

일반논문 (Regular Paper)

방송공학회논문지 제17권 제6호, 2012년 11월 (JBE Vol. 17, No. 6, November 2012)

<http://dx.doi.org/10.5909/JBE.2012.17.6.1091>

ISSN 1226-7953(Print)

HEVC 부호화기 소프트웨어의 통계적 특성 및 복잡도 분석

안용조^{a)}, 황태진^{a)}, 유성은^{a)}, 한우진^{b)}, 심동규^{a)‡}

Statistical Characteristics and Complexity Analysis of HEVC Encoder Software

Yongjo Ahn^{a)}, Taejin Hwang^{a)}, Sungeun Yoo^{a)}, Woo-Jin Han^{b)}, and Donggyu Sim^{a)‡}

요약

본 논문에서는 현재 표준화가 진행 중인 HEVC (high efficiency video coding) 부호화기의 고속화, 최적화, 병렬화 연구에 앞서 통계적 특성 및 복잡도 분석을 수행하였다. HEVC는 H.264/AVC에 비해 약 2배의 압축 성능을 나타내지만 부호화기 복잡도는 크게 증가하여 이는 앞으로 해결해야 할 문제로 남아있다. HEVC의 높은 부호화기 복잡도를 해결하기 위한 고속화, 최적화, 병렬화 연구에 앞서, 본 논문에서는 HEVC 참조소프트웨어인 HM 7.1을 이용하여 HEVC 부호화기의 복잡도를 측정하였다. 추가적으로, 실제 응용에서 사용될 고속 HEVC 부호화기 소프트웨어에 대한 예상 복잡도를 고속 알고리즘이 적용된 HM 7.1 소프트웨어로 측정하였다. 복잡도 측정은 공통 실험 영상 및 조건을 사용하였으며 PC 환경에서 부호화기 소프트웨어의 동작 사이클을 측정하고 이를 분석하였다. 또한, 부호화를 통해 생성된 비트스트림을 이용하여 HEVC 부호화기 소프트웨어의 부호화 구조에 따른 통계적 특성과 제한적 부호화에 따른 통계적 특성에 대하여 제시하고 이를 분석한다.

Abstract

In this paper, we analyzed statistical characteristics and complexity of HEVC encoder as a leading research of acceleration, optimization and parallelization. Computational complexity of the HEVC encoder is approximately twice the compression performance compared to H.264/AVC. But, the increase of encoder complexity remains a problem to be solved in the future. Before performing the research on acceleration, optimization and parallelization to reduce high complexity of HEVC encoder, we measure the complexity each module for HEVC encoder using its reference software HM 7.1. We also measured the predicted complexity of fast HEVC encoder software, used in real applications, using HM 7.1 applying fast encoding method. The complexity is measured in terms of the operating cycle of the encoder software under the common test sequences and conditions in the Windows PC environment. In addition, we analyze statistical characteristics of HEVC encoder software according to encoding structures and limitation using coded bitstreams.

Keyword : HEVC, Encoder, Complexity, Statistical characteristic

a) 광운대학교 컴퓨터공학과 (Dept. of Computer Engineering, Kwangwoon University)

b) 가천대학교 소프트웨어 설계·경영학과 (Dept. of Software Development and Management, Gachon University)

‡ Corresponding Author : 심동규 (Donggyu Sim)

E-mail: dgsim@kw.ac.kr

Tel: +82-2-940-5470 Fax: +82-2-914-6471

※ 본 연구는 지식경제부 및 한국산업기술평가위원회의 산업원천기술개발사업(정보통신)의 일환으로 수행하였음.[10039199, 인지품질 기반 스케일러블 3D 비디오 코덱 핵심 기술 연구]

· Manuscript received October 11, 2012 Revised November 26, 2012 Accepted November 26, 2012

1. 서론

디지털 방송의 보편화와 다양한 멀티미디어 기기들의 등장으로 인해 고해상도, 고품질 영상에 대한 소비자들의 요구가 점차 증가하고 있다. 이러한 요구에 부응하여 H.264/AVC의 표준화를 진행한 ITU-T VCEG과 ISO/IEC MPEG은 JCT-VC를 결성하고, 차세대 비디오 부호화 기술인 HEVC 표준화를 진행하고 있다. HEVC는 2010년 1월 Call for proposal을 받고 공식적인 표준화를 시작하여, 2012년 10월까지 총 11차례의 회의를 개최하였다. 지금까지도 높은 부호화 성능을 보이는 새로운 비디오 표준을 위한 부호화 방법이 논의되고 있으며, 2013년 1월 12차 회의를 앞두고 있다.

HEVC는 기존의 비디오 코덱 표준인 H.264/AVC, MPEG-2/4와 동일한 블록기반의 하이브리드 비디오 부호화 구조를 가지면서 기존의 비디오 코덱 표준에 비해 부호화 성능을 향상시키기 위한 추가 기술들이 채택되었다. 먼저, 매크로블록 개념에서 세분화된 CU (coding unit), PU (prediction unit), TU (transform unit) 3가지 종류의 부호화 단위 적용과 계층적 쿼드-트리 (quad-tree) 구조의 부호화 수행과 같은 구조적 특징을 예로 들 수 있다. 또한, 64×64부터 8×8까지 다양한 크기의 부호화 단위를 사용하며, 32×32부터 4×4까지 다양한 크기의 변환을 수행한다. 이외에도 화면 내 예측 방향의 증가, 향상된 움직임 예측 결정 기법, 움직임 벡터 병합 및 세분화된 루프필터를 적용하여 H.264/AVC 대비 약 2배의 압축 성능을 보인다. 표 1은 HEVC, H.264/AVC 두 비디오 압축 표준의 참조소프트웨어인 JM 18.3과 HM 6.0간 비트율과 객관적 화질에 대한 성능 비교이다^[1]. 표 1에서는 H.264/AVC 대비 24.3-44.2%의 비트율이 감소하는 것으로 나타난다. 다양한 부호화 구조와 향상된 예측기법에 따라 표 1과 같은 부호화 성능향상이 있으나, 이에 따른 연산 복잡도가 H.264/AVC에 비해 크게 증가하였다^{[2][4]}. 이러한 HEVC 부호화기의 연산 복잡도 증가를 해결하기 위하여 조기종료 (early termination) 알고리즘 및 고속 모드 결정 (fast mode decision) 알고리즘과 같은 고속화 기술들이 기고되었다^{[5][7]}. 하지만, 현재까지 공개된 연구 결과로는 실시간 부호화에 어려움이 있기 때

문에 HEVC 상용화를 위해서는 최적화, 고속화 및 병렬화 연구가 더욱 활발히 진행되어야 할 것이다. 본 논문에서는 이러한 실시간 연구 개발에 앞서, 반드시 선행되어야 할 HEVC 부호화기 복잡도 분석 하고자 한다. 부호화기 복잡도 분석을 위해 HEVC 참조소프트웨어인 HM 7.1^[8]을 이용하여 PC환경에서 동작 사이클을 측정하였다. 또한, HEVC 부호화기의 다양한 통계적 특성을 분석함으로써 HEVC 부호화기에 대한 추가적인 연구들을 위한 기초자료를 제시하고자 한다.

본 논문의 구성은 다음과 같다. II장에서는 HEVC 부호화기의 전체적인 구성 및 현재까지 채택된 세부 기술에 대하여 간략하게 살펴보고, III장에서는 HEVC 참조 소프트웨어인 HM 7.1 부호화기 소프트웨어의 복잡도를 5개의 기능 모듈로 분류하여 제시한다. 추가적으로 실제 응용에서 사용될 것으로 예상되는 고속 HEVC 부호화기의 복잡도를 HM 소프트웨어에 채택된 고속 알고리즘을 HM 7.1에 적용하여 분석한다. 또한, IV장에서는 HM 7.1 부호화기 소프트웨어를 기반으로 HEVC 부호화의 다양한 통계적 특성을 제시하고, 툴 on-off 실험을 통해 부호화 기술의 부호화 속도 대비 부호화 성능을 살펴본다. 마지막으로, V장에서는 결론을 맺는다.

