

특집논문 (Special Paper)

방송공학회논문지 제21권 제3호, 2016년 5월 (JBE Vol. 21, No. 3, May 2016)

<http://dx.doi.org/10.5909/JBE.2016.21.3.341>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

하드웨어 기반 HEVC 인트라 인코더에서 다운 샘플링을 사용한 고속 Rough Mode Decision

장 지 훈^{a)}, 이 채 은^{a)†}

Down Sampling for Fast Rough Mode Decision for a Hardware-based HEVC Intra-frame encoder

Ji Hun Jang^{a)} and Chae Eun Rhee^{a)†}

요 약

HEVC 표준은 기존의 H.264 표준을 대체할 차세대 고효율 영상 압축 코덱이다. H.264 표준에 비해 약 50% 수준으로 비트레이트를 감소시켰지만 계산 복잡도는 약 1.4배 정도 증가하였다. 계산 복잡도를 낮추기 위해 다양한 고속화 알고리즘들이 제안되어 왔다. 인트라 코딩에는 rough mode decision(RMD) 기법이 적용되었다. 최적의 모드를 선정하기 위한 rate-distortion optimization (RDO) 과정은 복잡도가 높기 때문에 RMD를 사용하여 더 간소화된 방법으로 RDO 단계를 위한 후보 모드들을 선정한다. 그러나 큰 사이즈의 블록들의 경우 RMD 과정 역시 계산 복잡도를 줄일 필요가 있다. 본 논문에서는 RMD 과정에서 참조 픽셀을 가져오고, 예측 픽셀 생성하는 과정에서 다운 샘플링을 적용하였으며 참조 소프트웨어에 적용된 기존 RMD 방식에 비해 계산량을 70%가량 줄일 수 있었다. 이때 BDBR 증가는 0.04%로 미미한 수준이다. 제안한 다운샘플링 기법을 RMD 하드웨어에 적용하면 게이트 카운트는 약 33%, 버퍼의 크기는 약 66% 줄어든다.

Abstract

HEVC is the next compression standard and is expected to be used widely replacing the conventional H.264/AVC standard. The compression ratio of the HEVC is twice times than H.264/AVC, whereas its computational complexity is increased by up to 40%. Many research efforts have been made to reduce the computational complexity and to speed up encoding. For intra coding, the rough mode decision (RMD) is commonly applied. The rate-distortion optimization (RDO) process to decide the best mode is too complex so that RMD chooses the candidate modes with a simple process and sends the candidates to RDO process. However, for large-size blocks, the RMD also requires considerable computations. In this paper, a down-sampling scheme is proposed for the RMD process. The reference pixel loading, predicted pixel generation are performed using the down-sampled pixel data. When the proposed scheme is applied to the RMD, the computational complexity is reduced by 70% with a marginal bitrate increase of 0.04%. In terms of area of hardware-based RMD, the gate count and the buffer size is reduced 33% and 66%, respectively.

Keyword : Down-sampling, Rough mode decision, HEVC, intra prediction

a) 인하대학교 전자공학부(Dept. of Information and Communication Engineering, Univ. of Inha)

† Corresponding Author : 이채은(Rhee Chae Eun)

E-mail: chae.rhee@inha.ac.kr

Tel: +82-31-330-6443

ORCID: <http://orcid.org/0000-0002-7851-1703>

※ 이 논문의 연구결과 중 일부는 "IWAIT 2016" 에서 발표한 바 있음.

※ 이 논문은 2015년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2015R1C1A1A02037625).

· Manuscript received March 21, 2016; Revised May 24, 2016; Accepted May 24, 2016.

1. 서론

High Efficiency Video Coding(HEVC) Standard는 기존 H.264 Standard에 비해 약 50% 수준으로 비트레이트를 감소시켰지만 계산 복잡도는 1.4배 정도 증가하였다^[1]. HEVC의 예측은 크게 두 종류로 Intra Prediction과 Inter Prediction으로 나뉜다. 그중 Intra Prediction의 경우 최소 8x8 pixels 크기부터 최대 64x64 pixels의 크기를 갖는 Coding Unit(CU)에 대해서 35가지 모드에 대해 예측을 수행한다. 이는 기존의 H.264가 4x4 pixels 크기의 블록을 9가지 모드에 대해서 예측을 하는 것과 비교했을 때, 예측 모드 수는 약 3.9배 많아졌으며 다양한 사이즈의 블록을 사용하여 예측을 수행하므로 압축 성능은 높아졌지만 계산 복잡도도 높아졌다. 최적의 모드를 찾기 위해서는 모든 예측 모드와 블록 사이즈에 대하여 복원 이미지를 생성하여 bitrate 및 왜곡의 정도를 계산하는 Rate Distortion Optimization(RDO)를 수행해야 한다^[2]. 하지만 RDO 과정이 복잡도가 높기 때문에 모든 경우에 대하여 RDO를 수행하는 것은 지나치게 높은 계산 복잡도를 필요로 하는 문제가 있다.

