

+ 호요성 · 광주과학기술원(GIST) 교수

TECH & TREND

> UHD 고화질 영상 서비스와 HEVC 영상 압축 표준 ③

1. 계층적 부호화 구조

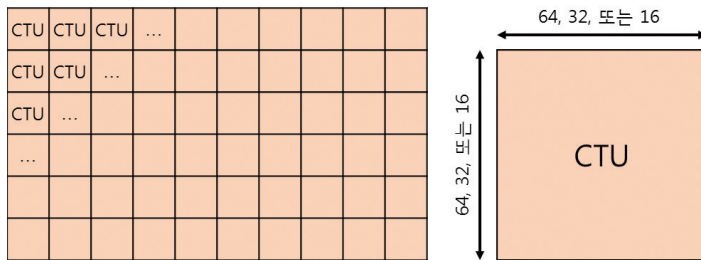
10년 전과 달리 다양한 전자기기의 발달로 4K UHD(Ultra High Definition)급 영상의 획득이 가능해졌으며, 이제 사람들은 8K UHD 영상을 논하고 있다. 모바일 기기에서도 HD(High Definition)급보다 해상도가 큰 2048×1530급 영상을 다룰 수 있게 되었다. HEVC(High Efficiency Video Coding)[1]는 HD 모바일, 홈 시네마, UHDTV(Ultra High Definition TV)와 같은 고화질 응용분야에서의 부호화 성능 향상을 목표로 개발된 차세대 영상 압축 표준이다. 최근 HEVC 표준이 국제 표준으로 승인됨에 따라 UHD 고화질 영상 서비스에 대한 희망이 한층 더 부풀어 오르고 있다.

HEVC 표준의 주요 기술은 HEVC FDIS(Final Draft International Standard)[2]에 자세히 설명되어 있으며, JCT-VC(Joint Collaborative Team on Video Coding) 문서 관리 시스템[3]에서 그 간의 기술 개발 상황과 새롭게 진행되는 이슈를 살펴볼 수 있다. 하지만 HEVC를 처음 접하는 이들이 이러한 표준 문서들을 곧바로 살펴보기는 힘들기 때문에 이번 호에서는 지난 호에서 간략히 소개한 HEVC의 주요 기술을 그림을 곁들여 이해하기 쉽게 설명한다.

먼저 HEVC의 가장 큰 특징인 계층적 부호화 구조에 대해서 알아본다. HEVC의 CTU(Coding Tree Unit), CU(Coding Unit), CTB(Coding Tree Block), CB(Coding Block), PB(Prediction Block), TB(Transform Block)는 기존의 표준에서 매크로블록(Macro block)과 블록에 해당하는 개념이다. 큰 해상도의 영상을 효율적으로 부호화하기 위해서는 좀 더 큰 부호화 단위가 필요하다. 여기서 문제가 되는 것은 영상의 작은 디테일 또한 무시할 수 없으므로 4×4 크기와 같은 작은 크기의 블록 역시 필요하다는 점이다.

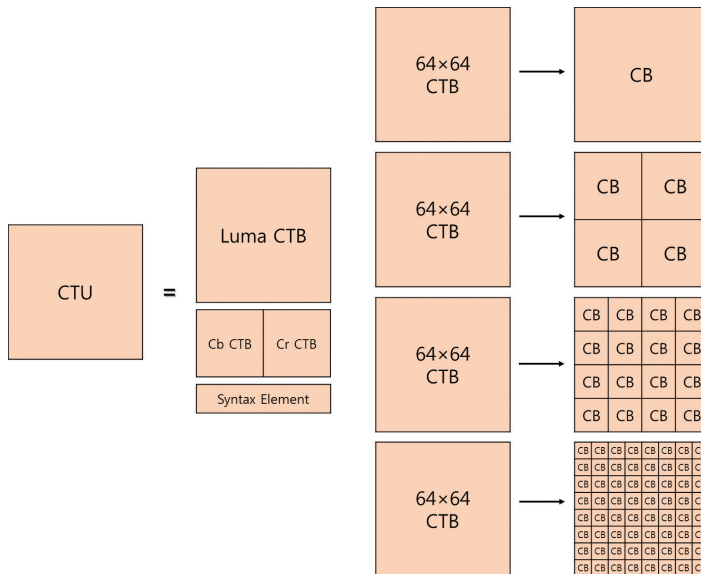
이러한 문제점을 해결하기 위해 HEVC는 유연하고 효과적인 계층적 부호화 구조를 사용한다. 부호화의 첫 단계로 한 장의 영상, 즉 프레임은 [그림 1]과 같이 다수의 CTU로 분할된다. CTU의 가로와 세로 정보는 SPS(Sequence Parameter Set)를 통해 복호기로 전달되며 전체 영상에 대해 모든 CTU는 동일한 크기를 가진다. HEVC에서 사용 가능한 CTU의 크기는 64×64, 32×32, 또는 16×16이다.

