

HEVC 인트라 인코더를 위한 RDO 알고리즘의 개선

*원의연 채수익

서울대학교 전기정보공학부

*saintcanel@sdgroup.snu.ac.kr chae@sdgroup.snu.ac.kr

An improved RDO algorithm for the HEVC intra encoder

*Won, Eui-Yeon Chae, Soo-Ik

Department of Electrical Engineering and Computer Science

Seoul National University

요약

High Efficiency Video Coding 비디오 표준은 다양한 분할방식 및 35가지 예측모드를 허용하기 때문에 최적의 분할 및 예측모드를 결정하기 위한 연산량이 많다. 이를 줄이기 위하여 본 논문에서는 CU분할의 결정에 있어 가설검정을 이용하여 early splitting 및 early pruning을 위한 임계값을 설정하고, early splitting의 경우 연산의 결과값이 임계값보다 클 경우, early pruning의 경우 연산의 결과값보다 임계값보다 작을 경우 CU의 분할을 조기에 결정하는 논문[1]의 방법을 차용하여 CU의 분할을 조기에 결정하며, 추가로 비트율 및 왜곡을 연산하는 예측모드의 개수를 줄임으로써 인코더의 성능을 향상시켰다. 또한 예측모드를 결정할 때 RDOQ를 수행하지 않음으로써 예측모드를 빠르게 결정하며, TU의 분할에 있어서도 CBF의 값 및 임계값을 이용한 early pruning 방법을 수행하여 인코더의 성능을 향상시켰다. 실험결과 5.9%의 luma BDRate의 증가가 있었으나, 63.7%의 인코딩 시간이 절감되었다.

1. 서론

High Efficiency Video Coding (HEVC)비디오 표준은 기존의 비디오 표준인 H.264/AVC보다 두 배 이상의 압축률을 목표로 하여 표준화 작업을 진행하고 있으며, 2013년 2월 12차 회의의 결과로 Final Draft International Status (FDIS) 가 제안된 상태이다. HEVC는 연산단위에 해당하는 코딩단위 (Coding Unit; CU), 예측을 수행하는 블록의 단위인 예측단위(Prediction Unit; PU), 잔차신호의 변환을 수행하는 변환단위(Transform Unit; TU)의 분할단위를 가지고 있으며, HEVC의 참조코드인 HM-10.0[2]의 경우 CU 및 PU, TU의 분할을 다음과 같이 정한다.

CU분할의 경우 top-down 순서로 연산을 수행하여 각 CU에 대한 최적의 예측모드 및 TU의 분할을 찾아 비트율 및 원본이미지와 복원이미지 사이의 열화의 정도를 계산하며, 분할한 경우와 분할하지 않았을 때의 CU의 비트율 및 열화의 정도를 비교하여 CU의 분할 여부를 bottom-up 순서로 결정한다.

최적의 예측모드 결정의 경우 HM-10.0은 35가지 예측모드에 전체에 대해 CU, TU의 분할 및 예측모드를 표현하기 위한 비트를 계산하고 원본 이미지와 복원이미지 사이의 열화의 정도를 계산하는 대신 예측모드의 비트 및 예측소와 원본화소와의 열화만을 고려하여 일부의 예측모드를 고르는 low-complexity Rate Distortion

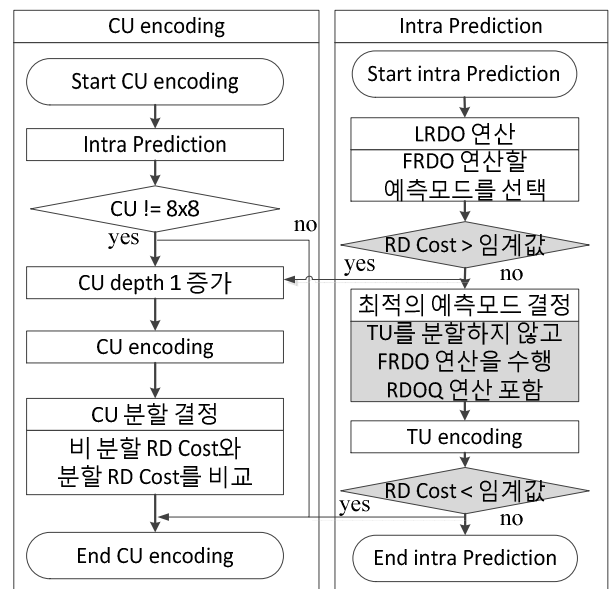


그림 1. 제안하는 인트라 인코더의 RDO 알고리즘

Optimization(LRDO) 연산을 수행한다. 이 때 LRDO 연산을 통하여 8x8 블록 이하의 경우에는 8+Most Probable Mode(MPM), 16x16 블록 이상의 경우에는 3 + MPM의 예측모드를 선택하며, 각각의 선택한 예측모드에 대해 TU를 분할하지 않고,

복원이미지를 생성하여 비트율 및 열화의 정도를 계산하는 Full Rate Distortion Optimization (FRDO)를 수행한다, 이 때 FRDO의 결과값 중 비트율 및 열화의 정도가 제일 작은 예측모드가 최적의 예측모드로 선택된다.