HEVC에서 부호화 효율을 최대한 유지하면서, 복잡도를 낮추기 위한 방법으로 Rough Mode Decision(RMD)가 사용된다. RMD가 적용된 HEVC의 최적의 모드 선택 과정은 그림 1과 같다. Intra Prediction이 시작되면 35가지의 모든 모드에 대해서 예측 블록을 생성한다. 이어지는 RMD 과정

에서 SATD 값을 계산하고 PU의 크기에 따라 N 개의 후보 모드들을 결정한다^[3]. 다음으로 RMD 결과 선정된 후보 모드와 Most Probable Mode(MPM)에 대해서 RDO 과정을 수행한다. 일반적인 자연 영상에서 인접한 픽셀은 비슷한 영상 특성을 갖기 때문에 Intra prediction의 모드 역시 비슷할 확률이 높다고 볼 수 있다. 이런 특성을 고려하여 주변 PU의 모드를 기반으로 현재 PU의 모드를 부호화하며, 이 과정에서 선정되는 모드들을 MPM이라 한다^[4]. 후보 모드들과 MPM에 대한 R-D cost가 계산되면 최적의 모드를 선정하고 Intra prediction은 종료된다. RMD과정에서 선정된 N개의 후보 모드에 대해서만 RDO를 수행하게 되는데 N은 Prediction Unit(PU)의 크기에 따라 결정되며 PU의 크기가 4x4 또는 8x8일 경우 N은 8이고, 나머지 경우 3값을 갖는다. 후보 모드에 대해서만 RDO를 수행하기 때문에 복잡도를 크게 줄이고 압축 효율은 비슷하게 유지할 수 있다. RMD는 복원 블록과 원본 블록 사이의 왜곡 정도를 계산하는 대신 예측 블록과 원본 블록 사이의 왜곡만을 고려하는 방법이다. 복원 과정이 없기 때문에 RDO보다 간단하며, RMD에서는 간단한 덧셈과 뺄셈으로 구성된 Sum of absolute Transformed differences(SATD)값으로 Cost를 계산한다.

RDO를 적용하는 모드 수를 줄여서 인트라 예측 전체 계산 복잡도가 감소할 수 있었지만 35개의 모드에 대해 RMD를 모두 수행한다는 점과 다양한 블록 사이즈에 대하여 모두 RMD를 수행하여야 하므로 RMD 과정에서의 계산 복잡

