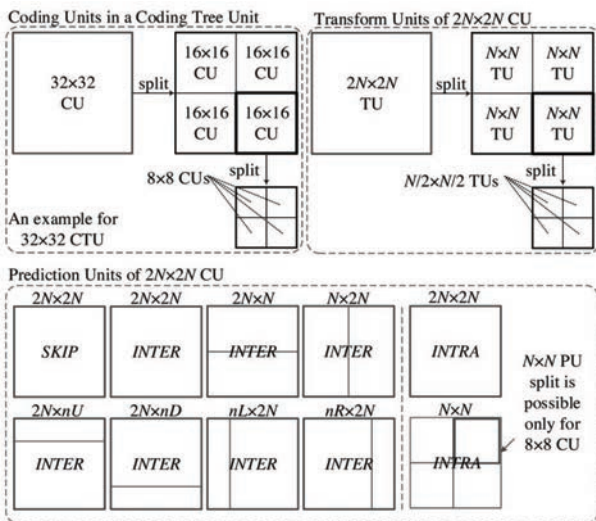
AVC의 매크로블록에 대응되는 HEVC의 CTU는 64×64, 32×32, 16×16, 8×8 크기 중의 하나로 선택이 가능하다. HEVC의 기본 부호화 단위인 CU는 <그림 2>와 같이 각 크기마다 분할 직전 CU의 가로, 세로 크기의 1/2 크기인 쿼드트리 형태로 분할될 수 있다. 예를 들어 <그림 2>와 같이 CTU 크기가 32×32인 경우, 가능한 CU 분할 블록의 크기는 32×32, 16×16, 8×8이다. HEVC의 CU 부호화 과정은 화면간/화면내 예측 모드로 수행될 수 있다. 화면간/화면

HEVC는 기존 16x16매크로블록보다 큰 확장된 부호화 단위인 CTU(Coding Tree Unit)를 가지며 이는 쿼드트리 구조로 분할되는 유연한 CU(Coding Unit)를 가진다.

내 예측을 위한 예측 블록 유닛을 HEVC에서는 PU라 하며 <그림 2>와 같이 CU가 2N×2N일 때 2N×2N SKIP, 2N×2N Inter, 2N×N Inter, N×2N Inter, 2N×2N Intra, N×N Intra 예측 모드중 하나를 선택하여 부호

화 된다(단, 8×8 CU의 경우에만 4×4 Intra 모드를 지원한다). 뿐만 아니라 위 직사각 형태의 PU 모드 외에 2N×nU, 2N×nD, nL×2N, nR×2N 형태의 모드도 지원하며^[2], 화면간 예측 모드에서 대칭형/비대칭형 PU 모드의 예가 <그림 2>에 보여진다. 예를 들어 64×64 CU는 64×64 SKIP, 64×64 Inter, 64×32 Inter, 32×64 Inter, 64×16 Inter, 64×48 Inter, 16×64 Inter, 48×64 Inter, 64×64 Intra PU 모드 중 하나를 사용하여 부호화 될 수 있다.

HEVC의 TU는 <그림 2>와 같이 PU를 수행해 얻어진 CU 내의 잔차신호에 대하여 PU의 형태와 상관없이 쿼드트리 형태로 32×32에서부터 4×4 까지 정수화된 이산여현변환(DCT: Discrete Cosine Transform)을 통해 변환 부호화 과정을 수행한다. HEVC의 변환 과정은 AVC와는 다르게 TU 블록이 PU 블록의 경계를 넘는 잔차신호에 대해 수행될 수 있는 것이 특징이다^[1]. AVC의 CBP(Coded Block Pattern)에 해당하는 HEVC의 CBF(Coded Block Flag)는 각각의 TU 블록이 변환 및 양자화 이후 영이 아닌 양자화 변환 계수 포함여부



<그림 2> HEVC의 2N×2N CU에 대한 PU 구조

에 대한 정보를 시그널링 한다.

HEVC에서 최적의 분할 블록 (CU/PU/TU) 조합은 식 (1)과 같이 AVC에서 부호화 모드 결정을 위해 사용한 최소 윌-왜곡 값을 기반으로 하는 최적의 모드 결정 방법을 이용한다.

$$J_n = SSE_n(QP) + \lambda(QP) \cdot R_n(QP) \quad (1)$$

식 (1)에서 R_n 은 n-번째 CU, PU, TU 블록 조합 모드를 사용하여 부호화하였을 때 발생한 비트수이며, 이때 SSE_n (Sum of Squared Error)은 원신호($f_n(x, y)$)와 복원된 신호($\hat{f}_n(x, y)$)간 차이의 제곱합으로서 식 (2)와 같이 계산된다.

$$SSE_n(QP) = \sum_{x,y} (f_n(x, y) - \hat{f}_n(x, y))^2 \quad (2)$$

결국, 식 (1)을 통하여 발생 비트 수를 최소화 하면서 복원 화질의 왜곡이 최소화되는 부호화 모드가 최종 부호화 모드로 선택된다.

<그림 3>은 실제 BQMall의 한 영상에서 CTU 내의 CU, PU, TU의 최종 분할 블록 모드를 도식화한 그림이다. CU 블록이 복잡도가 높거나 움직이는 물체의

경계에 걸쳐 있는 경우, 작은 CU 블록으로 분할되어 부호화 된 것을 볼 수 있다. 앞서 TU에 관해 언급했듯이 $nR \times 2N$ PU 블록의 경우 $nR \times 2N$ 경계에 상관없이 $2N \times 2N$ 크기로 변환($2N \times 2N$ TU)이 수행된 것을 확인 할 수 있다. 또한, HEVC에서는 각 크기의 CU마다 SKIP 모드가 지원되는 것이 특징인데 <그림 3>에서 보듯이 다양한 크기의 많은 CU 블록이 SKIP 모드(노란색으로 표시된 부분)를 선택하여 부호화된 것을 알 수 있다.