CTB를 설명하기에 앞서 표준 문서 내의 명칭에 대한 규정을 알아보자. 표준 문서의 계층적 부호화 구조 설명에서 사용하는 여러 용어 중 Unit이라 지칭하는 것들과 Block이라 지칭하는 것들이 있다. CTU 또는 CU와 같이 Unit이라 지칭하는 것들은 부호화 논리 단위(logical unit)로, 실제 비트스트림으로 부호화되는 정보이다. 반면에, CTB, CB, PB, 또는 TB와 같이 Block이라 지칭하는 것들은 비디오 프레임 버퍼를 의미한다.

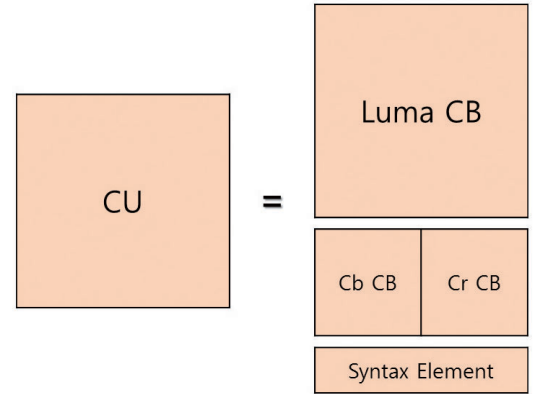


[그림 1. CTU(Coding Tree Unit)의 개념]

논리 단위 중 하나인 CTU는 [그림 2]와 같이 총 세 개의 CTB와 관련 구문 요소(Syntax Elements)로 나뉜다. 휘도(Luma) CTB는 16×16에서 64×64의 CTU와 동일한 크기를 가지며, 색채(Cb와 Cr) CTB는 색차 포맷에 따라 다른 크기를 가진다. CTB는 화면내 예측을 수행할지 화면간 예측을 수행할지를 결정하기에는 너무나 크다. 따라서 각 CTB는 다수의 CB들로 분할될 수 있으며, 사용 가능한 CB의 크기는 CTB의 크기에서부터 8×8까지 다양하다.



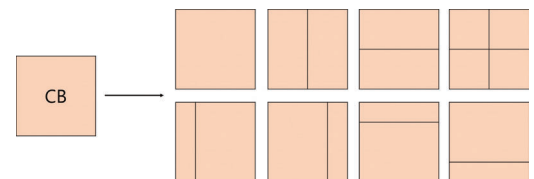
[그림 2. CTU, CTB(Coding Tree Block), CB(Coding Block)의 관계]



[그림 3. CU와 CB의 관계]

[그림 2]는 64×64 CTB가 다수의 CB로 분할되는 방법을 보여준다. CB는 예측 트리와 변환 트리의 루트가 되는데, 화면내 예측을 수행할지 화면간 예측을 수행할지를 알려주는 예측 유형(prediction type) 정보는 CU 레벨에서 부호화된다. CU는 [그림 3]과 같이 세 개의 CB와 관련 구문 요소로 구성된다.