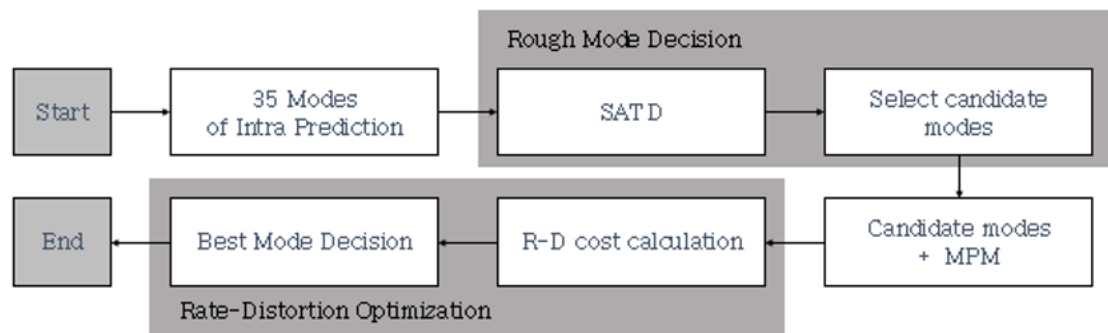


그림 1. HEVC의 화면 내 예측 고속화 알고리즘
Fig. 1. Fast Mode Decision Algorithm in HEVC

도 역시 높다고 할 수 있다. 특히, 32x32 또는 64x64 크기의 블록 사이즈에서는 높은 계산 복잡도와 함께 요구되는 하드웨어 리소스도 매우 크다. 본 논문은 큰 블록 사이즈에서 계산 복잡도를 낮추기 위해 다운 샘플링을 사용하는 방법을 보인다. 큰 사이즈의 블록을 가로, 세로 각 절반 크기로 다운 샘플링하여 계산량을 줄이고 빠른 처리가 가능하다. RMD를 통해 후보 모드 선정만 하고 RMD과정에서 얻은 결과 픽셀로 실제 부호화가 진행되는 것은 아니기 때문에 다운 샘플링을 하더라도 spatial correlation만 어느 정도 유지된다면 압축 성능에도 큰 영향을 주지 않는다는 장점이 있다.

본 논문의 구성은 다음과 같다. 2장에서는 다운 샘플링 기법에 대해 설명하고, 3장에서는 2장에서 설명한 기법을 적용한 RMD 하드웨어에 대해 기술한다. 본 논문에 제안된 기법과 기존의 방법에 의한 실험 결과와 결론을 4장에 말한다.

II. 제안하는 다운 샘플링 기법

RMD단계에서 참조 픽셀들을 이용하여 예측 픽셀을 생성하게 된다. 예측 픽셀 생성을 위해서는 현재 CU 크기 2배 길이의 참조 픽셀들을 필요로 한다. 그림 2의 (a)는 크기가

16x16인 CU의 참조 픽셀들을 나타낸다. 가로, 세로 각 32개의 픽셀들이 필요하며 CU의 좌측 상단의 픽셀까지 총 65개의 참조 픽셀을 사용한다. 크기가 32x32인 CU의 경우 총 129개의 참조 픽셀이 필요하며, 64x64크기의 CU에서는 총 257개의 참조 픽셀이 필요하다. 현재 CU의 크기가 계산 복잡도와 참조 픽셀을 저장하기 위해 필요로 하는 버퍼의 크기에 크게 영향을 주는 것을 알 수 있다. 보통 자연 영상에서 인접 픽셀들 간에 높은 spatial correlation이 있다^[5]. 실제로 다운샘플링 전후의 RMD수행에서 선택된 예측 모드를 비교해 보면 다운 샘플링 후 선택된 예측 모드가 다운 샘플링 이전 RMD로 예측된 모드 중에 속할 확률이 약 70%이다. 이러한 관찰 결과로 다운샘플링을 수행하더라도 spatial correlation이 크게 바뀌지 않는다고 가정할 수 있다. 이러한 특성을 기반으로 RMD를 위한 예측 픽셀을 생성하는 과정에서 참조 픽셀과 원본 픽셀에 가로, 세로 2:1 비율로 다운 샘플링을 수행하였다. 그림 2의 (b)는 다운 샘플링 기법이 적용된 RMD를 나타낸다. 그림 2의 (a)와 (b) 모두 16x16 크기의 CU에 대한 RMD과정을 나타내고 있으며 (b)를 보면 가로, 세로 각 2:1 비율로 다운 샘플링했기 때문에 총 33개의 참조 픽셀을 사용하여 64개의 예측 픽셀을 생성한 것을 알 수 있다.