최적의 예측모드를 선택하면, 최적의 예측모드에 대해 TU를 분할해가며 최적의 TU분할을 찾는데, TU 분할도 CU분할과 마찬가지로 top-down 순서로 진행되며, 분할된 4개의 TU와 분할하지 않은 TU의 비트율 및 열화의 정도를 비교하여 TU의 분할여부를 결정하게 된다.

그러나 이러한 HM의 분할 및 예측모드 결정방법은 모든 CU의 분할에 대해 계산을 수행해야 한다는 점과, 예측모드에 대한 FRDO 수행시 최대 10개의 예측모드에 대해 FRDO 연산을 수행해야 한다는 점, FRDO 연산에는 복원된 이미지를 만들기 위하여 전체 인코딩 시간의 20% ~ 35%의 시간이 소요되는 Rate Distortion Optimization Quantization(RDOQ)를 사용한다는 단점이 존재하며, 이를 개선하고자 본 논문은 2장에서 CU의 분할과 관련하여 임계값을 이용한 CU의 early splitting 및 early pruning에 대해 설명하고, FRDO 연산하는 예측모드의 개수를 축소하며 예측모드 결정시 RDOQ 연산을 배제하는 방법을 제안한다. 추가로 CBF 및 임계값을 이용하여 TU분할을 조기에 결정하는 방법을 제안하며, 3장에서는 2장에서 제안한 방법론을 적용하였을 때의 실험결과를 제시하였다. 4장은 논문의 결론에 해당한다.

2. 본론

2.1 CU분할의 조기결정

CU의 분할의 경우 가설검정을 이용하여 LRDO 연산의 결과값이 임계값보다 클 경우 FRDO를 수행하지 않고 early splitting을 수행하며, FRDO Cost가 임계값보다 작을 경우 추가의 CU분할과 관련된 연산을 하지 않는 early pruning을 수행하여 CU의 분할을 조기에 결정하는 방법[1]이 제안된바 있으며, 이를 그림 1의 음영부분으로 표현하였다. 이 때 임계값을 구하기 위하여 유의수준을 설정하는데 유의수준이란 임계값에 의한 early splitting 혹은 early pruning이 틀릴 확률을 의미한다. early splitting의 경우 LRDO 연산의 결과값이 임계값보다 큰데 분할되지 않는 확률에 해당하며, early pruning의 경우 FRDO 연산의 결과가 임계값보다 작는데 분할되는 확률에 해당한다. 유의수준을 설정하면, early splitting의 경우 비분할 LRDO 연산결과 분포의 유의수준에 해당하는 값이 임계값이 되며, early pruning의 경우 분할 FRDO 연산결과 분포의 유의수준에 해당하는 FRDO 값이 임계값으로 설정된다. 이 때 유의수준은 0.1 ~ 0.5의 다양한 값을 가질 수 있으나, 본 논문에서는 BDRate의 증가율이 제일 작은 0.1을 유의수준으로 하여 CU의 분할을 조기에 결정하고자 한다.

2.2 인트라 예측모드 결정시 RDOQ 연산을 배제하는 방법

HM-10.0은 예측모드를 결정하기 위하여 8+MPM 혹은 3+MPM개의 예측모드에 대해 FRDO연산을 수행하여 최적의 예측

모드를 선택하게 되는데, 이 때 FRDO연산에 있어 TU를 분할하지 않은 상태에서 RDOQ를 수행하는 방법을 사용하고 있다. 그러나 RDOQ의 연산은 양자화 계수를 변경해가며 비트율을 고려하여 최적의 양자화 계수를 찾는 과정으로써[3], 만일 RDOQ의 연산을 연산량이 적은 다른 연산으로 대체하였을 때 동일한 예측모드를 결정한다면, 인코더의 성능을 향상시킬 수 있다. 본 논문에서는 예측모드의 결정시 RDOQ 연산을 배제하는 방법을 제안하며, 이 방법이 유용하기 위해서는 RDOQ 연산을 배제한 경우에도 RDOQ 연산을 수행한 경우와 예측모드의 선택이 동일한 경우가 많아야 한다. 이를 측정하기 위하여 동일한 PU에 대해 RDOQ 연산을 수행하였을 때와 RDOQ 연산을 배제하였을 때 같은 예측모드를 결정하는 정도를 측정하였으며, 1080p 이미지인 BasketballDrive 500장, Cactus 500장, Kimono 240장과 WQVGA인 BQSquare 600장을 이용하여 QP 22, 27, 32, 37의 측정값의 평균을 취한결과를 표 1에 나타내었다.