CB는 예측 유형을 결정하기에는 충분하지만, 화면간 예측에서의 움직임 벡터 또는 화면내 예측 모드를 결정하기에는 너무 크다. 예를 들어 8×8 CB의 중간 부분에 눈과 같은 매우 작은 물체가 움직이고 있다면 8×8 CB에 대해 하나의 움직임 벡터(Motion Vector, MV)를 사용하는 것보다 CB를 좀 더 작게 나누어 각 부분에 대해 서로 다른 움직임 벡터를 사용하는 것이 더 나을 것이다. 그 결과로 사용되는 것이 바로 PB이다. 임의의 CB는 시/공간적 예측 가능성에 따라 PB로 분할될 수 있다. [그림 4]는 사용 가능한 PB 분할을 보여준다.



[그림 4. PB(Prediction Block) 분할]

예측 후의 변환 부호화에서 사용하는 기본 단위는 TB이다. 앞에서 설명한 바와 같이, CB는 복잡한 고주파 부분

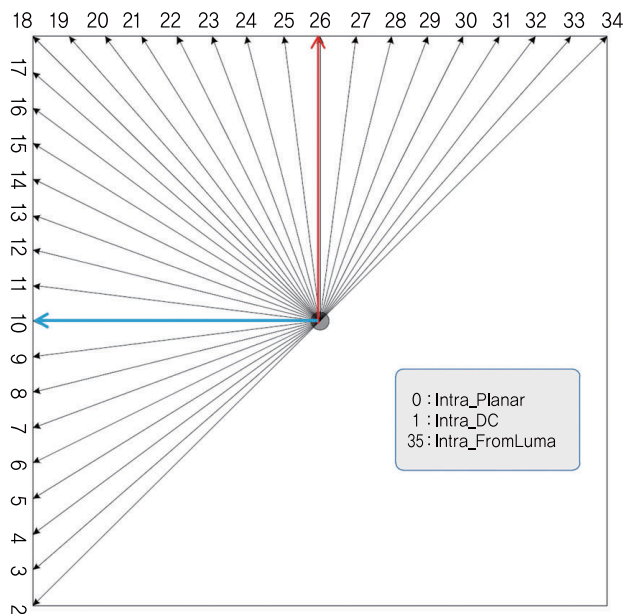
과 편평한 저주파 부분을 모두 포함할 수 있기 때문에 CB 단위로 변환을 수행하는 것은 적합하지 않다. 따라서 임의의 CB는 더 작은 변환 블록인 TB들로 분할된다. HEVC에서는 임의의 CB에 대해, TB와 PB가 독립적으로 결정되므로 PB보다 큰 TB를 사용할 수 있다. [그림 5]는 하나의 CB가 TB로 분할되는 예를 보여준다.



[그림 5. TB(Transform Block) 분할]

2. 화면내 부호화

인트라 예측은 하나의 화면 내에서 인접한 화소간의 상관도를 이용해 예측을 수행하는 방법이다. 일반적으로 참조할 수 있는 데이터의 양이 적어 화면간 예측보다 예측 효율이 높지 않지만, 독립적으로 부호화가 수행되고, 부호화 속도가 빠르다는 장점이 있다. H.264/MPEG-4 AVC에서 최대 9개의 예측 모드를 사용한 것과는 달리, HEVC 화면내 부호화에서는 [그림 6]과 같이 DC 및 Planar 모드와 33가지 방향성을 갖는 예측 모드들을 사용한다 [4].



[그림 6. 화면내 예측 모드[2]]

현재 블록 주변의 경계 화소들을 예측을 위한 참조 화소로 이용하며, 경계 화소들은 이전에 부호화되고 복호화되어 사용 가능하다. 다양한 방향성을 갖는 예측 신호를 얻기 위해 우선 주변 참조 화소들을 평활화(smoothing)한 후, 필터링 된 참조 화소를 이용해 보간(interpolation)을 수행한다.

휘도 블록에서는 현재 블록의 크기에 상관없이 35개의 예측 모드를 모두 사용하고, 색차 블록에서는 Planar, 수직, 수평, DC 예측 모드, 그리고 DM_Chroma_idx 모드를 추가로 사용한다. DM_Chroma_idx 모드는 휘도 블록의 예측 모드를 색차 블록에서 그대로 사용하는 모드로 휘도 성분과 색차 성분 간의 값 특성이 유사한 경우 부호화 효율을 얻을 수 있다.