다운 샘플링 진행 과정에서 추가적인 연산을 최대한 배제하기 위해 보간과 같은 연산을 사용하지 않고 그림3과

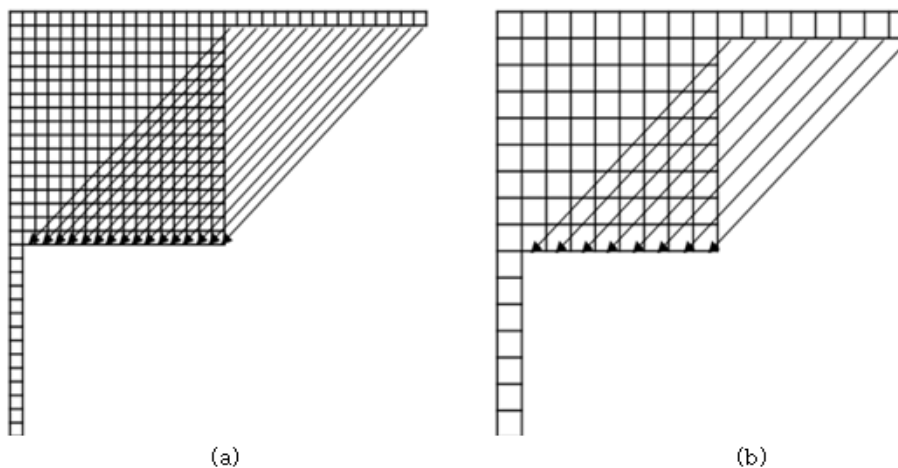


그림 2. 16x16 CU의 화면 내 예측 (a) 다운샘플링 이전의 참조 픽셀과 예측 (b) 다운샘플링 된 참조 픽셀과 예측
 Fig. 2. Intra prediction of 16x16 CU (a) without downsampling (b) with downsampling

같이 픽셀을 선택한다. 가로 방향과 세로 방향의 홀수 번째 픽셀을 참조 픽셀로 선정한다. 다음으로, 가로 방향과 세로 방향의 짝수 번째 픽셀을 원본 픽셀로 선정한다. 즉, 그림 3이 크기가 16x16인 CU의 RMD과정을 위한 예측 픽셀 생성을 나타낸다고 할 때, 회색 음영으로 표시한 픽셀들이 다운 샘플링 기법에서 선정되는 픽셀들이다. Original RMD와 비교했을 때 예측 픽셀 생성과 SATD 계산에 사용되는 픽셀 수가 감소했음을 볼 수 있다. SATD계산에 8x8 하다마드 변환이 사용되며, 16x16크기의 블록에 대한 SATD값 계산에는 8x8 하다마드 변환이 4번 수행된다. SATD계산에 사용되는 픽셀 수가 감소하면 8x8 하다마드 변환의 횟수가 감소하게 되고 SATD에 필요한 계산 복잡도는 선형적으로 감소하게 된다. 참조 픽셀의 경우 약 2분의 1로 감소했고, 원본 픽셀의 경우 4분의 1의 픽셀만 필요하므로 각 픽셀들을 저장하기 위해 필요한 메모리 공간도 감소하게 된다.

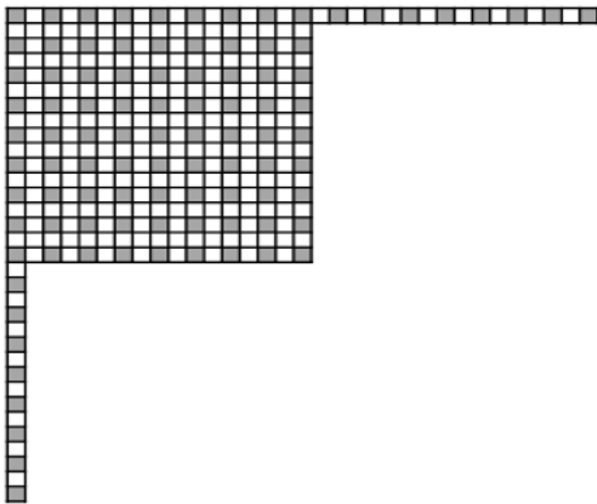


그림 3. 다운 샘플링 기법에서 선택되는 예측 픽셀과 원본 픽셀
Fig. 3. Selected reference pixels and original pixels in Down-sampling scheme

제안하는 기법은 64x64 그리고 32x32 크기의 CU에만 적용한다. 16x16과 8x8CU의 계산 복잡도는 그렇게 크지 않기 때문이다. 또한 16x16과 8x8과 같은 작은 크기의 CU에서는 다운 샘플링이 공간적 특성을 많이 변화시킬 가능

성이 비교적 높다는 문제가 있다. Original Intra prediction 과정에서 수행되는 다른 모든 과정들은 똑같이 수행하는 것으로 한다. 예를 들면, 8x8과 16x16 크기의 CU에 수행되는 Bi-linear filtering은 그대로 8x8과 16x16에만 수행하며 32x32 그리고 64x64크기의 CU에는 수행하지 않는다.