표 1. JM 18.3과 HM 6.0간 비트율과 PSNR 비교
Table 1. Bitrate and PSNR comparison between JM 18.3 and HM 6.0

Class	Sequence	BD-Bitrate (%)	BD-PSNR (dB)
B	Kimono	-44.20	1.87
	ParkScene	-31.12	1.23
	Cactus	-34.13	0.95
	BasketballDrive	-40.59	1.23
	BQTerrace	-41.66	0.77
C	BasketballDrill	-33.20	1.72
	BQMall	-30.26	1.44
	PartyScene	-26.62	1.34
	RaceHorse	-24.26	1.07
Average of Class B		-38.35	1.21
Average of Class C		-28.58	1.39

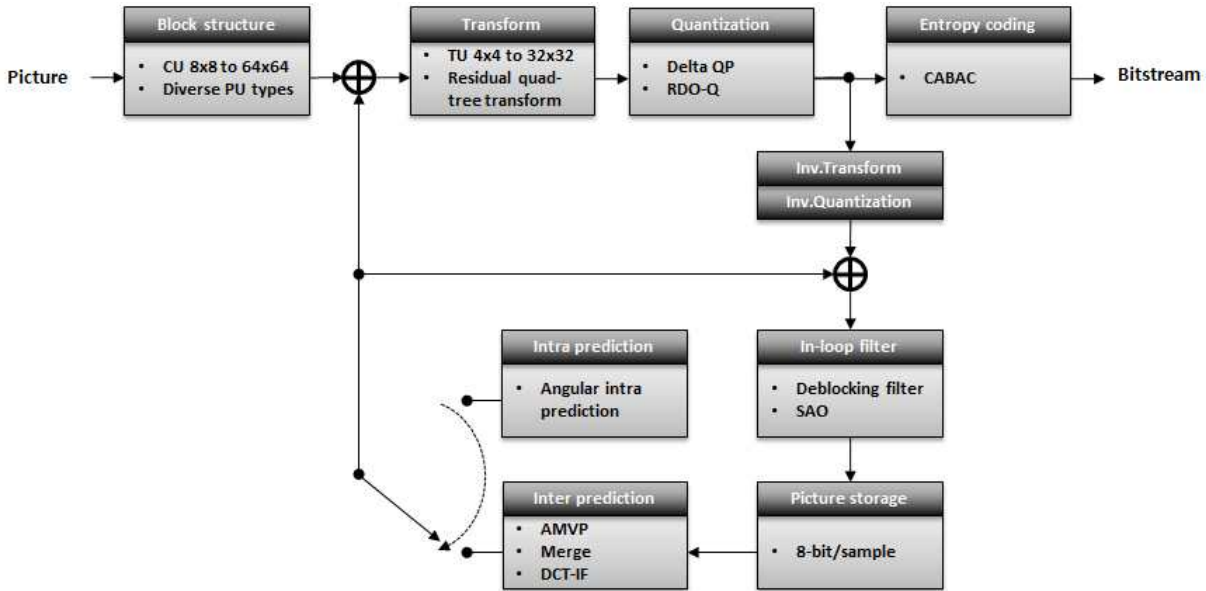


그림 1. HEVC 부호화기의 블록 다이어그램
 Fig. 1. Block diagram of HEVC encoder

II. Overview of HEVC encoder

1. HEVC 부호화기 구조

그림 1에 나타난 것처럼, HEVC는 MPEG-2/4, H.264/AVC 등의 기존 비디오 코덱들과 유사하게 공간적, 시간적 예측, 변환, 양자화 및 엔트로피 부호화 등을 특징으로 하는 하이브리드 부호화 구조를 채택하였다.

HEVC는 기존 코덱보다 효율적인 압축 성능을 확보하기 위해 다양한 부호화 구조를 지원한다. 가장 두드러진 특징으로, HEVC는 총 3가지 종류의 시간적 예측 구조를 이용한다. 부호화하는 모든 프레임이 IDR (instantaneous decoding refresh)을 사용하는 올 인트라 (AI, all intra) 구조, 시퀀스의 첫 장만을 IDR로 사용하며 단방향 예측을 수행하여 지연이 적게 발생하는 환경에 적합한 저 지연 (LD, low delay)구조, IDR 혹은 CRA (clean random access)를 일정 시간단위로 부호화하여 원하는 시간 위치로의 이동이 기술적으로 가능하도록 설계한 임의 접근 (RA, random access) 구조가 있다. 이러한 시간적 예측 구조 외에도 다양화된 화면 분할 구조를 지원하는 것 또한 HEVC의 큰 특징이다.

기존 비디오 코덱에서 사용되던 슬라이스 (slice) 뿐만 아니라 타일 (tile)이라는 직방형태의 화면 분할 구조를 지원한다. 타일은 기존 슬라이스와 비슷하게 인접 타일간의 참조가 불가능 하면서, 슬라이스에 비해 참조 가능 블록의 수가 적기 때문에 약 0.8-3.8%의 비트율 감소를 보이는 분할 구조이다⁹⁾. 또한, HEVC의 가장 큰 부호화 구조적 특징으로 MPEG-2/4, H.264/AVC에서 사용된 고정 크기의 16x16 매크로 블록보다 훨씬 다양한 크기를 가지면서 기능별로도 세분화된 CU, PU, TU 단위를 이용한다는 점을 들 수 있다. CU는 부호화가 이루어지는 기본단위로 64x64에서 8x8까지 쿼드-트리 구조로 구성된다. 이때, 쿼드-트리로 구성된 최상위 CU를 CTU (coding tree unit)라고 한다. PU는 화면 내, 화면 간 예측에 사용되는 단위로 각 PU는 참조 인덱스, 움직임 벡터, 화면 내 예측 모드 등의 정보를 포함한다. 각 PU는 최대 2Nx2N, 2Nxn, Nx2N, NxN, nLx2N, nRx2N, 2NxnU, 2NxnD 8가지의 모양이 있다. TU는 변환이 이루어지는 단위로 32x32에서 4x4까지 쿼드-트리 구조로 구성된다. 하나의 CU는 여러 개의 TU로 분할 될 수 있으며, PU와 독립적으로 분할 될 수 있다. 그림 2는 HEVC의 부호화 블록 구조인 CU, PU, TU의 예를 나타낸다.

2. HEVC 부호화기의 기능 모듈

HEVC 부호화기를 기능 모듈로 분류하면 크게 화면 내 예측 (intra prediction), 화면 간 예측 (inter prediction), 변환-양자화 (transform-quantization), 루프 필터 (loop filter), 엔트로피 부호화 (entropy coding) 5가지로 나눌 수 있다. 본 절의 각 항에서는 전술한 5가지 기능별 모듈에 대하여 현재 HEVC에 채택된 부호화 기술에 대하여 간략하게 소개한다.

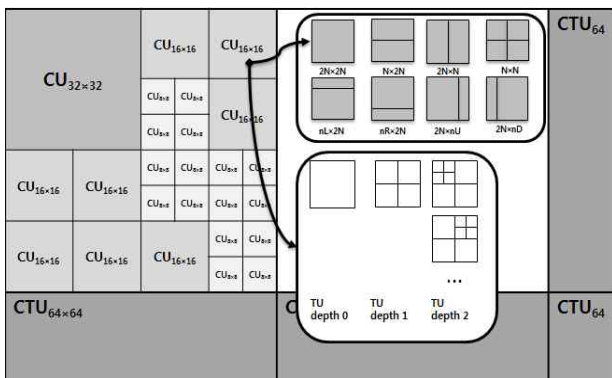


그림 2. HEVC의 부호화 블록 구조 (CU, PU, TU)
Fig. 2. Structure of coding block (CU, PU, TU) in HEVC

2.1 화면 내 예측

그림 3은 HEVC의 화면 내 예측 방향 및 모드를 나타낸다^[10]. HEVC의 화면 내 예측은 H.264/AVC에서 8개 방향을 포함한 9개의 모드를 사용하던 것과 달리, 33개 방향에 33개 모드를 사용하여 기존의 화면 내 예측 방법 보다 훨씬 더 세분화된 각도의 예측이 가능하도록 설계되었다. 방향성의 증가와 더불어 균등한 영역에서 주관적 화질을 높이기 위한 Intra_Planar 모드와 같은 새로운 모드가 부호화 효율 향상을 위해 추가되었다^[11]. 이러한 방향성 및 모드 추가 이외에도 유효하지 않은 화소를 위해 인접 화소를 이용하는 패딩 (reference sample padding) 기술 및 예측 화소에 대한 저역 필터 (low-pass filter) 기술이 추가되었다. 예측 화소에 대한 전처리 기술은 H.264/AVC에서는 8x8 블록에서만 적용되었으나, HEVC에서는 모든 크기의 블록에 대해 적용하고 있다. 또한, DC, 수직, 수평 모드에 대하여 경계

화소 스무딩 (boundary value smoothing) 기술이 추가되었다.