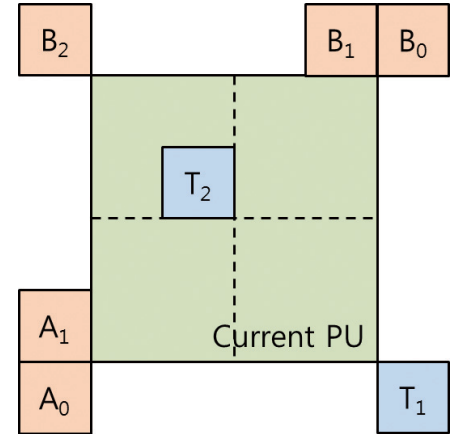
3. 화면간 부호화

HEVC 화면간 움직임 예측의 주요 특징은 AMVP(Advanced Motion Vector Prediction) 기술과 Merge 기술이다[5]. H.264/MPEG-4 AVC에서는 주변 블록들의 움직임 벡터들의 중간값(median)을 이용해 하나의 움직임 벡터 예측값을 생성하고, 실제 움직임 벡터와 이 예측값의 차를 부호화한다.

HEVC에서는 다양한 움직임 벡터 예측값들을 경쟁시켜 현재 움직임 벡터와 최소의 차이를 갖는 움직임 벡터 예측값을 이용하는 AMVP 기술을 사용한다. AMVP는 두 개의 움직임 벡터 예측 후보를 사용한다. 먼저 두 개의 공간적 움직임 벡터 예측 후보를 탐색한 후, 그 중 하나 이상의 후보를 사용할 수 없거나 두 개의 후보가 중복되는 경우, 하나의 시간적 움직임 예측 후보를 탐색한다.

[그림 7]은 공간적 및 시간적 움직임 벡터 예측 후보들을 보여준다. 후보들은 [그림 7]과 같이 해당 블록의 위치에 따라 결정된다. AMVP의 공간적 움직임 벡터 예측 후보는 [그림 7]의 {A0, A1} 중에서 하나의 후보를, {B0, B1, B2} 중에서 하나의 후보를 선정한다. 만약 해당 블록이 현재 슬라이스를 벗어나거나 화면내 부호화된 블록일 경우에는 후보가 될 수 없다. 시간적 후보로는 [그림 7]과 같이 참조 화면의 collocated PU의 오른쪽 대각선 아래에 위치한 T1 블록을 사용한다. 이 블록이 사용 가능하지 않을 경우, collocated PU의 중간에 위치한 T2 블록을 사용한다.

현재 움직임 벡터와 예측 움직임 벡터의 차이가 0인 경우에는 Merge 모드를 사용한다. Merge 모드는 현재 PB에 대한 움직임 정보를 복호기로 전송하지 않고, 공간적 또는 시간적으로 인접한 블록들로부터 산출하는 기술이다. Merge 모드가 사용되는 경우, 사용 여부를 알리는 플래그와 어떠한 움직임 벡터 예측 후보가 사용되었는지를 알리는 인덱스 정보가 전송된다. Merge 모드를 사용하면 AMVP의 경우보다 전송해야 하는 비트량이 감소하므로 부호화 성능이 향상된다. Merge 모드에서의 공간적 움직임 벡터 예측 후보 탐색 순서는 {A1, B1, B0, A0, B2}이다. 시간적 움직임 벡터 예측 후보 탐색 방법은 AMVP의 경우와 동일하다.



[그림 7. 화면간 예측을 위한 후보군]

4. 루프내 필터 기술

HEVC는 보다 높은 부호화 효율을 위해 디블록킹 필터[6]와 SAO(Sample Adaptive Offset)[7] 두 가지 루프 내 필터링(in-loop filtering)을 사용한다. 루프 내 필터링은 복호화 과정 내에 필터 과정이 포함된 것을 의미하며, 필터가 적용된 영상을 참조 영상으로 활용하게 되어 부호화 효율도 달성할 수 있다. 디블록킹 필터는 예측 및 양자화로 인해 발생하는 블록 경계의 왜곡을 효과적으로 제거함으로써 복호 영상의 주관적 화질을 향상시키는 기술이다.