제안하는 다운 샘플링 기법을 사용하여 최적의 모드를 선정하는 과정에서 사용되는 계산식은 수정이 필요하다. (1)에 보인 식은 Original HEVC에서 코스트 계산에 사용되는 식이다. 다운 샘플링을 했기 때문에 총 픽셀 수가 감소하게 되었으므로 예측 픽셀과 원본 픽셀의 차분 값의 총합과 이를 기반으로 계산하는 SATD값도 약 4분의 1정도로 감소하게 된다. 따라서 식 (1)을 그대로 사용하면 SATD 값이 약 4분의 1정도로 감소한 상태이기 때문에 의도하지 않은 결과가 나올 수 있다. 이를 개선하여 식 (2)와 같이 SATD 값을 4배 보정해줌으로써 기존의 RMD 결과와 비슷한 결과가 나오도록 하였다.

$$\text{Cost} = \text{SATD} + \lambda \cdot \text{mode bits} \quad (1)$$

$$\text{Cost} = 4 \cdot \text{SATD} + \lambda \cdot \text{mode bits} \quad (2)$$

III. 다운 샘플링 기법이 적용된 RMD 하드웨어

그림 4는 RMD 하드웨어의 구조를 나타낸다. Original RMD 방식을 사용하고 8x8, 16x16 그리고 32x32 크기의 CU를 처리할 수 있는 하드웨어이다. 8x8, 16x16 그리고 32x32 크기의 CU를 처리하는 모듈들은 서로 같은 구조로 되어 있으며 Neighbor Pixel Buffer 모듈은 참조 픽셀을 저장하는 버퍼로 처리하는 CU의 크기에 따라 버퍼의 크기가 달라진다. Original Pixel Buffer 모듈은 원본 픽셀을 저장하는 버퍼로 CU의 크기에 따라 크기가 달라지는 모듈이다. Smoothing Reference Pixel 모듈은 Intra prediction 과정에서 smoothing 과정을 거친 참조 픽셀을 필요로 하는 모드에 사용될 픽셀을 생성하는 모듈로 smoothing과정을 진행한다. Smoothed Pixel Buffer와 Non smoothed Pixel Buffer는 각각 smoothing이 완료된 참조 픽셀과 smoothing을 하지