이와 같이 HEVC는 SKIP 모드를 포함한 다양한 크기의 블록 예측 및 가변 블록 변환을 통해 입력 신호에 적응적인 효과적 부호화 수행이 가능해져서 크게 개선된 부호화 성능을 발휘 할 수 있게 되었다.

2. 화면내 예측 모드

AVC에서 휘도 신호의 화면내 예측 방법은 16×16

크기의 블록에 대하여 DC와 plane 모드를 포함한 4개의 방향성 예측 모드를 가지고, 8×8 과 4×4 크기의 블록에 대해서는 <그림 4(a)>와 같이 DC 예측 모드를 포함하여 총 9개의 방향

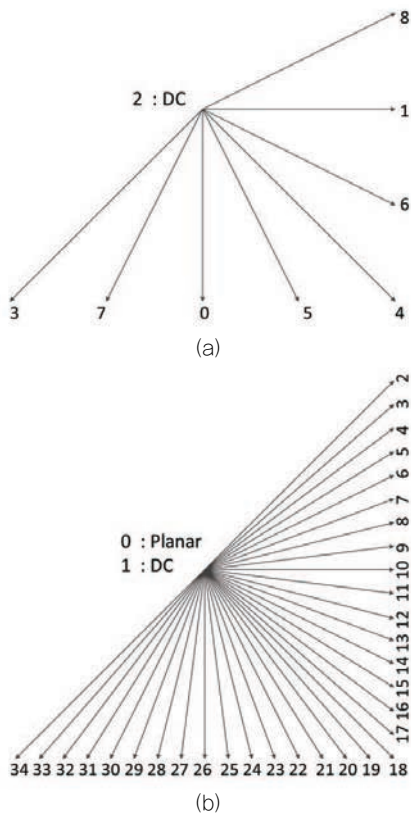
성 예측 모드를 통해 화면내 예측 부호화를 수행한다^[3]. 반면, HEVC의 휘도 신호에 대한 화면내 예측 방법은 AVC 보다 정교한 방향성 예측을 수행하는데 <그림 4(b)>와 같이 DC와 planar 모드를 포함한 총 35개의 방향성 예측 모드를 4×4 , 8×8 , 16×16 , 32×32 , 64×64 크기의 블록에 대해 적용하며, 방향성 예측 데이터를 생성하기 위해 $1/32$ -화소 단위의 정교한 선형 보간 방법을 사용한다^[4].

기존 AVC에 비해 HEVC는 휘도 신호 블록에서 총 35개의 부호화 모드를 지원하기 때문에 부호화 모드를 시그널링하기 위한 비트 오버헤드가 커지게 된다. 따라서 화면내 예측 모드 정보를 효율적으로 부호화하기 위하여 주변 블록 모드 정보 및 선택 비율이 높은 모드 정보를 포함하여 총 3개의 MPM(Most Probable Mode) 정보를 사용한다^[5]. AVC의 16×16 plane 모

HEVC의 휘도 신호에 대한 화면내 예측 방법은 총 35개의 방향성 예측 모드를 지원하여 AVC 보다 정교한 방향성 예측을 수행한다.



<그림 3> CTU내 CU, PU, TU 부호화 선택 예^[12]



〈그림 4〉 AVC 및 HEVC의 화면내 방향성 예측 모드

드와 유사한 HEVC의 planar 모드는 16×16 plane 모드에 비하여 병렬처리와 단순 연산이 가능하게 되었다. HEVC의 planar 모드는 CTU내 모든 CU 크기의 휘도 및 색차 신호에 적용 가능하며 이는 텍스처가 점차적으로 변하는 단순한 영역에서의 압축 성능 및 주관적 화질을 향상하는데 매우 중요한 부호화 도구로 사용된다.

색차 신호 블록의 화면내 예측을 위하여 AVC에서는 DC(모드 0), Horizontal(모드 1), Vertical(모드 2), Plane(모드 3) 모드만을 사용한다. 하지만, HEVC는 planar (모드 0), vertical (모드 26), horizontal (모드 10), DC (모드 1) 모드 이외에 휘도 신호 블록에서 최적의 방향성 모드로 선택되었던 모드를 재사용하는 모드(DM 모드)를 추가로 사용하여 색차 신호 블록의 화면내 예측 부호화를 수행한다. 색차 신호는 휘도 신

화면내 예측 부호화 모드는 다양한 예측 부호화 블록의 종류, 향상된 움직임 벡터 예측 방법(Advanced Motion Vector Prediction 및 Merge)을 적용하였다.

호에 비하여 복잡도가 낮고, 복잡도가 높다 하더라도 휘도의 방향성과 비슷한 경우가 많기 때문에 위와 같은 다섯 가지 예측 모드만을 이용하여 색차 신호의 화면내 방향성 예측 부호화를 효과적으로 수행할 수 있다^[5].

3. 화면간 예측 모드

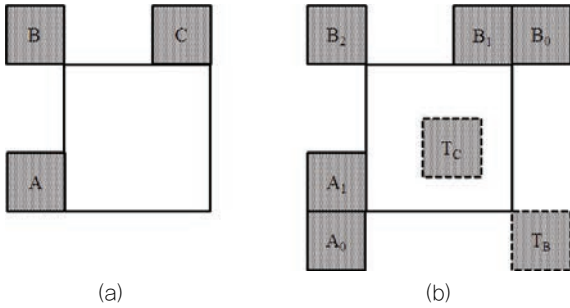
화면간 예측 부호화 모드는 AVC에 비해 예측 부호화 블록의 종류가 다양해졌으며, 움직임 벡터의 예측 성능을 향상하기 위해 이전 비디오 표준에서 사용하던 참조 블록 보다 더 많은 참조블록을 사용하고, 사용된 참조 블록에 대한 정보를 복호화기로 전송한다^[1]. 또한 움직임 벡터에 사용되는 정보량을 줄이기 위해 Merge 모드로 명명되는 새로운 움직임 벡터 부호화 기술을 적용하였다.