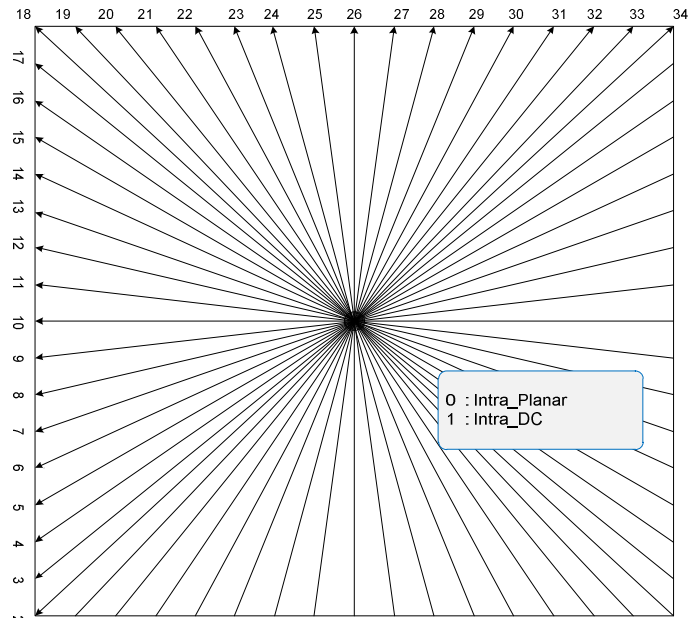


그림 3. 화면 내 예측 방향 및 모드^[10]
Fig. 3. Intra prediction mode directions

표 2. HEVC DCT-IF의 필터 계수
Table 2. Filter coefficients of HEVC DCT-IF

Component	α	Filter(α)
Luma	1/4	{-1, 4, -10, 58, 17, -5, 1, 0}
	1/2	{-1, 4, -11, 40, 40, -11, 4, -1}
Chroma	1/8	{-2, 58, 10, 2}
	3/8	{-6, 46, 28, -4}
	1/4	{-4, 54, 16, -2}
	1/2	{-4, 36, 36, -4}

2.2 화면 간 예측

HEVC의 화면 간 예측에는 AMVP (advanced motion vector prediction), 병합 (merge) 모드와 DCT-IF (discrete cosine transform based interpolation filter)가 이용된다. HEVC AMVP는 KTA에 제안된 MVC (motion vector competition)처럼 여러 개의 MVP (motion vector predictor) 후보를 사용하여, 주변 움직임 벡터의 중간 값을 사용하던

H.264/AVC에 비해 MVD (motion vector difference)를 효과적으로 줄일 수 있는 기법이다. 병합 모드는 주변 블록의 움직임 정보를 그대로 사용하는 대신, 주변 블록과의 병합 유무에 대한 플래그와 병합 후보의 인덱스를 부호화하는 기술이다. 병합 기술의 추가에 따라, 기존 비디오 코덱에서 사용하던 skip-run을 이용한 Skip 모드 대신, CU 단위로 플래그를 직접 명시해주는 방법을 사용하여 Skip 모드 부호화가 수행된다. AMVP와 병합 모드는 예측 후보의 인덱스를 부호화한다는 공통점이 있으나, AMVP의 경우 2개의 후보, 병합 모드는 최대 5개의 후보를 가진다는 점과 후보 선택과정에 있어 차이가 있다. AMVP, 병합 모드와 같은 예측 후보 결정 기술과 더불어 움직임 예측 및 보상의 정밀도를 높이기 위한 보간 필터에도 DCT-IF가 새롭게 채택되었다. H.264/AVC에서는 고정 6-tap 필터를 사용하였으나, HEVC에서는 휘도 성분에서는 8-tap (7-tap) DCT-IF, 색차 성분에서는 4-tap DCT-IF를 사용한다. DCT-IF란, DCT 과정에서 분수 화소 성분의 위치 값을 복원하는 계수를 구할 수 있으며, 이 계수 값들을 이용하여 보간을 수행하는 기술이다^[12]. H.264/AVC에서 1/4화소를 보간함에 있어 1/2화소를 이용하여 양선형 (bi-linear) 필터를 적용하는 방식과 달리, HEVC에서는 1/4화소에 대한 직접 필터 연산을 수행한다. 또한, 필터 계수의 개수가 증가하여 보다 정확한 보간은 가능해졌으나, 이에 따른 계산 복잡도의 증가는 실시간 부호화에 있어 큰 문제점으로 지적되고 있다. 표 2는 HEVC DCT-IF의 필터 계수를 나타낸다.

2.3 변환-양자화

HEVC는 기존의 비디오 코덱과 달리 변환 과정만을 위한 단위인 TU를 별도로 정의하며, 변환을 위한 커널의 크기는 4×4, 8×8, 16×16, 32×32로 다양하게 지원한다. TU의 구조는 쿼드-트리로 구성되며, PU와는 독립적으로 분할될 수 있으나 TU가 속한 CU의 크기를 초과할 수 없다. HM 2.0까지는 4×4, 8×8 변환은 H.264/AVC와 동일한 변환 커널을 사용하였고, 16×16, 32×32 변환은 Chen 알고리즘 기반의 DCT를 사용하였다. HM 3.0 이후, 변환과정의 중간 값이 16 비트 표현이 가능한 형태의 partial butterfly 형태의 2-D 분할 변환^[13]이 제안되어 현재까지 사용되고 있다.

이외에도 4×4 화면 내 예측에서는 모드에 따른 방향 의존성을 가지는 DST (discrete sine transform)를 지원하고, main 프로파일에는 채택되지 않았으나 정방형태의 변환 외에도 NSQT (non square quad-tree transform) 기술도 제안되었다.

2.4 루프 필터

HEVC의 루프 필터는 기존의 블록킹 열화를 제거하기 위한 디블록킹 필터 뿐 아니라, 양자화에서 오는 압축손실을 보상하기 위한 SAO (sample adaptive offset)와 ALF (adaptive loop filter)가 추가되었다^[14]. 디블록킹 필터의 경우 PU, TU 경계에서 발생하는 블록킹 열화를 제거하기 위한 기술로서 H.264/AVC와 유사하지만, 연산 복잡도를 낮추기 위해 필터과정을 간소화하였다. 이와 더불어 HEVC에서 채택된 SAO는 부호화과정을 통해 발생하는 원본 영상과 복원 영상간의 왜곡을 보상하기 위해 화소단위의 오프셋을 이용하는 기술이다. 디블록킹 필터와 달리, SAO 기술은 원본 영상과 복원 영상간의 오류를 직접 계산하기 때문에 주관적 화질뿐만 아니라 객관적 화질의 향상도 기대할 수 있다. ALF는 VCEG 회의에서 관련 기고서가 제안되었고, KTA 소프트웨어에 채택된 위너 (winner) 필터를 기반으로 하는 기술이다. 각 블록에 대한 필터 적용 여부 및 필터 계수를 전송하는 기술로서, HEVC 표준화 초기 단계에서는 HE (high efficiency)모드에서 ALF를 사용하였으나, 지난 7차 제네바 회의에서 프로파일에 대한 논의가 진행되면서 main 프로파일에서 제외되었다.

2.5 엔트로피 부호화

HEVC 엔트로피 부호화는 7차 제네바 회의 이전까지는 HE모드에서는 CABAC (context adaptive binary arithmetic coding)을 사용하고, LC (low complexity) 모드에서는 CAVLC (context adaptive variable length coding)를 사용하였다. 하지만, 7차 제네바 회의 이후, 모든 프로파일에서 CABAC만을 사용하고 있다. HEVC CABAC은 H.264/AVC CABAC의 기본구조를 그대로 채택하였고, 초기화 (initialize), 바이너리제이션 (binarization), 컨텍스트 모델러 (context modeler), 이진 산술 부호화 (binary arithmetic

coding), 확률 업데이트 (probability update)가 순차적으로 진행된다^[15].

III. HEVC 부호화기 복잡도 분석

본 논문에서는 HEVC 부호화기의 복잡도를 분석하기 위해 HEVC 참조소프트웨어인 HM 7.1을 실행하여 PC환경에서 소요되는 사이클을 측정하였다. HEVC 부호화기의 복잡도 측정을 위하여 HEVC 공통 실험 조건^[16]에 따라 표 3의 Class B (1920×1080), Class C (832×480)영상을 부호화 하였다. Intel® i7 2.80GHz Quad-core CPU, 4GB 메모리를 사용하였으며, Windows7 64bit OS 환경에서 부호화 사이클 측정을 수행하였다. 또한, PC환경에서 연산 사이클 측정은 Intel® VTune Amplifier XE 2011을 사용하였다. 세부적인 실험 환경은 표 4와 같다. 본 논문에서는 HEVC

부호화기에 대하여 복잡도 측정을 위해 II장에서 전술한 5가지 기능 모듈로 분류하였다. 또한, main 프로파일에서 임의 접근 및 저 지연 구조에 대하여 QP (quantization parameter) 22, 27, 32, 37을 사용하여 복잡도 측정을 수행하였다. 실제 어플리케이션에서의 수행을 고려하여 파일 입/출력과 관련된 복원 영상의 저장과 관련된 연산의 복잡도는 제외하였다. 본장의 1절에서는 HM 7.1을 사용하여 HEVC 부호화기의 복잡도를 분석하였으며, 2절에서는 추후 실제 어플리케이션에 적용될 HEVC 고속 부호화기에 대해 HM에 채택되어있는 고속 알고리즘을 적용한 고속 HM 부호화기의 예상 복잡도를 분석하였다.