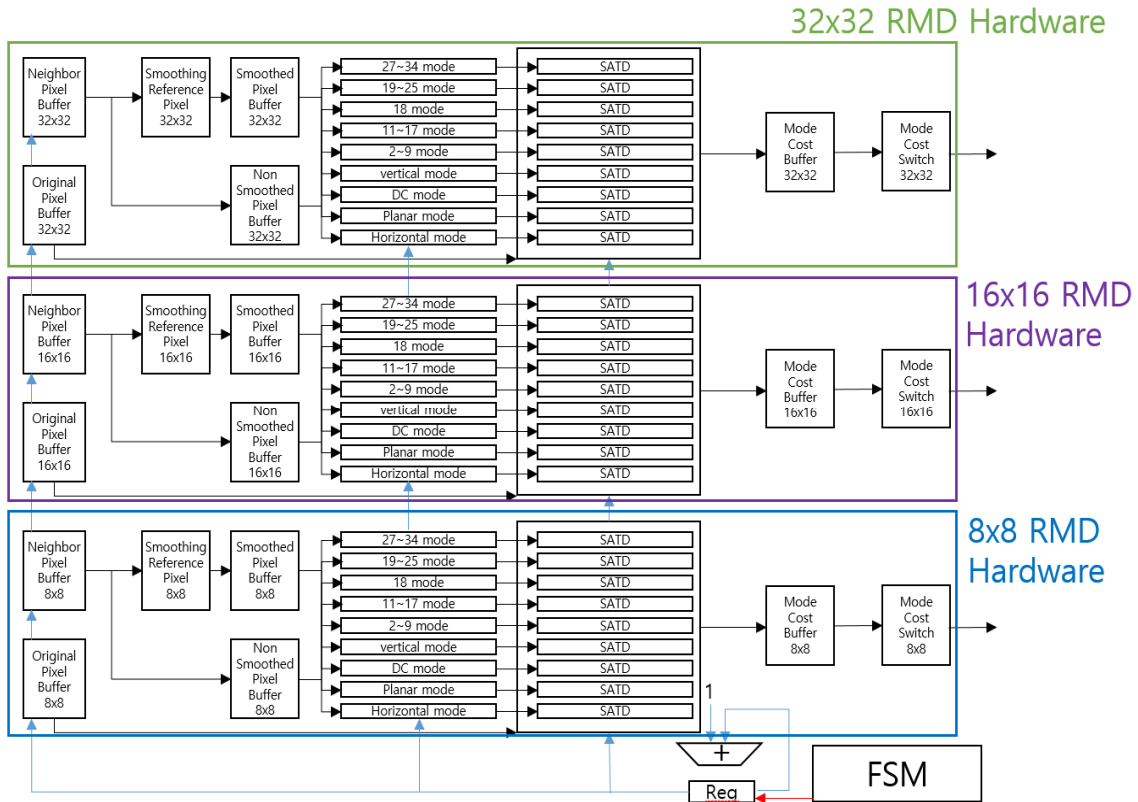


그림 4. Original RMD 하드웨어 구조
 Fig. 4. Original RMD Hardware Architecture

많은 참조 픽셀을 저장하는 버퍼이며 CU의 크기에 따라 다른 크기를 갖는다. 두 버퍼에 저장된 픽셀 값들로 35가지 모드의 예측 픽셀이 생성되고, 생성된 예측 픽셀은 SATD 모듈에서 Original Pixel Buffer에 저장된 원본 픽셀과 함께 SATD 계산을 진행한다. 각 예측 모드의 SATD 값은 Mode Cost Buffer에 저장되며, 35개의 모든 모드의 계산이 종료 되면 Mode Cost Switch 모듈에서 N개의 후보 모드들을 선정한다.

32x32 CTU를 가정할 때 8x8 크기의 CU와 16x16 크기의 CU가 계층적인 형태로 위치하게 된다[6]. 따라서 32x32한 CTU에서 처리해야 하는 CU는 32x32 크기의 CU 1개, 16x16 크기의 CU 4개 그리고 8x8 크기의 CU 16개이다. 그림 4의 하드웨어는 32x32 CTU내의 모든 CU들을 한 번에 처리하는 하드웨어로 동작 스케줄을 보면 그림 5과 같다. 그림 5는 32x32, 16x16 그리고 8x8 크기의 CU를 처리하는

RMD 하드웨어가 병렬적으로 동작하는 스케줄을 나타낸다. 괄호 안의 숫자는 예측기의 개수를 나타내며 32x32, 16x16 그리고 8x8 크기의 CU를 처리하는 예측기가 각각 5개씩 병렬적으로 동작한다는 것을 의미한다. 네모 칸은 7개의 예측 모드를 처리하는 시간을 나타내기 때문에 5개의 예측기가 병렬적으로 동작하여 네모 한 칸 동안 35개의 예측모드를 처리한다는 것을 의미한다. 32x32 CU의 35가지 모드에 대해서 RMD를 수행하는 동안 같은 영역의 16x16 크기의 CU 4개 그리고 같은 영역의 8x8 크기의 CU 16개에 대한 RMD도 동시에 수행을 하게 된다. 모든 예측기의 처리 속도는 8pixels/cycle 이므로 8x8, 16x16 그리고 32x32 크기의 CU들에 대한 RMD과정이 함께 종료된다.

제한하는 기법을 그림 4의 하드웨어 구조에 적용할 경우 32x32 크기의 CU는 16x16으로 다운 샘플링되기 때문에 32x32를 처리를 위한 하드웨어를 제거할 수 있다. 제한하