3.1 Advanced Motion Vector Prediction (AMVP)

〈그림 5〉에서 보듯이 H.264/MPEG-4 AVC는 움직임 벡터의 예측을 위해 동일 화면내의 주변 블록 3개만을 이용하지만, HEVC는 기본적으로 동일 화면내의 주변 블록 5개를 사용하고 선택적으로 한 개의 collocated 블록을 참조 블록으로 사용하여 움직임 벡터의 예측(AMVP)을 수행한다^[6].

〈그림 5(a)〉에서 AVC는 3개의 주변 블록의 움직임 벡터 중 중간 값을 일률적으로 움직임 벡터 예측 값으로 사용하며, 복호화기에서도 추가적인 정보 없이 주변 블록의 움직임 벡터 중 중간 값을

계산할 수 있으므로 움직임 벡터 예측을 위해 부가적으로 전송되는 정보가 없다. 반면에, 〈그림 5(b)〉에서 보듯이 HEVC는 좌측 2개의 참조 블록(A_0, A_1) 중 A_0 를 기본 후보 블록으로 사용하고 A_0 가 참조 블록으로 사용가능하지 않을 때 A_1 를 좌측 후보 블록으로 사용한다. 상단 3개의 참조 블록(B_0, B_1, B_2)도 동일하게 B_0 를 기본 후보 블록으로 사용하고 B_0 가 참조블록으로 사용가능하지 않을 때 B_1 를 후보 블록으로 사용하고,



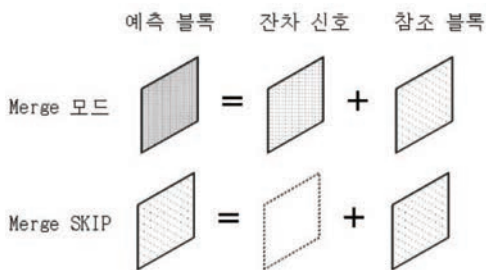
〈그림 5〉 H.264/AVC와 HEVC의 MV 예측 비교

B_0 과 B_1 모두 사용가능하지 않을 때 B_2 를 후보 블록으로 사용한다. 만약 후보 1이나 후보 2가 존재하지 않을 경우 collocated 블록을 나머지 후보로 사용한다. 〈그림 5(b)〉에 보이듯이 collocated 블록은 기본적으로 T_B 가 사용되고 T_B 가 사용가능하지 않을 때 T_C 가 사용된다. AMVP는 부호화 성능 향상을 위해 움직임 벡터 스케일링 과정(scaling process)을 사용한다. Scaling process는 현재 화면과 참조 화면의 시간상 차이를 보정하여 움직임 벡터에 적용함으로써 부호화 성능 향상을 가능하게 하는 방법이다. 스케일링 값은 식 3과 같이 현재 화면과 참조화면의 Picture Order Count (POC) 차이와 현재 화면과 움직임 예측에 사용되는 참조 블록의 참조화면 POC 차이의 비로 정해진다^[6].

$$\text{Scaling Factor} = \frac{POC_{curr} - POC_{ref}}{POC_{curr} - POC_{neighbor_ref}} \quad (3)$$

스케일링을 포함한 움직임 벡터 예측 과정을 요약하면 아래와 같다.

1. 좌측 주변 블록에서 후보 블록을 결정.
2. 1번에서 결정된 후보블록의 움직임 벡터를 스케



〈그림 6〉 Merge 모드와 Merge_SKIP의 비교

일링하여 움직임 예측 벡터로 사용.

3. 1과 2 중 최소 RD 비용 관점에서 최적의 움직임 벡터를 선택.
4. 상단 주변 블록에서 후보 블록을 결정.
5. 3번에서 스케일링된 움직임 벡터가 선택되지 않았을 경우 4번의 움직임 예측 벡터에 스케일링 과정을 적용함. 단, 3번에서 스케일링된 움직임 벡터가 선택되었을 경우 5번 과정은 생략.

최종적으로 결정된 2개의 후보 블록 중 최적의 움직임 예측 성능을 보이는 후보 블록을 움직임 예측 블록으로 선택하고, 후보 블록 중에서 어떤 블록을 참조 블록으로 사용하였는지에 대한 정보를 부호화기로 전송한다. 움직임 벡터 예측 블록에 대한 정보량이 증가하지만 기존의 비디오 부호화 표준 보다 더 정밀한 움직임 예측이 가능하므로 전체적인 정보량을 감소시킬 수 있다.

3.2 Merge 모드와 Merge_SKIP 모드

움직임 벡터 Merge 모드는 기본적인 Merge 모드와 Merge_SKIP 모드 2가지로 이루어져 있다. 우선 기본적인 움직임 Merge 모드는 AMVP와 동일하게 5개의 주변 블록과 2개의 collocated 블록을 사용한다. 〈그림 6〉은 Merge 모드와 Merge_SKIP 모드의 비교를 보인다.