1. HM 7.1 부호화기 소프트웨어 복잡도 분석

표 5와 표 6은 HEVC 부호화기에서 각 기능 모듈의 연산 사이클이 차지하는 비율을 나타낸다. 연산 사이클 수를 바

표 3. HEVC 공통 실험 조건에 따른 실험영상
Table 3. Test sequences upon HEVC common test conditions

Class	Sequence number	Sequence name	Frame count	Frame rate
Class B	S01	Kimono	240	24
	S02	ParkScene	240	24
	S03	Cactus	500	50
	S04	BQTerrace	600	60
	S05	BasketballDrive	500	50
Class C	S06	RaceHorses	300	30
	S07	BQMall	600	60
	S08	PartyScene	500	50
	S09	BasketballDrill	500	50

표 4. HEVC 부호화기 복잡도 측정을 위한 실험 환경
Table 4. Test environments for complexity analysis of HEVC encoder

Component	Description
CPU	Intel(R) CoreTM i7 CPU 860
Num. of cores	4
Num. of threads	8 (Hyper-threading)
Clock speed	2.8 GHz
Memory	4.00GB (DDR3)
OS	MS Windows 7 64bit
Compiler	MS Visual studio 2008
Analysis tool	Intel(R) VTune Amplifier XE 2011

표 5. 임의 접근에서 HM 7.1 부호화기의 복잡도 비율
 Table 5. Complexity ratio of HM 7.1 encoder in case of random-access

Class	Module	QP			
		22	27	32	37
B	Entropy	6.6	3.4	1.0	0.9
	Intra	3.3	2.2	2.1	1.4
	Inter	68.4	78.1	83.9	85.7
	Tr+Q	20.4	15.2	11.7	10.6
	Loop filter	0.2	0.2	0.2	0.1
	etc	1.2	1.1	1.3	1.5
C	Entropy	6.5	3.9	2.8	1.3
	Intra	2.9	2.7	2.2	1.8
	Inter	68.8	74.9	79.8	83.3
	Tr+Q	20.7	17.0	13.9	12.4
	Loop filter	0.2	0.2	0.2	0.1
	etc	1.0	1.5	1.4	1.2

표 6. 저 지연에서 HM 7.1 부호화기의 복잡도 비율
 Table 6. Complexity ratio of HM 7.1 encoder in case of low-delay

Class	Module	QP			
		22	27	32	37
B	Entropy	6.1	2.8	0.4	0.3
	Intra	3.4	2.0	1.2	1.2
	Inter	71.3	81.2	87.3	89.1
	Tr+Q	18.6	13.0	9.9	8.5
	Loop filter	0.2	0.2	0.2	0.1
	etc	0.8	1.2	0.8	0.9
C	Entropy	5.3	3.1	1.1	0.4
	Intra	3.0	2.5	1.8	1.5
	Inter	72.6	79.1	83.5	87.2
	Tr+Q	18.2	14.9	12.1	10.1
	Loop filter	0.2	0.2	0.2	0.1
	etc	1.1	0.6	1.6	1.0

탕으로 측정된 HEVC 부호화기의 복잡도는 엔트로피 부호화의 경우 0.3-6.6%를 차지했으며, 화면 내 예측은 1.2-3.3%, 화면 간 예측은 68.4-89.1%, 변환 및 양자화는 10.1-20.7%와 루프 필터는 0.1-0.2%를 차지하는 것으로 나타났다. 이외에도 내부 변수의 복사 및 함수 호출 등으로 소요되는 사이클은 0.6-1.6%를 차지하였다. HEVC 부호화기에서 가장 높은 복잡도를 차지하는 모듈은 화면 간 예측으로, 부호화기 전체에서 68.4-89.1%를 차지하였다. HEVC 부호화기의 화면 간 예측에서 예측 영상에 대한 보간 필터 연산이

부호화기 전체의 30%를 차지하였으며, 윗-왜곡 최적화 (RDO, rate-distortion optimization) 수행을 위한 SATD (sum of absolute transformed difference)가 13-15%, SAD (sum of absolute difference)가 10-12%를 차지하였다. 특히 보간 필터 연산은 단일 연산으로는 HEVC 부호화기에서 가장 높은 비율을 나타냈으며, 휘도 성분에 대한 8-tap (7-tap) 보간 필터 연산이 70%를 차지하였다. 보간 필터의 경우 필터 tap의 증가에 따른 산술 연산량 및 메모리 복사의 증가가 주요한 원인으로, 추후 GPU 혹은, SIMD (single instruction multiple data)를 이용한 데이터 단위 병렬화를 통해 고속 수행이 필요할 것이다. 또한, 현재 HM 소프트웨어에 구현되어 있는 PU 단위 보간 필터의 중복적인 수행을 줄이기 위해 프레임 단위 보간 필터의 사용이 가능하다. 하지만, 프레임 단위 보간 필터의 경우 복원영상의 저장을 위한 필요한 메모리가 증가하는 단점이 존재하므로 이를 절충할 수 있는 알고리즘 개발이 필요할 것이다. 화면 간 예측 다음으로 높은 복잡도를 차지한 모듈은 변환 및 양자화로, HEVC의 쿼드-트리 기반 TU 구조에 따른 변환 수행횟수의 증가와 32×32, 16×16 변환과 같은 변환 커널 크기의 증가로 인하여 전체 부호화기의 10.1-20.7%를 차지하였다. 또한, 윗-왜곡 최적화 과정에서 최적의 TU 쿼드-트리를 결정하기 위해 다양한 TU 크기에 대한 재귀적 수행구조 또한 변환 및 양자화 복잡도 증가의 주요한 원인 중 하나이다. 변환은 두 가지 방식, partial butterfly 방식과 행렬 곱셈 (matrix multiplication) 방식으로 구현이 가능하며 HM 소프트웨어에는 partial butterfly 방식으로 구현되어 있다. SIMD를 이용한 고속화에는 행렬 곱셈 방식이 적합하며, 이를 통한 고속화를 통해 복잡도를 낮출 수 있을 것으로 기대된다. 엔트로피 부호화는 전체 부호화기 중 0.3-6.6%를 차지하였으며, QP에 가장 의존적인 특성을 나타냈다. 엔트로피 부호화의 경우 main 프로파일에서는 CABAC만을 사용하고 있으며, 고속화, 병렬화가 어렵다는 단점이 존재한다. 추후 고속화, 최적화 및 병렬화를 통한 HEVC 고속 부호화기가 개발되었을 때는 엔트로피 부호화의 복잡도 비율이 증가할 것으로 예상되며, 이는 HEVC 고속 부호화를 위해 해결해야 할 문제 중 하나이다. 마지막으로, 루프 필터의 경우 부호화기의 0.1-0.2%의 낮은 복잡도를 가지며,

SAO 기술이 추가되었으나 H.264/AVC에 비해 더블록킹 필터 연산이 간소화되고 필터연산을 수행하는 화소의 수가 크게 줄어들었다.

기능 모듈에 따른 복잡도 비율 외에도 HEVC 부호화의 구조적 특성에 따라 CU의 크기 및 모드에 따른 복잡도 비율은 표 7과 같다. HEVC는 64×64에서 8×8까지 가변적인 CU 크기를 이용하여 부호화를 수행한다. II장에서 전술한 바와 같이, 가변적인 CU 크기에 따른 RDO 수행횟수의 증가는 HEVC 부호화의 높은 복잡도의 원인으로 꼽힌다. 표 7에서 나타나는 바와 같이, HEVC 부호화기 복잡도에 대하여 CU 크기별 비율을 살펴보면, 16×16, 32×32, 8×8, 64×64 순으로 나타났으며, 전체 부호화기 복잡도 중 각각 28.4%, 26.9%, 24.6%, 19.9%를 차지하였다. 또한, 모든 CU 크기에서 화면 간 예측의 복잡도가 85.3%로 가장 높은 비중을 차지하였으며, merge skip과 화면 내 예측 순으로 나타났다. 특히, 8×8 CU 크기에서는 4×4 크기의 화면 내 예측을 수행하기 때문에 다른 CU 크기에 비하여 화면 내 예측 비율이 0.5-2.0배 높게 나타났다.