[그림 8]은 영상을 부호화할 때 디블록킹 필터를 사용하지 않은 경우와 사용한 경우의 복호 영상의 화질을 비교한 것이다. 디블록킹 필터를 사용하지 않은 경우에는 블록 왜곡(blocking artifacts)이 눈에 띄지만, 디블록킹 필터를 사용함으로써 이를 다소 완화할 수 있음을 알 수 있다. HEVC의 디블록킹 필터는 H.264/MPEG-4 AVC의 디블록킹 필터와 상당히 유사하지만, 복잡한 연산 과정을 간략화한 기술이다.



(a) 디블록킹 필터를 사용하지 않은 경우

(b) 디블록킹 필터를 사용한 경우

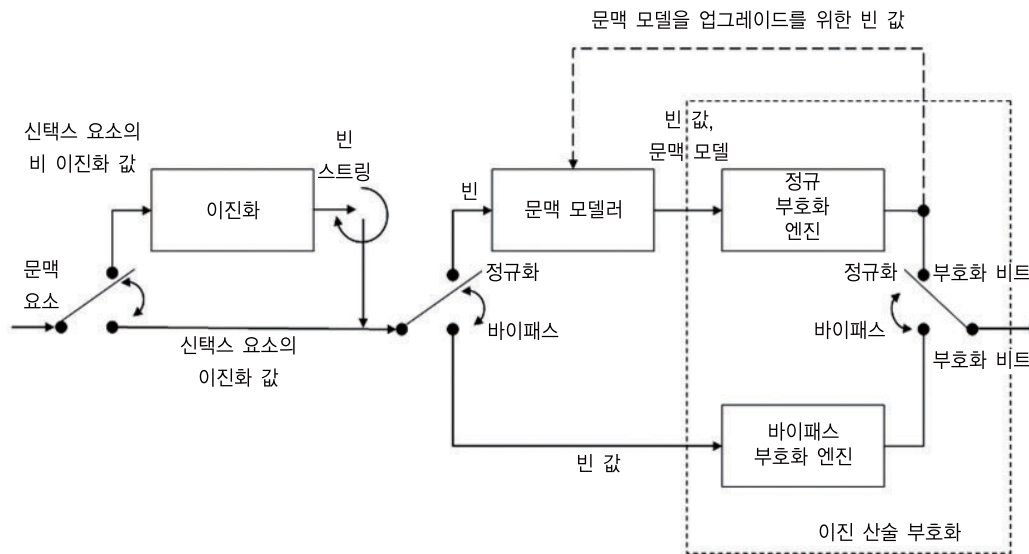
[그림 8. 디블록킹 필터를 통한 블록 왜곡 (blocking artifacts) 완화[6]]

SAO는 기존의 비디오 압축 표준에는 존재하지 않던 기술로 HEVC에서 새롭게 선보이는 인루프 필터링 방법이다. SAO 기술은 양자화 과정을 통해 발생하는 원본 영상과 복호 영상 간의 왜곡을 완화시키기 위한 것으로 주관적 화질 및 부호

화 효율을 함께 향상시키는 기술이다. 양자화는 변환의 출력인 변환 계수(transform coefficient)에 대해 수행되므로, 주파수 도메인에서 양자화가 이루어진다. 따라서 양자화로 인해 물체 가장자리에 발생하는 링잉 현상(ringing artifacts) 또는 화소값이 원본값에 비해 일정한 크기만큼 커지거나 작아지는 현상이 발생한다. SAO 기술은 일정 화소 집합에 화소 단위의 오프셋(offset)을 더해 전체적인 화소 간의 차이를 줄임으로써 양자화 에러를 보상한다. 이때 왜곡의 정도가 다른 화소를 구분해 서로 다른 오프셋을 적용하는 적응적인 보상 방법을 사용함으로써, 영상의 특성에 따라 부분적으로 다르게 발생하는 왜곡을 효율적으로 최소화할 수 있도록 한다. 이 기술은 복잡도 대비 압축 효율이 높다는 장점을 갖고 있다.