〈그림 6〉에서 Merge 모드는 AMVP와 다르게 참조 화면 번호나 움직임 차분 정보(MVD: Motion Vector Difference) 등이 전송되지 않고 참조 블록의 참조 화면 번호와 움직임 벡터가 현재 블록의 움직임 예측에 그대로 사용된다. 따라서 오직 merge_mode_flag 정보와 Merge 모드 후보 블록 번호만이 전송된다. 반면에, Merge_SKIP 모드는 기본적인 Merge 모드와 다르게 잔차 신호가 전송되지 않는다. 기본적인 Merge 모드는 참조 블록의 움직임 정보를 이용하여 움직임 예측을 수행하므로, 예측 참조 블록이 존재하고 예측 참조 블록과 현재 블록의 잔차 신호가 전송된다. 하지만 Merge_SKIP 모드는 움직임 예측 과정을 수행하지 않고 예측 참조 블록을 그대로 현재 블록의 예측 블록으로 사용하므로 잔차 신호가 전송되지 않는다.

〈표 1〉 휘도 신호의 필터계수

부화소 위치	필터 계수
1/4	-1, 4, -10, 58, 17, -5, 1, 0
1/2	-1, 4, -11, 40, 40, -11, 4, -1
3/4	0, 1, -5, 17, 58, -10, 4, -1

3.3 부화소 수준 움직임 예측을 위한 보간 필터

HEVC는 움직임 예측의 정확도를 향상시키기 위해 복원 영상에 보간 필터를 적용하여 부화소 영상을 만들고 이를 움직임 예측에 사용한다^[1]. 휘도 신호를 예로 들어 설명하면, 1/2 화소 위치의 보간 화소 값을 생성하기 위하여 8-tap 필터를, 1/4 화소 위치에서는 7-tap 필터를 한번에 적용한다. 휘도 신호의 보간에 사용되는 3 가지 종류의 필터 계수는 〈표 1〉과 같다.

〈그림 7〉은 정수 화소를 사용하여 보간한 부화소들의 위치를 나타낸다. 대문자로 쓰인 화소가 정수 화소이고 소문자로 쓰인 화소가 1/2 또는 1/4 부화소이다.

보간 과정은 우선 〈표 1〉에 있는 보간 필터를 정수 화소에 적용하여 a, b, c, d, h, n 위치의 부화소를 생성하고 앞서 생성된 부화소를 이용하여 나머지 9개의 부화소를 생성한다. 예를 들면, $a_{0,0}$ 는 1/4 위치이므로 1/4 위치의 필터계수를 $A_{-3,0}$, $A_{-2,0}$, $A_{-1,0}$, $A_{0,0}$, $A_{1,0}$, $A_{2,0}$ 의 정수 화소 위치에 적용하여 $a_{0,0}$

〈그림 7〉 HEVC의 보간 화소

화소를 보간한다. $b_{0,0}$ 의 경우 위와 동일한 정수 화소에 1/2 화소를 위한 필터를 적용한다. $c_{0,0}$ 또한 동일한 정수 화소에 3/4 화소를 위한 필터를 적용한다. 각 위치의 필터 계수에서 알 수 있듯이 부화소와 정수화소와의 거리에 비례하여 필터 계수를 조정함으로써 동일한 정수화소를 이용하여 각 부화소를 보간 할 수 있도록 하였고 이에 따라 1/2 부화소와 1/4 부화소를 동시에 보간 할 수 있는 이점이 있다.

보간 필터를 통하여 기존 영상보다 4배 확장된 영상을 이용할 수 있게 되었고 이를 통하여 정수 화소 움직임 뿐 아니라 부화소 움직임단위로 움직임 예측을 수행할 수 있게 된다. 이로 인해 더 정교한 움직임 예측이 가능하게 되어 전송해야 할 잔차 신호의 양을 감소시켜 전체적인 부호화 성능향상에 기여한다.

4. 변환 및 양자화

기존의 하이브리드 비디오 부호화 방식과 유사하게 HEVC에서는 화면내 또는 화면간 예측이 수행된 이후 예측 블록과 원 신호 블록 간의 4×4 , 8×8 , 16×16 , 32×32 크기의 잔차신호 블록에 대해 변환 및 양자화 과정을 수행한다. HEVC의 변환 과정은 변환 커널 계수가 모든 크기에 대하여 일관성을 갖는 변환 커널이 최종적으로 채택되어 작은 크기의 변환 커널 계수를 따로 저장할 필요가 없이 큰 변환 커널 계수만을 이용해서 작은 변환 커널 값의 생성이 가능하다^[5]. 뿐만 아니라 HEVC의 변환 과정이 16 비트 이내에서 연산이 가능하도록 절삭연산 (clipping operation)을 수행하게 하여 하드웨어 구현시 복잡도 증가를 억제하였다. 한편, 양자화 및 역양자화 과정의 효율적인 설계를 위하여 다중 norm을 사용한 AVC와는 달리, HEVC에서는 모든 변환 커널 요소(주파수) 위치에서 단일 norm을 사용했기 때문에 변환 계수의 모든 위치에서 주어진 QP 값에 대해 동일한 양자화 스텝으로 양자화를 수행한다.