표 7. HM 7.1에서의 CU 크기 및 모드에 따른 복잡도 비율
Table 7. Complexity ratio of each CU size and mode in HM 7.1

Size	Mode	RA(%)	LD(%)	Average(%)
8×8	Intra	1.8	1.4	1.6
	Inter	20.9	20.7	20.8
	Skip	2.4	2.0	2.2
16×16	Intra	0.8	0.7	0.8
	Inter	24.5	25.3	24.9
	Skip	3.0	2.5	2.7
32×32	Intra	1.1	1.0	1.0
	Inter	22.5	23.8	23.1
	Skip	3.0	2.7	2.8
64×64	Intra	1.2	1.1	1.2
	Inter	16.3	16.8	16.5
	Skip	2.4	2.1	2.2

2. HM 고속 부호화기 소프트웨어 복잡도 분석

전술한 HM 7.1 부호화기의 복잡도는 HEVC 부호화기의 초기 복잡도라 할 수 있으나, 실제 어플리케이션에 적용되기에는 최적화 및 고속화가 적용되지 않았다는 한계점을

가진다. 본 논문에서는 HEVC 부호화기의 초기 복잡도 뿐만 아니라, HM 7.1에 채택된 고속화 알고리즘을 적용한 복잡도를 제시함으로써 실제 어플리케이션에 적용될 HEVC 부호화기의 복잡도 예측치를 제시한다. HM 7.1에 적용된 고속화 기법 중 CBF (coded block flag) 고속 모드^[5]를 이용하여 고속화가 적용된 HEVC 부호화기로 가정하였다. CBF 고속 모드는 최적의 CU 경쟁 과정에서 CBF가 0인 PU가 발생할 경우 조기종료를 수행하는 알고리즘으로 BD-Bitrate는 1% 증가하는데 비해 부호화 속도는 평균 40% 감소한다. 표 8과 표 9는 고속 알고리즘을 적용한 HM 7.1 부호화기의 모듈별 복잡도 비율을 나타낸다. 고속 HEVC 부호화기에서는 화면 간 예측의 경우 59.1-82.8%를 나타냈으며, 기존 HM 7.1에 비해 화면 간 예측의 비율이 평균 8.3% 감소하였다. 반면, 변환 및 양자화의 복잡도는 12.8-25.8%로 상대적으로 평균 2-4% 증가한 것으로 나타났다. 엔트로피 부호화 또한 0.8-8.5%의 복잡도 비율을 나타냈다. 표 8과 표 9에서 나타나는 바와 같이, HEVC 부호화기에 대한 최적화 및 고속화가 적용될 경우 화면 간 예측의 복잡도 비율은 현재보다 더 낮아 질 것으로 예상된다. 표 10은 고속 알고리즘을 적용한 HM 7.1의 CU 크기 및 모드별 복잡도를 나타낸다. 고속화 알고리즘 이전과 비교하여 64×64, 32×32와 같은 큰 블록의 연산 복잡도 비율이

표 8. 임의 접근에서 HM 고속 부호화기의 복잡도 비율
Table 8. Complexity ratio of HM fast encoder in case of random-access

Class	Module	QP			
		22	27	32	37
B	Entropy	8.5	4.9	3.3	2.3
	Intra	4.5	3.5	3.2	2.8
	Inter	59.1	67.9	73.6	76.5
	Tr+Q	25.6	21.6	17.9	16.1
	Loop filter	0.2	0.3	0.4	0.2
	etc	2.2	1.9	1.8	2.2
C	Entropy	7.4	5.7	4.2	3.0
	Intra	4.5	3.8	3.4	3.0
	Inter	60.2	65.5	69.9	73.2
	Tr+Q	25.8	22.9	20.3	18.9
	Loop filter	0.2	0.3	0.3	0.2
	etc	2.1	1.9	2.0	1.9

2.7%, 1.3% 증가하는 것을 확인 할 수 있으며, Skip 모드의 복잡도 비율은 6.9% 증가하는 것으로 나타났다. 현재 HM 소프트웨어에서 최적의 CTU 결정과정은 재귀적으로 모든 서브-CU들에 대하여 RDO를 수행하기 때문에 높은 수준의 복잡도를 가지는 한계가 내재되어있다. 하지만, 조기종료와 같은 고속 알고리즘이 적용될 경우에는 16×16, 8×8과 같은 작은 블록의 연산 복잡도가 상대적으로 줄어들 것이다. 또한, 소프트웨어 최적화를 적용할 경우 메모리 복사가 빈번하게 일어나는 움직임 탐색에 대한 연산 복잡도 비율이 줄어들 것으로 예상된다.

표 9. 저 지연에서 HM 고속 부호화기의 복잡도 비율
 Table 9. Complexity ratio of HM fast encoder in case of low-delay

Class	Module	QP			
		22	27	32	37
B	Entropy	7.3	3.7	2.3	0.8
	Intra	4.0	3.2	2.4	1.9
	Inter	65.3	73.5	79.3	82.8
	Tr+Q	22.1	18.3	14.7	12.8
	Loop filter	0.1	0.2	0.2	0.1
	etc	1.3	1.1	1.3	1.7
C	Entropy	6.2	4.2	2.9	1.8
	Intra	4.0	3.5	2.9	2.4
	Inter	65.9	71.6	75.8	79.0
	Tr+Q	22.3	19.3	17.2	15.3
	Loop filter	0.1	0.2	0.2	0.1
	etc	1.6	1.3	1.2	1.5

IV. HEVC 부호화기 통계적 특성 분석

본 장에서는 HEVC 참조소프트웨어인 HM 7.1을 기반으로 HEVC 부호화의 통계적 특성과 HEVC에 적용된 향상된 기술에 대한 성능비교를 제시한다. HEVC 부호화기의 통계적 특성을 제시하기 위하여, HM 7.1 부호화기 소프트웨어를 기반으로 표 3의 영상을 부호화한 비트스트림을 사용하였다. 또한, HEVC의 향상된 기술들에 대한 성능비교를 위하여, HM 7.1 부호화기 소프트웨어에서 RDOQ (rate-distortion optimized quantization), SAO, AMP에 대한 on-off

실험을 통하여 복잡도 대비 비트 증가율을 제시한다.

1. 부호화 단위에 따른 통계적 특성

II장에서 진술한 바와 같이, HEVC는 기존의 비디오 코덱과 비교하여 세분화된 부호화 단위를 사용한다. 먼저, HEVC는 32×32부터 4×4까지 다양한 크기의 변환 커널을 사용하여 변환을 수행하며, 쿼드-트리 TU 구조를 갖는다. 표 11은 QP와 크기에 따른 TU 선택 비율을 나타낸다. 표 11에서의 선택 비율은 영상 내에서 각 TU가 차지하는 4×4

표 10. HM 고속 부호화기에서의 CU 크기 및 모드에 따른 복잡도 비율
 Table 10. Complexity ratio of each CU size and mode in HM fast encoder

Size	Mode	RA(%)	LD(%)	Average(%)
8×8	Intra	3.2	2.3	2.7
	Inter	16.5	16.7	16.6
	Skip	4.6	3.2	3.9
16×16	Intra	1.4	1.1	1.3
	Inter	19.3	20.5	19.9
	Skip	5.3	4.1	4.7
32×32	Intra	1.7	1.6	1.6
	Inter	20.6	23.4	22.0
	Skip	5.1	4.2	4.6
64×64	Intra	2.0	1.7	1.8
	Inter	16.4	18.1	17.3
	Skip	3.9	3.1	3.5

표 11. HM 7.1에서의 QP와 크기에 따른 TU 선택 비율
 Table 11. Selected ratio of TU according to QP and size in HM 7.1

Class	Size	QP			
		22	27	32	37
B	4×4	4.4	1.7	0.7	0.3
	8×8	15.1	3.5	1.6	0.9
	16×16	9.3	6.2	3.9	2.6
	32×32	16.1	16.3	12.6	9.3
	Non-TR	55.2	72.3	81.2	86.8
C	4×4	4.9	2.4	1.1	0.5
	8×8	7.7	4.6	2.7	1.6
	16×16	10.3	7.5	5.1	3.7
	32×32	15.4	13.6	10.3	7.3
	Non-TR	61.7	72.0	80.8	86.9

단위 면적의 비율을 나타낸다. 표 11에서 나타나는 바와 같이, 변환을 수행하지 않는 블록의 선택 비율은 평균 55.2-86.9%로 가장 높게 나타났다. 이를 제외한 실제 변환이 수행되는 블록 중에서는 32×32 변환의 선택 비율이 33.5-65.7%로 가장 높게 나타났으며, 16×16 변환은 19.7-27.7%로 나타났다. 또한, 8×8 변환은 10.0-36.2%의 비율을 나타냈으며, 4×4 변환은 4.5-14.8%로 가장 낮게 나타났다. 이때, 실제 변환을 수행하는 블록 중 4×4, 8×8 TU로 선택되는 비율은 평균 60.2%, 24.8%로 높게 나타났으나, 16×16, 32×32 변환에 비하여 처리하는 면적이 낮게 나타났다. TU 크기는 QP에 의존적인 특성을 나타냈으며, QP가 증가할수록 큰 변환 블록의 선택 비율이 증가하는 것을 확인 할 수 있었다. 또한, 4×4 TU 중, DST는 평균 53.1%의 선택 비율을 나타내는 것을 확인할 수 있었다. DST는 화면 내 예측 모드에 따라 수행 여부가 결정되므로 영상에 매우 의존적인 특성을 가지며, 영상에 따라 최대 75.3%, 최소 3.5%의 선택 비율을 나타내며 큰 편차를 보였다.