5. 엔트로피 부호화 기술

H.264/MPEG-4 AVC에서 두 개의 엔트로피 부호화 기술을 사용했던 것과 달리, HEVC는 문맥 기반 적응적 이진 산술 부호화(Context-based Adaptive Binary Arithmetic Coding, CABAC)[8][9]를 하나의 엔트로피 부호화 기술로 사용한다. CABAC은 실험을 통해 얻은 확률로 부호화해야 할 각 구문요소들의 문맥 모델(context model)을 만들고, 이를 이용해 엔트로피 부호화를 수행한다. CABAC은 [그림 9]와 같이 크게 이진화, 문맥 모델링, 이진 산술 부호화, 확률 업데이트 부분으로 구성된다.



[그림 9. 문맥 기반 적응적 이진 산술 부호화]

이진화 단계는 이진값이 아닌 선택스 요소를 이진화된 시퀀스로 매핑하는 과정이다. 만약 이진값을 갖는 선택스 요소가 입력 신호로 주어지면, 첫 번째 단계인 이진화 과정은 생략된다. 이진화된 시퀀스의 각 이진 값은 문맥 모델링 단계로 들어간다. 여기서 이진값은 이진화 과정에서 생성된 이진 시퀀스의 각각의 비트를 의미한다. 문맥 모델링 단계에서는 현재 이진값에 대응하는 확률 모델을 이전에 부호화된 선택스 요소, 혹은 이진값에 따라 선택한다. 확률 모델

을 결정한 후, 결정된 확률 모델과 주어진 이진값을 이용하여 이진 산술 부호화를 수행한다. 마지막으로 선택된 확률 모델을 업데이트한다.

6. 결론



이번 호에서는 HEVC(High Efficiency Video Coding) 표준의 핵심 기술을 자세히 살펴보았다. HEVC는 기존의 H.264/MPEG-4 AVC와 마찬가지로 블록 기반 하이브리드 부호화(hybrid coding) 구조를 띄고 있지만, 각 요소 기술 단위로 많은 발전이 이루어져 전체적으로 뛰어난 압축 효율을 얻을 수 있다. 또한 HEVC 표준은 설

계 단계에서 부호화 효율뿐 아니라 병렬화를 고려했기 때문에 코덱의 복잡도가 이전의 H.264/MPEG-4 AVC 표준에 비해 크게 증가하지 않을 것으로 예상된다. 따라서 향후 UHDTV(Ultra High Definition TV), 모바일 고품질 영상 스트리밍, 고화질 3차원 영상 서비스 등 다양한 분야에 널리 이용될 것으로 판단된다.

참고문헌

- [1] 호요성, 최정아, “UHD 고화질 영상 압축 기술 : HEVC 알고리즘 이해와 프로그램 분석,” 진샘미디어, 2013.
- [2] ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/ WG11, “High Efficiency Video Coding (HEVC) Text Specification Draft 10 (for FDIS & Consent),” 12th JCT-VC Meeting, JCTVC-L1003, Jan. 2013.
- [3] JCT-VC Document Management System, <http://phenix.int-evry.fr/jct/>
- [4] J. Lainema, F. Bossen, W. Han, J. Min, and K. Ugur, “Intra coding of the HEVC standard,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1792–1801, Dec. 2012.
- [5] M. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand, “Block merging for quadtree-based partitioning in HEVC,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1720–1731, Dec. 2012.
- [6] A. Norkin, G. Bjøntegaard, A. Fuldseth, M. Narroschke, and M. Ikeda, “HEVC deblocking filter,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1746–1754, Dec. 2012.
- [7] C. Fu, E. Alshina, A. Alshin, Y. Huang, C. Chen, C. Tsai, C. Hsu, S. Lei, J. Park, and W. Han, “Sample adaptive offset in the HEVC standard,” IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1755–1764, Dec. 2012.
- [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] 최정아, 호요성, “HEVC 화면내 부호기를 위한 효율적인 변환 계수 부호화 방법,” 스마트미디어저널, 제1권, 제2호, pp. 06–11, 2012년 06월.