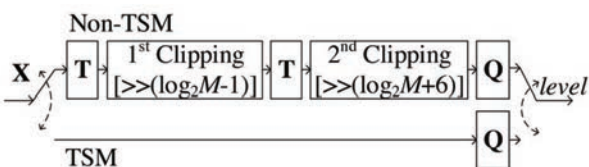
HEVC의 변환 커널은 DCT 기반의 정수 변환 커널 뿐만 아니라 화면내 예측 모드의 4×4 블록에 대해서는 DST(Discrete Sine Transform) 기반의 정수 변환

커널을 적용한다^[5]. 이는 4×4 화면내 예측의 경우 예측 블록내 화소간의 상관도가 높기 때문에 DST를 적용함으로써 부호화 성능 개선이 가능하였기 때문이다^[4]. 또한, 4×4 블록의 화면내 및 화면간 예측 모드에서 변환을 수행하지 않고 바로 양자화를 수행하는 변환 스킵 모드(TSM: Transform skip mode)를 지원한다^[1]. 텍스트(글자)가 많이 존재하는 영상 등에서 볼 수 있듯이, 블록 내 텍스처가 복잡한 신호의 경우 변환을 수행하지 않는 것이 부호화 성능을 향상하게 된다. 따라서 이런 특별한 경우 부호화 성능 및 주관적 화질을 향상하기 위하여 변환 스킵 모드가 지원된다.

〈그림 8〉은 변환 및 변환 스킵과 양자화 과정을 나타낸다. 〈그림 8〉에서 M은 변환 블록의 크기에 해당한다. 변환 스킵 모드인 경우 변환 과정은 수행하지 않고 양자화 과정만을 수행하며, 변환 스킵 모드가 아닌 경우 변환 및 두 번의 절삭연산(1st Clipping과 2nd Clipping)을 통한 2-D 변환 및 양자화 과정을 수행 한다.

5. 엔트로피 부호화 (Entropy Coding)

비디오 부호화에서 엔트로피 부호화 과정은 부호화의 각 모듈에서 생성된 신택스 요소를 최종 비트스트림으로 출력한다. HEVC에서는 AVC에서 이용된 엔트로피 부호화기중 하나인 CABAC(context-adaptive binary arithmetic coding)^[7]을 이용한다^[1]. 신택스 요소를 압축하기 위해 문맥 정보를 얻어서 이진 산술 부호화를 한다는 점에서 AVC의 CABAC과 HEVC의 CABAC은 개념적으로 거의 동일하다. 그러나 HEVC의



〈그림 8〉 HEVC의 변환 및 변환 스킵과 양자화 과정

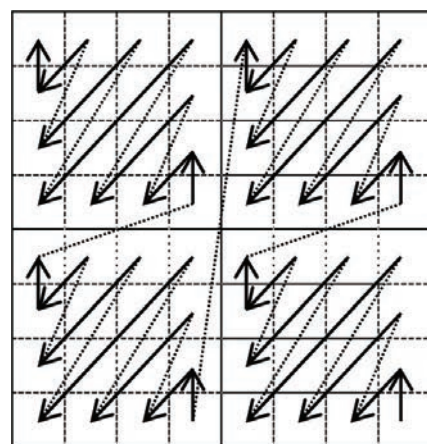
경우 AVC에 비해 총 문맥 정보 수가 감소하였고 다양한 기술을 이용하여 엔트로피 부호화 처리 용량(throughput)을 개선하였다^[8]. 또한, AVC와 다르게 쿼드 트리 구조의 CU나 TU의 깊이 역시 문맥 정보에 영향을 준다^[1].

HEVC 표준에서 잔차신호를 변환 및 양자화하여 생성된 변환계수를 엔트로피 부호화하기 위한 계수 스캐닝은 4×4 크기로 이루어진다. 즉 변환 블록의 크기가 4×4 보다 큰 16×16 이나 8×8인 경우에는 16개 또는 4개의 4×4 영역으로 나누어 스캐닝 한다. 〈그림 9〉는 8×8 크기 변환 블록의 변환 계수를 대각선 방향으로 스캐닝하는 순서를 나타낸 것이다.

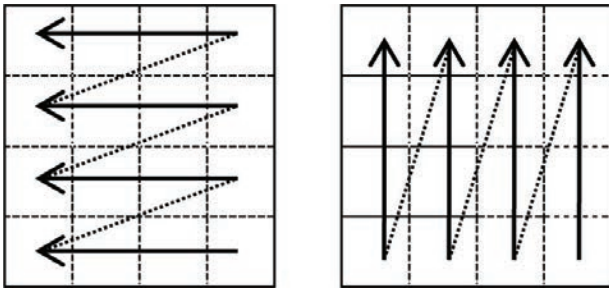
화면내 예측으로 부호화 된 경우는 변환계수 스캐닝 방향이 〈그림 9〉에서 나타나는 대각선 방향 외에도 화면내 예측의 방향에 따라 수평 방향, 수직 방향이 가능

하다. 〈그림 10〉은 수평방향, 수직방향으로의 변환계수 스캐닝을 나타낸다.

이렇게 대각선 방향 외에 수직, 수평 방향으로 변환 계수를 스캐닝 하는 것은 화면내 예측의 방향성에 따라 달라지는 잔차신호의 양자화된 변환 계수 배치를 효율적으로 부호화하기 위함이다.



〈그림 9〉 HEVC에서 8×8 TU 에 대한 대각선 (diagonal) 스캐닝 방법^[9]



〈그림 10〉 HEVC에서 화면내 예측의 방향성 스캐닝 방법^[9]

변환 계수를 스캐닝 한 후에 HEVC는 AVC와 유사하게 0이 아닌 마지막 변환계수의 위치, 중요도 맵, 변환 계수의 부호와 레벨 등을 전송한다. 그러나 HEVC에서는 변환 블록의 크기가 이전 표준에 비해 매우 크기 때문에 이를 더 적합하게 처리 가능하도록 변화 되었다.