HEVC는 CU와 TU외에도 예측을 위한 부호화 단위인 PU를 사용한다. II장에서 전술한 바와 같이, PU는 각 CU 크기별로 최대 8가지 모양을 갖는다. 또한, 예측 방법에 따라 크게 화면 내 예측, 화면 간 예측, Skip 모드 3가지로 분류할 수 있다. 표 12와 표 13은 Class B와 Class C에서

표 12. Class B에서 QP에 따른 CU 크기 및 PU 모드의 선택 비율
Table 12. Selected ratio of CU size and PU mode according to QP in Class B

Size	Mode	QP			
		22	27	32	37
8×8	Intra	3.7	1.8	1.1	0.7
	Inter	6.3	1.7	0.5	0.2
	Skip	2.8	1.9	1.2	0.9
16×16	Intra	2.6	2.1	1.7	1.4
	Inter	16.6	6.3	2.7	1.3
	Skip	6.8	5.6	3.9	2.9
32×32	Intra	2.3	2.3	2.6	2.6
	Inter	16.0	12.5	7.8	5.1
	Skip	9.9	12.4	9.9	8.4
64×64	Intra	0.3	0.4	0.6	1.0
	Inter	8.7	12.7	11.3	8.8
	Skip	10.6	26.6	43.3	55.2

표 13. Class C에서 QP에 따른 CU 크기 및 PU 모드의 선택 비율
Table 13. Selected ratio of CU size and PU mode according to QP in Class C

Size	Mode	QP			
		22	27	32	37
8×8	Intra	3.9	2.8	1.8	1.1
	Inter	6.8	3.5	1.6	0.6
	Skip	3.9	3.3	2.3	1.4
16×16	Intra	1.2	1.6	1.8	1.7
	Inter	14.3	9.6	6.2	3.7
	Skip	8.0	7.7	7.3	6.1
32×32	Intra	0.2	0.4	0.7	1.1
	Inter	16.9	13.6	11.3	9.4
	Skip	12.2	13.5	15.2	16.8
64×64	Intra	0.9	0.0	0.0	0.1
	Inter	9.7	11.4	10.3	9.1
	Skip	11.7	20.6	30.6	39.5

QP에 따른 CU 크기 및 PU 모드의 선택 비율을 나타낸다. 표 12와 표 13에서도 표 11과 동일하게 각 CU 크기 및 PU 모드가 영상에서 차지하는 4×4 단위 면적의 비율을 나타낸다. Class B와 Class C의 선택 비율에서 나타나는 바와 같이, 부호화 하는 영상의 크기가 클수록 큰 크기의 CU의 선택 비율이 높게 나타나는 것을 확인 할 수 있다. 반면, Class C의 경우에는 Class B에 비하여 작은 CU의 선택 비율이 상대적으로 증가하였다. 또한, 64×64와 32×32의 큰 CU에서는 화면 내 예측이 차지하는 비율이 크게 감소하는 것을 확인 할 수 있었으며, 16×16과 8×8의 작은 CU에서는 화면 내 예측 비율이 상대적으로 높게 나타났다. 본 논문에서 다루고 있는 Class B, Class C 영상보다 큰 UHD (ultra high definition)와 같은 영상 부호화의 경우 이러한 현상은 더욱 심화될 것으로 예상되며, 이러한 경우 영상 크기에 따른 CU 크기 및 모드의 선택적 부호화가 필요할 것이다. 영상의 크기 변화 외에도 QP가 증가함에 따라서 Skip 모드의 선택 비율이 증가하였으며, Class B의 경우 QP 37에서 64×64 Skip 모드의 선택비율이 55.2%로 영상의 절반 이상의 블록들이 선택됨을 알 수 있다. Skip과 화면 간 예측의 비율이 높게 나타나는 이유는 AMVP와 병합을 사용한 예측 움직임 선택 및 참조 프레임 선택의 정확성 향상과 보간

필터의 향상된 성능을 꼽을 수 있다.

화면 간 예측 모드의 선택 비율을 보다 자세히 살펴보면 표 14와 같다. 표 14에서 나타나는바와 같이, 화면 내 예측 모드의 선택 비율은 Planar 모드와 DC 모드의 선택비율이 평균 21.6%와 11.8%로 높게 나타났다. 방향성에 따른 모드들의 경우 영상에 의존적인 특성을 가지며 수평 (horizon-

tal) 모드와 수직 (vertical) 모드가 각각 3.4%, 11.2%로 나타났다으며, 이외 방향성 모드의 경우 모드별 1-2%의 비율이 나타났다. 화면 간 예측 모드의 선택 비율에 관한 표는 표 15이며, 화면 간 예측 모드에서는 2N×2N의 선택 비율이 55.9-58.0%로 높게 나타났다. 이외, 2N×N과 N×2N의 경우 8.8-10.4%, 13.3-16.6%로 나타났으며, AMP모드인 nL×2N, nR×2N, 2N×nU, 2N×nD의 경우 16.6-20.4%의 비율을 나타냈다.

표 14. 화면 내 예측 모드의 선택 비율
 Table 14. Selected ratio of intra prediction mode

Mode	Ratio(%)	Mode	Ratio(%)	Mode	Ratio(%)
0	21.6	12	1.4	24	2.0
1	11.8	13	1.1	25	5.0
2	1.2	14	1.1	26	11.2
3	0.9	15	1.1	27	2.7
4	1.0	16	1.1	28	2.1
5	1.2	17	1.4	29	1.9
6	2.3	18	1.9	30	1.5
7	1.8	19	1.8	31	1.2
8	1.7	20	1.8	32	1.0
9	1.8	21	1.9	33	1.0
10	3.4	22	2.0	34	1.1
11	1.9	23	2.1		

표 15. 화면 간 예측 모드의 선택 비율
 Table 15. Selected ratio of inter prediction mode

Size	Mode	QP			
		22	27	32	37
8×8	2N×2N	10.5	5.2	2.7	1.4
	2N×N	1.1	0.8	0.4	0.2
	N×2N	1.6	1.3	0.8	0.4
16×16	2N×2N	16.2	11.7	9.0	6.8
	2N×N	3.0	1.9	1.4	0.9
	N×2N	3.9	3.0	2.4	1.8
	AMP	8.6	5.8	4.1	2.7
32×32	2N×2N	18.0	19.0	18.8	18.3
	2N×N	3.3	3.2	3.1	3.1
	N×2N	4.4	4.9	5.4	5.9
	AMP	9.3	9.2	9.2	9.0
64×64	2N×2N	11.2	19.9	26.6	31.6
	2N×N	3.0	4.3	4.4	4.6
	N×2N	3.4	6.1	7.5	8.5
	AMP	2.5	3.6	4.2	4.9

2. 제한적 부호화에 따른 통계적 특성

본 논문에서는 전술한 HEVC 부호화의 통계적 특성 외에도 RDOQ, SAO, AMP에 대한 on-off와 CTU 크기를 제한하는 실험을 수행하여 다양한 툴들의 부호화 성능 대비 속도를 비교한다. 또한, 최대 TU 크기 및 TU 계층적 깊이를 제한하는 실험을 통하여 변환 커널에 크기 및 계층적 깊이에 따른 부호화 성능 및 속도를 살펴본다. 그림 4에서는 AMP, SAO, RDOQ에 대하여 총 8가지 조합을 이용하여 각 툴의 on-off에 따른 부호화 시간의 감소율 대비 BD-Bitrate의 증가율을 나타내었다. 그림 4에서 RDOQ, SAO, AMP를 모두 동작시켰을 때 (REF)를 기준으로 SAO만을 동작시키지 않은 경우 (AMP+RDOQ) 평균 1.6%의 BD-Bitrate가 증가하였으나 평균 0.7%의 부호화 시간이 감소하여 SAO는 부호화 속도에 비하여 부호화성능 향상이 높은 것으로 나타났다. 반면, AMP만을 동작시키지 않은 경우 (SAO+RDOQ) 평균 0.5%의 BD-Bitrate가 증가하고 평균 10.3%의 부호화 시간이 감소하여 AMP는 부호화 속도에 비하여 부호화성능 향상이 높지 않은 것으로 나타났다. RDOQ를 동작시키지 않은 경우 (SAO+AMP)는 평균 5.4%의 BD-Bitrate 증가를 나타내며, RDOQ는 전술한 3가지 툴 중에서 부호화 성능 향상이 가장 높은 것으로 나타났다. SAO, RDOQ, AMP를 모두 사용하지 않은 경우에는 평균 7.5%의 BD-bitrate가 증가하였으며 부호화 시간은 평균 18.3% 감소하는 것으로 나타났다. RDOQ와 SAO의 경우 부호화 시간 대비 부호화 성능이 높은 것으로 나타났으며, AMP의 경우 전술한 두 기술에 비하여 부호화 시간 대비 부호화 성능이 높지 않은 것으로 나타났다.