표 1. RDOQ와 non-RDOQ의 예측모드 결정의 불일치 정도 (%)

PU 크기	64	32	16	8	4
basketballDrive	20.65	22.98	19.68	17.90	7.19
Cactus	22.25	25.14	22.64	20.34	8.85
Kimono	21.13	24.63	21.02	14.73	6.07
BQSquare	23.17	27.64	24.77	24.59	14.21
average	21.80	25.10	22.03	19.39	9.08

측정결과 PU의 크기에 따라 정도는 다르지만 약 80%이상의 확률로 RDOQ를 연산하였을 경우와 RDOQ연산을 배제하였을 경우 동일한 예측모드를 선택하기 때문에 본 논문은 예측모드의 결정시 RDOQ 연산을 배제하고자 하며, 이는 그림 1의 회색부분의 RDOQ 연산을 하지 않는다는 의미이다.

2.3 FRDO를 수행하는 예측모드의 개수 축소

HEVC의 인트라 예측모드는 35가지나 되므로, 모든 예측모드에 대해 FRDO를 수행하는 것은 인코딩 시간관점에서 적합하지 않으며, 35가지의 예측모드 중 FRDO를 연산하는 예측모드를 선택하기 위하여 많은 방법론이 제안되어 왔다.[5] ~ [6] 하지만 이러한 방법론의 대부분은 LRDO를 수행하는 대신 엡지를 찾는 연산을 수행하여 FRDO를 연산하는 예측모드의 가짓수를 줄이는 방식을 취하기 때문에 LRDO를 이용하여 CU의 early splitting을 수행하는 본 논문의 방식에 적용하기에는 추가로 엡지 연산을 수행해야 한다는 점에서 적합하지 않다. 따라서 본 논문에 적용할 수 있는 FRDO 연산을 수행하는 예측모드의 개수를 줄이는 방법은 LRDO 연산결과를 이용하는 알고리즘에 한정된다.

이 때 LRDO의 연산결과를 이용하여 현재 연산하는 블록의 텍스처의 특성을 분석하고, FRDO연산을 수행하는 예측모드의 개수를 효율적으로 제어하는 논문으로 [7]이 존재하며, [7]에서는 LRDO의 연산결과를 이용하여 블록크기가 16x16 이상일 때, 만일 LRDO 연산의 결과값이 제일 작은 예측모드가 Planar / DC 라면 연산하는 블록은 방향성이 없다고 가정하여 Planar 및 DC에 대해서만 FRDO 연산을 수행한다. 8x8 블록크기 이하의 경우 항상 MPM에

대해 FRDO 연산을 수행하며, LRDO 연산의 결과값이 제일 작은 2가지의 각도 예측모드가 인접할 경우 2가지의 각도 예측모드에 대해서만 FRDO를 수행하고, 인접하지 않을 경우 3가지의 각도 예측모드에 대해 FRDO를 수행한다. 또한 Planar / DC의 LRDO Cost가 상위 4번째 이내일 경우 Planar / DC에 대해서도 추가로 FRDO를 수행하는 방법을 취하고 있으며, 이를 그림 2 및 3에 표현하였다.