6. 루프내 필터

루프내(in-loop) 필터는 복/부호화가 끝난 복원 영상에 적용되는 화질 향상 기술이다. 필터가 적용된 영상의 객관적 및 주관적 화질이 개선됨은 물론, 화질이 개선된 영상이 다음 부·복호화 과정에서 참조영상으로 이용된다는 점에서 부호화 효율 또한 향상된다. HEVC에는 두 종류의 루프내 필터를 채택하고 있다. 디블록킹 필터^[10]를 수행한 후 SAO 필터^[11]를 추가적으로 적용한다. 디블록킹 필터는 블록화 왜곡(blocking artifact)을 완화하기 위해 적용되는 필터로 AVC에서 이용되던 필터와 개념은 유사하지만 복잡도가 개선된 형태로 HEVC에 적용되었다. 한편 블록의 경계 주변에 있는 화소에만 적용되는 디블록킹 필터와 다르게 SAO 필터는 복원 영상의 모든 화소를 정해진 기준대로 분류하여 적응적으로 오프셋 값을 적용하여 원본영상과의 차이를 감소시킨다.

(1) 디블록킹 필터(deblocking filter)

블록화 현상이란 예측과 변환 및 양자화 등 비디오 부호화의 각 모듈이 블록 단위로 수행되기 때문에 나타

나는 현상이다. 원본 영상에서는 부드럽게 변화하던 화소 값이 인접한 두 블록의 부호화 특성의 상이함으로 인해 복원 영상에서는 블록의 경계 주변에서 불연속적으로 변화하며 나타나는 것이 블록화 현상이다. 이 블록화 현상은 주관적인 화질에 나쁜 영향을 미치며 부호화 효율 저하를 초래한다. 블록화 현상은 블록 경계에서 나타나므로 HEVC에서는 PU와 TU의 경계에 대해 디블록킹 필터를 적용한다. 이때 모든 경계에 대해 동일한 필터를 적용하지 않고 경계 강도(boundary strength, BS)와 경계 주변의 화소 값을 이용하여 계산된 값을 특정 임계치와 비교하여 필터 적용 여부 및 종류를 결정한다. 필터 적용 여부 및 종류를 결정하는데 있어서 중요한 기준인 BS 값은 AVC에서 0-4의 값을 가진 것과는 다르게 HEVC에서는 0-2의 값만을 갖도록 간소화 되었다. 또한 AVC에서는 4×4 크기의 블록에 필터가 적용되었지만 HEVC에서는 8×8크기 이

HEVC에는 디블록킹 필터를 수행한 후 SAO(Sample Adaptive Offset) 필터를 추가적으로 적용하여 두 종류의 루프내 필터를 채택하고 있다.

상의 블록에 한하여 적용시킴으로써 필터링 복잡도를 낮췄다.

또한 HEVC의 디블록킹 필터는 프레임 내의 모든 수직 경계에 대해 수평 필터가 적용된 이후에 프레임 내의 모든 수평 경계에 대해

수직 필터가 적용되며, 각각의 수평 필터링 및 수직 필터링 간에는 서로 의존성이 없으므로 다수의 필터링을 병렬적으로 수행할 수 있다.

(2) SAO(Sample Adaptive Offset)

SAO는 디블록킹 필터와는 달리 HEVC 표준에 새롭게 포함된 기술로서, 디블록킹 필터로 처리된 영상에 적용하여 원본 영상과의 오차를 줄이는 추가적인 루프내 필터이다. SAO는 이름에서 알 수 있듯이, 모든 화소를 4개의 카테고리로 분류하여 카테고리 마다 적응적으로 오프셋을 가감하여 화질을 향상시킨다. 화소를 4개의 카테고리로 분류하는 기준으로는 크게 두 가지 방법이 있다. 주변 화소와 현재 화소의 밝기 값을 비교하여 화소를 분류하는 EO(edge offset, SAO type 1-4)와 화소 값 자체를 이용하여 분류하는 BO(band

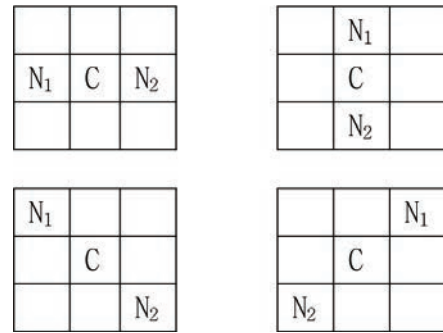
offset, SAO type 5)가 그 두 가지 방법이다. HEVC 부호화기에서는 CTU 별로 4개 타입의 EO와 1개 타입의 BO, 총 5개 타입의 SAO 타입 중 하나를 선택하여 CTU내의 화소를 4개의 카테고리 분류하여 오프셋을 가감한다. 또한, CTU에 따라 SAO를 적용하는 것이 오히려 전체 윌-왜곡 성능을 저하시키는 경우에 대해서는 해당 CTU는 SAO를 적용하지 않는다. 화소의 밝기 값을 기준으로 카테고리를 분류하는 BO는 복원 영상의 특정 밝기 영역에서 발생하는 오차를 줄인다. BO를 적용하는 경우 화소 값을 32 구간의 등 간격으로 분류한 후 CTU내의 모든 화소를 각 32개의 구간으로 분류한다. 그 중 원본 영상과 가장 밝기 차이가 많이 나는 연속되는 4구간에 대해 오프셋을 적용한다.

주변 화소와 현재 화소의 밝기 값의 관계를 이용하여 카테고리를 분류하는 EO는 부호화 과정 중에 에지 영역 주변에서 발생하는 주름 왜곡(ringing artifact)을 완화시키는 역할을 한다. EO를 적용하는 경우 우선 주변 화소를 정의하는 4가지 종류의 타입이 있다. <그림 11>은 현재 화소와 주변 화소를 정의하는 4개의 타입(SAO type 1-4)을 나타낸다. <그림 11>에서 나타나는 C와 N_1 , N_2 (숫자를 아래 첨자로 수정)의 밝기 값의 관계에 따라 C의 카테고리가 정해진다. <표 2>는 각 카테고리별 화소값의 관계를 나타낸다. 각 화소를 <표 2>에 따라 4개의 카테고리로 분류하여 오프셋을 적용한다.