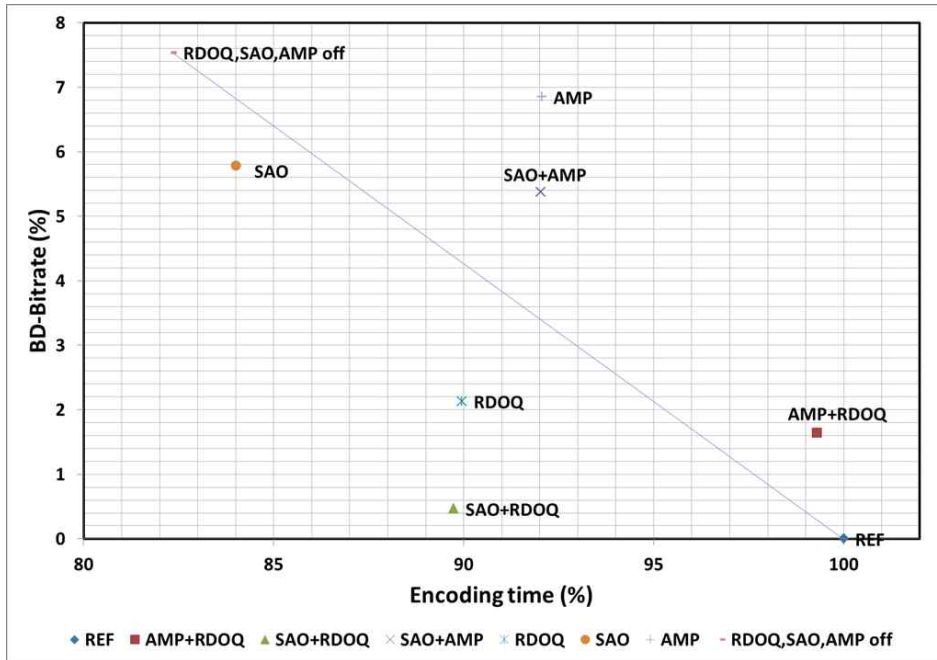


그림 4. RDOQ, SAO, AMP의 on-off에 따른 부호화 시간과 BD-Bitrate의 상관도
 Fig. 4. Correlation of encoding time and BD-Bitrate by maximum RDOQ, SAO, AMP

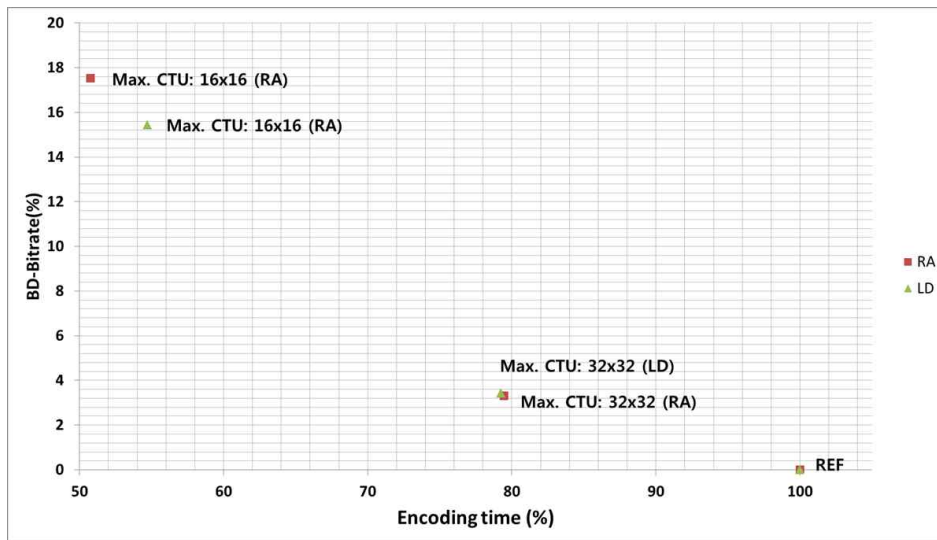


그림 5. CTU 크기의 변화에 따른 부호화 시간과 BD-Bitrate의 상관도
 Fig. 5. Correlation of encoding time and BD-Bitrate by CTU size change

그림 5는 CTU 크기를 64×64에서 16×16으로 변경하며 발생하는 부호화 시간의 감소율 대비 BD-Bitrate의 증가율을 나타내었다. CTU 크기를 64×64로 부호화하였을 때

(REF)를 기준으로 최대 CTU 크기를 32×32로 부호화하는 경우 평균 3.3-3.4%의 BD-Bitrate 증가가 있었으나 평균 20.8-21.1%의 부호화 속도가 감소하였다. 또한, CTU 크기

를 16×16으로 부호화하는 경우 평균 15.4-17.5%의 BD-Bitrate 증가가 있었으며, 부호화 속도는 평균 49.1-50.1% 감소하였다. CTU 크기를 16×16으로 제한할 경우, 예측을 수행하는 PU 크기의 제한과 최대 TU 크기가 16×16으로 제한되므로 비트율 증가가 크게 나타나는 것을 확인 할 수 있다. 영상의 크기와 응용의 종류에 따라 CTU 크기를 제한적으로 사용할 경우 CTU 크기를 32×32로 제한하여 부호화를 수행할 수 있을 것으로 예상되나, 16×16으로 제한할 경우 부호화 성능 저하가 심하게 나타나므로 CTU 크기를 16×16으로 제한하는 것은 지양해야 할 것이다. 그림 6은 최대 TU 크기와 변환 깊이를 변경하며 측정된 부호화 시간의 감소율 대비 BD-Bitrate의 증가율을 나타낸다. 최대 TU 크기는 32×32, 변환 깊이는 3인 경우 (REF)를 기준으로 최대 TU 크기를 16×16, 8×8로 제한하였으며, 변환 깊이를 3에서 1까지 제한하며 총 5가지 조합에 대한 실험을 수행하였다. 최대 TU 크기를 16×16, 변환 깊이를 3으로 유지한 경우, 즉, 32×32 변환을 제거한 경우에는 3.2-3.5%의 BD-Bitrate 증가하였으나 3.2-3.5%의 부호화 속도가 감소하였다. 최대 TU 크기를 8×8로 설정하여 8×8, 4×4 변환을 수행하는 경우에는 10.2-11.2%의 BD-Bitrate 증가와 7.6- 8.6%의 부호화 속도 감소가 나타나 큰 크기의 변환이 HEVC 부

호화 성능 향상에 큰 역할을 차지함을 확인 할 수 있었다.