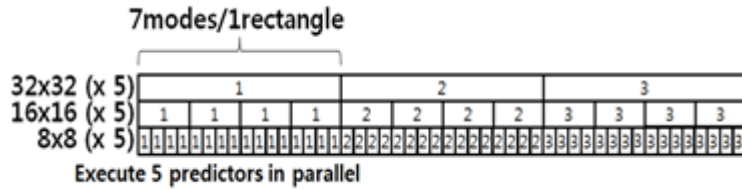


그림 5. RMD 하드웨어의 동작 스케줄
Fig. 5. Timetable of RMD Hardware

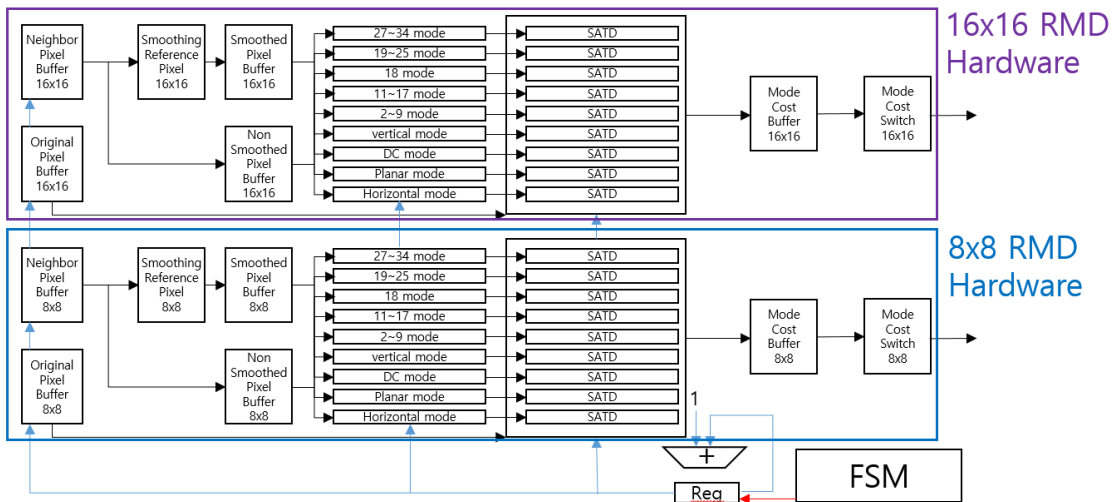


그림 6. 다운 샘플링 기법을 적용한 RMD 하드웨어 구조
Fig. 6. RMD Hardware Architecture of Down-sampling scheme

는 다운 샘플링 기법을 적용한 RMD 하드웨어 구조는 그림 6과 같다. 다운 샘플링 과정에서 추가적인 연산 없이 홀수 위치 또는 짝수 위치에 있는 pixel을 선택하는 과정만 있기 때문에 Neighbor Pixel Buffer와 Original Pixel Buffer 내부에서 필요한 픽셀을 선택하는 과정만 추가되므로 하드웨어 구현에 용이하며 기존 RMD hardware와 비교했을 때 추가되는 로직은 거의 없으며, 가장 큰 32x32 크기의 CU를 처리하는 부분을 제거할 수 있다.

32x32 RMD Hardware 내부 모듈의 메모리 요구량은 다음과 같다. Neighbor Pixel Buffer, Smoothed Pixel Buffer 그리고 Non Smoothed Pixel Buffer 모듈에서 참조 픽셀들을 모두 저장하기 때문에 각 129 Bytes씩 총 387 Bytes의 메모리가 요구된다. Original Pixel Buffer모듈에서는 32x

32크기의 Original Pixels을 저장하기 위해 1,024 Bytes의 메모리가 필요하다. Mode Cost Buffer에서는 각 모드의 Cost값을 저장하며 70 Bytes의 메모리가 필요하다. 제안하는 다운샘플링 기법을 적용함으로써 32x32 RMD Hardware를 제거할 수 있고, Cost Buffer를 제외한 나머지 메모리를 감소시킬 수 있다. 총 2,234 Bytes의 메모리 중 32x32 RMD Hardware에 해당하는 1,471 Bytes의 메모리가 제거된다.

IV. 실험결과 및 결론

제안하는 다운 샘플링 기법을 32x32 CTU를 처리하는 인

트라 예측에 적용하였다. RMD는 32x32크기의 CU를 처리하는 모듈을 제거할 수 있기 때문에 Gate counts가 약 33% 감소했으며, 참조 픽셀과 원본 픽셀 저장에 필요한 메모리가 약 73% 감소하였다. 그리고 RMD에서 대부분의 연산량을 차지하고 있는 SATD 계산 복잡도는 64x64 크기의 CU와 32x32 크기의 CU에서 75%만큼 감소했다.

표 1은 다운 샘플링 기법을 적용할 경우 y, u 그리고 v 성분의 BD-rate 변화를 보인다. Intra-only configuration으로 22,27,32 그리고 37의 양자화 파라미터를 사용하여 100 프레임 씩 인코딩 한 결과이다. 표1에서 BD-rate는 Y, U, V성분에서 각각 0.34%, 0.69% 그리고 0.82%의 BD-rate 증가가 있음을 볼 수 있다. 표 1을 통해 메모리 요구량과 계산 복잡도 감소 이득을 얻었지만 BD-rate 증가는 크지 않다는 것을 알 수 있다.