부호화 효율 향상을 위해 현재 CTU의 왼쪽이나 위쪽 CTU의 SAO정보를 merge하여 SAO 선택 요소에 할당되는 비트를 줄일 수도 있다.

7. 하이레벨 선택

JCT-VC 표준화 회의과정에서 AVC 하이레벨 선택을 기반으로 HEVC에서 새롭게 요구되는 사항들을 반영하여 HEVC 하이레벨 선택을 설계 하였다. 따라서, 본고에서는 AVC 하이레벨 선택에서 확장되거나 새롭게 추가된 HEVC 하이레벨 선택 기술을 중심으로 기술한다.



<그림 11> 주변화소를 정의하는 4 타입

<표 2> 화소 값의 관계에 따른 카테고리 분류

카테고리	화소 값의 관계
0	아래에 해당 없음
1	$C < N_1$ and $C < N_2$
2	$C < N_1$ and $C = N_2$ or $C = N_1$ and $C < N_2$
3	$C > N_1$ and $C = N_2$ or $C = N_1$ and $C > N_2$
4	$C > N_1$ and $C > N_2$

(1) 병렬 처리를 위해 추가 확장된 선택

HEVC에서는 화면의 부호화의 병렬 처리를 위하여 AVC에서 지원되었던 슬라이스 선택과 새롭게 채택된 타일(tile)과 WPP (Wavefront Parallel Processing)을 지원한다^[1]. 타일은 슬라이스와 비슷하게 각 타일이 서로 독립적으로 부·복호화가 수행된다. 또한, 타일은 항상 직사각형태의 모양만을 가질 수 있으며, 반드시 온전한 형태의 CTU를 포함하고 있어야 한다. HEVC에서 타일과 슬라이스를 혼용해서 사용할 수도 있는데, 이때 아래 두 조건중 하나의 조건이라도 만족해야 한다^[5].

- (a) 한 슬라이스 안에 있는 모든 CTU는 같은 타일에 속해야 한다.
- (b) 한 타일 안에 있는 모든 CTU는 같은 슬라이스에 속해야 한다.

WPP는 부호화 단위 블록을 기반으로 병렬 부·복호화 과정을 수행하기 위해, MPEG-2 Video, MPEG-4 Visual, AVS(Audio Video Standard), VC(Video Codec)-1을 비롯한 AVC에서 비표준 형태로 사용되어 왔던 기술이다. 하지만, 이러한 비표준 방식의 WPP는 엔트로피 부호화 및 복호화에서 생기는 병목 현상을 해결하지 못하는 단점이 있다^[13]. 따라서 HEVC는 CTU



블록들이 행(row) 단위로 병렬화 처리가 가능하도록 WPP를 정의하였는데, 현재 CTU 블록의 왼쪽과 위, 대각선 위쪽 CTU의 부호화(복호화)가 끝난 직후 곧바로 부호화(복호화)가 가능하도록 하였다. 또한, HEVC WPP는 엔트로피 부호화에서 CTU 행의 첫 번째 CTU 블록의 엔트로피 부호화 성능 저하를 최소화함과 동시에 병렬화 처리가 가능하도록 하기 위해, 현재 CTU 블록의 오른쪽 위 CTU 블록의 문맥 모델(context model)을 이용해 CABAC을 수행하도록 하였다. 결국, HEVC WPP는 CABAC의 엔트로피 부호화 성능 저하를 최소화 하면서도 CTU 행 단위의 병렬처리가 부호화기(복호화기) 구현에 적용 가능하도록 설계되었다.

(2) 임의접근을 위한 하이레벨 선택스

AVC는 임의 접근을 위하여 IDR (Instantaneous Decoding Refresh) 화면을 정의하고 있다. 만약 복호화기에서 IDR 화면이 입력되는 경우 복호화기의 참조 화면 버퍼를 비우게 된다. 하지만 GOP 내에 IDR보다 화면 디스플레이 순서가 앞선 화면간 예측 부호화된 화면의 경우, IDR에 의하여 참조 화면 버퍼가 비워졌기 때문에 선택이 더 제약된 참조 화면만을 사용해야 하므로 부호화 성능이 저하될 수 있다. 따라서 AVC에서 임의 접근을 위해 IDR 화면만을 정의했던 것에 비해, HEVC에서는 임의 접근을 위한 화면을 IRAP(Intra Random Access Point) 화면으로 정의하고, IDR 화면(NAL 타입 19, 20번), CRA(Clean Random Access, NAL 타입 21번) 화면, BLA(Broken Link Access, NAL 타입 16, 17, 18) 화면으로 나누어 세부적으로 정의한다^[13]. 또한, IRAP 화면 주변에서, 화면 재생 순서는 앞에 있지만 복호화 순서가 뒤에 있는 화면을 LP(Leading Picture)라 하고 LP 중에서 부호화가 가능한 화면을 RADL(Random Access Decodable Leading, NAL 타입 6, 7번) 화면으로, 부호화가 불가능한 화면을 RASL(Random Access

Skipped Leading, NAL 타입 8, 9번) 화면으로 정의한다.

AVC의 IDR 화면과 동일하게, HEVC의 IDR 화면은 복호화 순서상 IDR 화면이 나오는 경우 뒤에 복호화되는 화면이 IDR 화면 이전 화면을 참조할 수 없도록 한다. 하지만, HEVC에서의 새롭게 정의된 CRA 화면

HEVC에서는 복호화의 병렬처리를 위하여 AVC에서 지원되었던 슬라이스 선택스와 새롭게 채택된 타일(tile)과 WPP(Wavefront Parallel Processing)을 지원한다.