V. 결 론

본 논문에서는 HEVC 부호화기에 대하여 살펴보고, 고속화, 최적화 및 병렬화를 위해 선행되어야 할 통계적 특성 및 복잡도 분석을 수행하였다. HEVC 부호화기 소프트웨어의 복잡도 분석은 HM 7.1 부호화기 소프트웨어를 기반으로 5개의 기능 모듈에서 차지하는 복잡도 비율을 연산 사이클의 비율을 사용하여 제시하였다. 화면 간 예측은 HEVC 부호화기 중 68.4-89.1%로 가장 높은 복잡도 비율을 차지하였으며 변환 및 양자화는 10.1-20.7%로 높은 복잡도 비율을 차지하였다. 이 중, 참조 영상의 보간 필터 연산은 30%의 복잡도 비율을 차지하였으며, RDO 과정에서의 SATD, SAD 연산은 부호화기의 20-25%를 차지하였다. 또한, HM 7.1 부호화기 소프트웨어를 기반으로 측정된 초기 HEVC 부호화기의 복잡도 외에도 실제 어플리케이션에 적용될 것으로 예상되는 고속화 및 최적화된 HEVC 부호화기의 복잡도 예측치를 고속 알고리즘을 적용한 HM 소프트웨어를 바탕으로 살펴보았다. 고속화가 적용된 HM 부호화

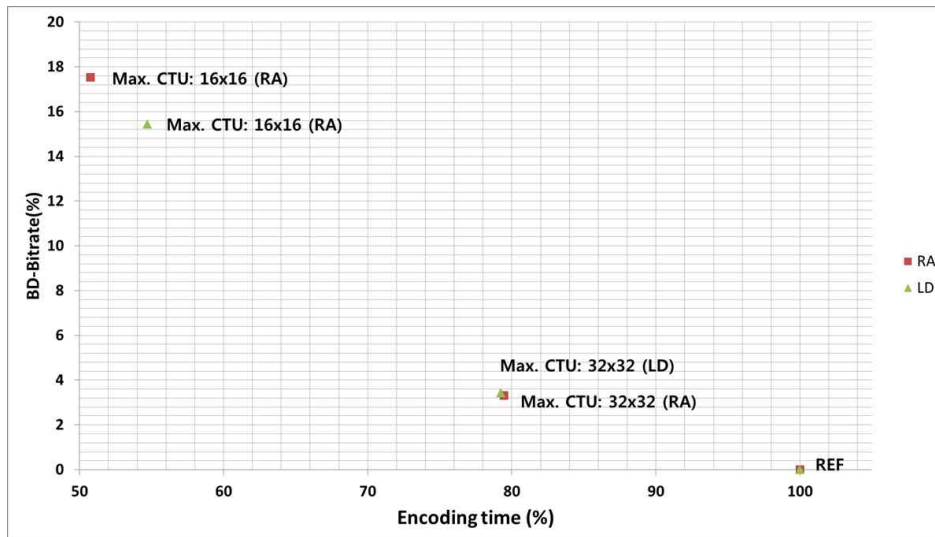


그림 6. 최대 TU 크기 및 변환 깊이에 따른 부호화 시간과 BD-Bitrate의 상관도

Fig. 6. Correlation of encoding time and BD-Bitrate by maximum TU size and TU depth change

기의 경우, 화면 간 예측의 복잡도 비율이 상대적으로 감소하는 것을 확인할 수 있었으며, 고속화 및 최적화가 적용될 경우 화면 간 예측의 복잡도 비율은 추가적으로 감소할 것으로 예상된다. HEVC 부호화기의 복잡도 분석과 더불어, TU 및 PU의 선택 비율과 같은 HEVC 부호화기의 통계적 특성 및 다양한 제한을 통한 부호화 기법의 비트율 대비 부호화 시간 성능들을 확인 할 수 있었다. TU 및 PU의 선택 비율의 경우 QP와 영상의 크기에 높은 의존성을 갖는 것으로 나타났으며, 큰 크기의 부호화 단위 선택 비율이 상대적으로 높게 나타나 부호화 크기의 증가가 HEVC 부호화 성능 향상에 중요한 역할을 차지하였음을 확인 할 수 있었다.

HEVC는 H.264/AVC에 비해 약 40%이상의 부호화 효율 향상이 있으나 2배 이상 증가하는 복잡도는 여전히 해결해야 할 문제점으로 지적되고 있다. 앞으로 많은 고속화, 최적화 및 병렬화 기법이 제안되어 HEVC 부호화기의 실시간 동작이 조속히 이루어지기를 희망해 본다.

참 고 문 헌

[1] Y. J. Ahn, W. J. Han, and D. G. Sim, "Study of decoder complexity for HEVC and AVC standards based on tool-by-tool comparison," SPIE Applications of Digital Image Processing XXXV, Proceedings of SPIE Volume 8499, Paper number 8499-32, San Diego, USA, August, 2012.

[2] F. Kossentini, N. Mahdi, H. Guermazi, H. M. Horowitz, S. Xu, B. Li, G. J. Sullivan, and J. Xu, "Informal subjective quality comparison of compression performance of HEVC working draft 5 with AVC high profile," JCTVC-H0562, 8th JCT-VC meeting, San Jose, USA, Feb, 2012.

[3] Y. Zhao, and L. Yu, "Coding efficiency comparison between HM 5.0 and JM 16.2 based on PQI, PSNR and SSIM," JCTVC-H0063, 8th JCT-VC meeting, San Jose, USA, February, 2012.

[4] B. Li, G. J. Sullivan, and J. Xu, "Compression performance of high efficiency video coding (HEVC) working draft 4," Proc. Of ISCAS, pp. 22-23, Seoul, Korea, May, 2012.

[5] R. H. Gweon, Y.-L. Lee, and J. Lim, "Early Termination of CU Encoding to Reduce HEVC Complexity," JCTVC-F045, Torino, IT, July, 2011.

[6] K. Choi, S.-H. Park, and E. S. Jang, "Coding tree pruning based CU early termination," JCTVC-F092, Torino, IT, July, 2011.

[7] J. Yang, J. Kim, K. Won, H. Lee, and B. Jeon, "Early skip detection for HEVC," JCTVC-G543, CH, November, 2011.

[8] HEVC reference software HM SVN repository "http://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-7.1".

[9] A. Fuldseth, M. Horowitz, S. Xu, A. Segall, M. Zhou, "Tiles," JCTVC-F335, Torino, Italy, July, 2011.

[10] B. Bross, W. J. Han, J. R. Ohm, G. J. Sullivan, and T. Wiegand, "High efficiency video coding (HEVC) text specification draft 7," JCTVC-H1003, San Jose, USA, February, 2012.

[11] J. Chen, T. Lee, "Planar intra prediction improvement," JCTVC-F483, Torino, IT, July, 2011

[12] E. Alshina, A. Alshin, J. H. Park, J. Lou, K. Minoo, "CE3: 7 taps interpolation filters for quarter pel position MC from Samsung and Motorola Mobility," JCTVC-G778, Geneva, CH, November, 2011.

[13] A. Fuldseth, G. Bjntegaard, M. Budagavi, V. Sze, "CE10: Core transform design for HEVC," JCTVCC-G495, Geneva, CH, November, 2011.

[14] C. M. Fu, C. Y. Chen, C. Y. Tsai, Y. W. Huang, S. Lei, "CE13: Sample Adaptive Offset with LCU-Independent Decoding," JCTVC-E049, Geneva, CH, March, 2011.

[15] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard," IEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 7, pp. 620-636, July, 2003.

[16] F. Bossen, "Common test conditions and software reference configurations," JCTVC-G1200, Geneva, CH, November, 2011.

저 자 소 개



안 용 조

- 2010년 2월 : 광운대학교 컴퓨터공학과 학사
- 2012년 2월 : 광운대학교 컴퓨터공학과 석사
- 2012년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 박사과정
- 주관심분야 : 영상압축, 최적화 및 병렬화

저 자 소 개



황 태 진

- 2012년 2월 : 광운대학교 컴퓨터공학과 학사
- 2012년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 석사과정
- 주관심분야 : 영상압축, 멀티미디어시스템



유 성 은

- 2011년 2월 : 광운대학교 컴퓨터공학과 학사
- 2011년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 석사과정
- 주관심분야 : 영상압축, 최적화 및 병렬화



한 우 진

- 1995년 2월 : KAIST 전산학과 공학사
- 1997년 2월 : KAIST 전산학과 공학석사
- 2002년 2월 : KAIST 전산학과 공학박사
- 2002년 3월 ~ 2003년 3월 : SL2 연구소장
- 2003년 4월 ~ 2011년 8월 : 삼성전자 DMC 연구소 수석연구원
- 2010년 10월 ~ 현재 : HEVC Working Draft Editor
- 2011년 9월 ~ 현재 : 가천대학교 소프트웨어설계경영학과 조교수
- 주관심분야 : 영상압축, 영상이해, 멀티미디어통신



심 동 규

- 1993년 2월 : 서강대학교 전자공학과 공학사
- 1995년 2월 : 서강대학교 전자공학과 공학석사
- 1999년 2월 : 서강대학교 전자공학과 공학박사
- 1999년 3월 ~ 2000년 8월 : 현대전자 선임연구원
- 2000년 9월 ~ 2002년 3월 : 바로비전 선임연구원
- 2002년 4월 ~ 2005년 2월 : University of Washington Senior research engineer
- 2005년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 부교수
- 주관심분야 : 영상신호처리, 영상압축, 컴퓨터비전