표 1. 다운 샘플링 기법의 BD-rate
 Table 1. BD-rate of Down-sampling scheme

Class	Sequence	BD-Rate (%)		
		Y	U	V
A	Traffic	0.20	0.36	0.40
	PeopleOnStreet	0.10	0.39	0.45
B	ParkScene	0.07	0.08	0.13
	Cactus	0.14	0.13	0.15
	BasketballDrive	0.64	1.27	1.06
	BQTerrace	0.07	0.17	0.25
E	Kimono	0.37	0.38	0.36
	FourPeople	0.13	0.48	0.50
	Vidyo1	0.34	0.69	0.82
C	Johnny	0.41	0.49	0.08
	KristenAndSara	0.35	0.51	0.62
	BasketballDrill	0.29	-0.01	0.18
	PartyScene	0.01	-0.01	0.06
	BQMall	0.05	0.08	-0.02
	Flowervase	0.22	0.46	0.31
D	Keiba	0.08	0.22	0.33
	RaceHorses	0.04	0.10	0.11
	BlowingBubbles	0.00	-0.07	0.00
	BasketballPass	0.22	0.40	0.34
	BQSquare	-0.01	0.11	0.21
Average		0.34	0.69	0.82

본 논문에서 제안하는 다운샘플링 기법을 Reference Software인 HM13.0에 적용했을 경우 RMD과정에 소요되는 시간을 측정했다. 표2는 Intel i5-4460 @ 3.20GHz CPU에서 실험을 진행했으며 표 1에서 진행한 실험과 같은 종류의 Sequence에 대해 100프레임 씩 인코딩하며 측정한 평균값을 나타낸다. 32x32 그리고 64x64 크기의 블록에서 SATD값 계산에 필요한 연산량은 각각 75%씩 감소했지만 32x32 그리고 64x64 크기에 대한 RMD과정을 보면 수행시간은 각각 약 56.5%, 58.4%만큼 감소했다. 또한 RMD과정에 소요되는 전체 시간(total)을 보면 약 26.4% 감소했음을 알 수 있다.

표 2. HM13.0에서 RMD과정의 평균 소요시간
 Table 2. Average processing time of RMD in HM13.0

	average processing time (micro sec)	
	HM 13.0	Down sampling scheme
8x8 RMD	20.64	19.76
16x16 RMD	58.53	56.00
32x32 RMD	204.59	88.92
64x64 RMD	785.93	326.81
Total	3861.73	2843.13

참 고 문 헌 (References)

- [1] JW Kim, DH Kim, JG Kim, "An Early Termination Algorithm of Prediction Unit(PU) Search for Fast HEVC Encoding", JBE, Vol19, No.5, p628-630, Septemver, 2014.
- [2] EY Won, SI Chae, "An Improved RDO Algorithm for the HEVC intra encoder", KSBE Summer Conference, p123-126, June, 2013.
- [3] Tariq, Junaid, and Sam Kwong. "Hybrid Fast Intra Mode Decision And Early Termination of Prediction Unit(PU) Splitting For HEVC", Systems, Man, And Cybernetics (SMC), 2015 IEEE International Conference on. IEEE, 2015.
- [4] Lainema, Jani, et al. "Intra coding of the HEVC standard." Circuits and Systems for Video Technology, IEEE Transactions on 22.12, 2012
- [5] Zhang, Qiuwen, et al. "A fast and efficient coding unit size decision algorithm based on temporal and spatial correlation." Optik-International Journal for Light and Electron Optics 126.21, 2015
- [6] Lee, Jaeho, et al. "A fast CU size decision algorithm for HEVC." Circuits and Systems for Video Technology, IEEE Transactions on 25.3, 2015

저 자 소 개



장 지 훈

- 2014년 : 인하대학교 정보통신공학부 졸업(학사)
- 2014년 ~ 현재 : 인하대학교 정보통신공학부 재학(석사)
- ORCID : <http://orcid.org/0000-0002-5004-2642>
- 주관심분야 : 영상 압축, SoC 설계



이 채 은

- 2000년 : 서울대학교 컴퓨터공학과 졸업(학사)
- 2002년 : 서울대학교 전기컴퓨터공학부 졸업(석사)
- 2011년 : 서울대학교 전기컴퓨터공학부 졸업(박사)
- 2002년 ~ 2005년 : 삼성전자 SYSTEM LSI 사업부 선임연구원
- 2013년 ~ 현재 : 인하대학교 정보통신공학과(조교수)
- ORCID : <http://orcid.org/0000-0002-7851-1703>
- 주관심분야 : 영상 압축, SoC 설계