은 복호화 순서상 CRA 보다 이후에 있는 LP들이 CRA 화면 이전에 있는 화면을 참조하는 것이 허용되는데 이때, 복호화가 불가능한 RASL 화면과 복호화가 가능한 RADL 화면이 발생하게 된다. 또한, HEVC는 두 개의 비

트스트림을 이어 붙이기(splice) 위한 BLA 화면을 지원하며, 반드시 결합되는 비트스트림은 IRAP 화면으로 시작되어야 한다^[13]. BLA 화면은 이어 붙여진 비트스트림 내에 IRAP 화면에 해당하는 LP가 없는 경우 BLA_N_LP(NAL 타입 18)로 정의되고, RADL 화면만 존재 하는 경우에는 BLA_W_RADL(NAL 타입 17)로, RADL과 RASL 화면 모두 존재하는 경우에는 BLA_W_LP(NAL 타입 16)로 정의된다^[13].

IV. 결론

본고에서는 차세대 비디오 부호화 규격인 HEVC의 기본 구성 요소들에 대해 소개 하였다. HEVC는 기존의 비디오 압축 코덱들과 비교하여 압축 성능을 크게 향상시켰으며, 향후 도래하는 UHD TV 방송이나 3D 비디오 부호화 응용에 널리 활용될 것으로 예상된다. 뿐만 아니라, 스마트폰 등 모바일 응용분야와 IPTV 등 스트리밍 분야에도 넓은 응용 분야를 확보할 것으로 예상된다. 본고의 Part II에서는 HEVC 표준에 새롭게 채택된 각 요소 기술의 특징 및 부호화 성능에 서술한다.

참고 문헌

[1] G. J. Sullivan, et al., "Overview of the High Efficiency Video Coding (HEVC) Standard," IEEE Trans. Circuits

Syst. Video Technol., vol. 22, No. 12, pp. 1649–1668. Dec. 2012.

[2] I. K. Kim, et al., “Block Partitioning Structure in the HEVC Standard,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, No. 12, pp. 1697–1706. Dec. 2012.

[3] T. Wiegand, et al., “Overview of the H.264/AVC video coding standard,” IEEE Trans. Circuits Syst. Video Technol., vol. 13, No. 7, pp. 560–576. Jun. 2003

[4] J. Lainema, et al., “Intra Coding of the HEVC Standard,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, No. 12, pp. 1792–1801. Dec. 2012.

[5] B. Bross, et al., “High Efficiency Video Coding (HEVC) test specification draft 10,” document JCT-VC-L1003, Geneva, CH, Jan. 2013.

[6] J. L. Lin, et al., “Motion Vector Coding in the HEVC Standard,” IEEE Trans. Signal Processing, vol. 7, No. 6, pp 957–968, Dec. 2013.

[7] D. Marpe, et al., “Context-adaptive binary arithmetic coding in the H.264/AVC video compression standard,” IEEE Trans. Circuits Syst. Video Technol, vol. 13, no. 7, pp. 620–636, Jul. 2003.

[8] V. Sze, and M. Budagavi, “High Throughput CABAC Entropy Coding in HEVC,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, No. 12, pp. 1778–1791. Dec. 2012.

[9] J. Sole, et al., “Transform Coefficient Coding in HEVC,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, No. 12, pp. 1765–1777. Dec. 2012.

[10] N. Andrey, et al., “HEVC Deblocking Filter,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, No. 12, pp. 1746–1754. Dec. 2012.

[11] F. Chih-Ming, et al., “Sample Adaptive Offset in the HEVC Standard,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, No. 12, pp. 1755–1764. Dec. 2012.

[12] Dual HEVC Bitstream Analyzer,
[http://www.hanulsoftware.com/site/product-](http://www.hanulsoftware.com/site/product-dhba.php)

dhba.php

[13] 조현오, 심동규, “고효율 영상 부호화 기술 HEVC 표준 기술의 이해”.



김재일

- 2006년 2월 한국기술교육대학교 정보통신공학과 전자공학 전공 학사
- 2006년 3월~현재 한국과학기술원 정보통신공학과 석박사 통합과정

〈관심분야〉
 비디오 부호화, 인지 비디오 부호화 기술, 패턴 인식, 방송시스템



안상수

- 2006년 8월 충남대학교 전기전자전파전공 학사
- 2007년 3월~현재 한국과학기술원 정보통신공학과 석박사 통합과정

〈관심분야〉
 비디오 부호화, 율-왜곡값 예측, 패턴 인식



김대은

- 2011년 2월 고려대학교 전기전자전파공학부 학사
- 2014년 2월 한국과학기술원 전기 및 전자공학과 석사
- 2014년 3월~현재 한국과학기술원 전기 및 전자공학화 박사과정

〈관심분야〉

비디오 부호화, 인지 비디오 부호화



김문철

- 1989년 2월 경북대학교 전자공학과 학사
- 1992년 12월
University of Florida, USA, Electrical & Computer Engineering 석사
- 1996년 8월
University of Florida, USA, Electrical & Computer Engineering 박사
- 1997년 1월~2001년 2월
한국전자통신연구원 방송미디어연구부 선임연구원
- 2001년 2월~2008년 2월
한국정보통신대학교 공학부 조교수/부교수
- 2009년 3월~현재 한국과학기술원 전기및전자공학과 부교수/정교수

〈관심분야〉

2D/3D (인지) 비디오 부호화, 통계적 기계 학습, 패턴인식, 영상 분석 및 이해