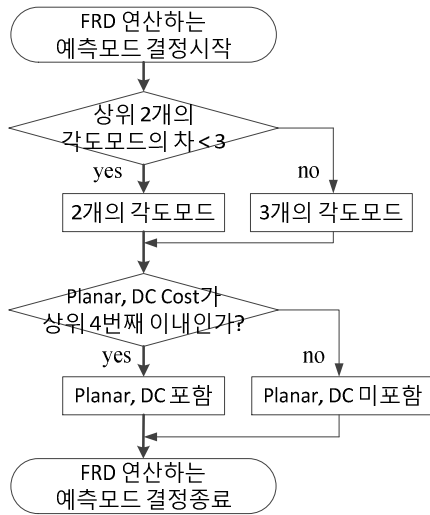


그림 2. 8x8 PU 이상일 때 FRDO 연산하는 예측모드 결정과정

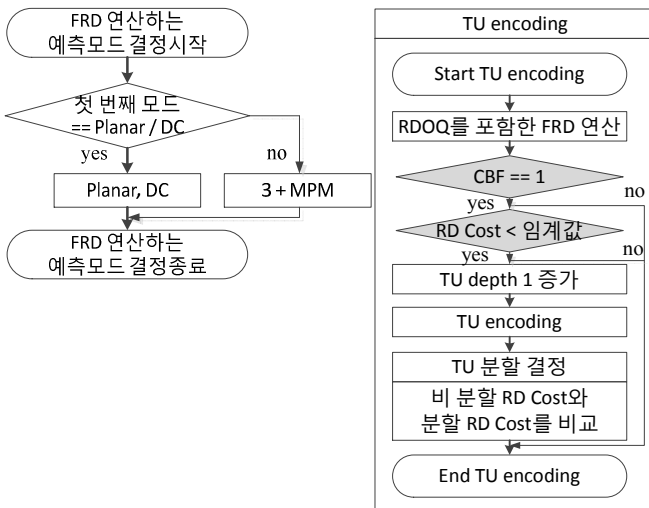


그림 3. 16x16 PU 이상일 때 FRDO 연산하는 예측모드 결정과정 및 TU의 early pruning

최종적으로 제안하는 알고리즘이 FRDO 연산을 수행하는 예측모드의 개수는 표 2와 같다.

표 2. FRDO를 연산하는 예측모드의 개수

PU의 크기	HM-10.0	제안하는 알고리즘[7]	
		최 소	최 대
8x8 이하	8 + MPM	2+MPM	5+MPM
16x16 이상	3 + MPM	2	3+MPM

2.4 TU분할의 early pruning

Coded block flag (CBF) 신호는 TU의 모든 계수가 0일 때를 효율적으로 처리하기 위한 신호로써, CBF가 0일 경우 잔차신호의 전송을 위한 비트가 매우 적기 때문에 비트율이 충분히 작을 것이라 기대할 수 있다. 이러한 성질을 이용하여 인터의 PU모드 예측의 경우 CBF를 이용하여 PU 크기를 조기에 결정하는 방법[4]이 존재하며, 본 논문의 경우 CBF의 정보를 이용하여 만일 CBF값이 0이 등장한다면 TU에 대해 early pruning을 수행하는 방법을 제안한다. 이러한 방법론이 효과가 있는지 알기 위해 CBF가 0인 TU가 등장하였을 때의 분할하지 않은 TU가 최적일 확률을 측정하였으며, 2.1절에서의 실험환경과 동일하게 측정된 실험결과를 표 3에 표현하였다. 측정결과 CBF가 0일 경우 분할하지 않은 TU가 최적일 확률이 약 96% 정도로 높기 때문에 early pruning을 수행하여 추가로 TU분할을 수행하지 않도록 한다.

표 3. CBF가 0일 때 분할하지 않은 TU가 최적일 확률 (%)

TU 크기	32	16	8
basketballDrive	94.28	96.97	98.91
Cactus	91.72	96.16	98.34
Kimono	96.33	98.55	98.78
BQSquare	90.98	95.61	97.85
average	93.33	96.82	98.47

하지만 인트라 모드에서는 CBF의 값이 0인 경우가 적은 것으로 알려져 있기 때문에 CBF가 1일 때도 TU의 분할을 조기에 결정할 수 있는 방법론이 필요하며 이를 위해 [1]에서 사용하였던 가설검정을 TU에도 적용하고자 한다. TU의 경우 LRDO 연산과정이 존재하지 않기 때문에 FRDO 연산의 결과값을 이용한 TU분할의 early pruning만을 수행하도록 하며, 이는 TU의 FRDO 연산의 결과값이 임계값보다 작을 경우 early pruning을 수행한다는 뜻이 된다. 이러한 CBF 및 임계값을 이용한 early pruning 알고리즘을 그림 3에 표현하였으며, 이 때 TU의 분할은 BDRate에 큰 영향을 주지 않는 것으로 알려져 있기 때문에 유의수준을 0.5로 설정하여 많은 경우에 대해 early pruning을 수행하도록 하여 인코더의 속도를 높이고자 한다.

3. 실험결과

2장에서 제안한 알고리즘을 HM-10.0에 추가로 구현하고, HM-10.0 All-intra configuration[8]으로 수행하였으며, QP 22, 27, 32, 37에 대해 1080p 이미지인 BasketballDrive 500장, Cactus 500장, Kimono 240장과 WQVGA인 BQSquare 600장을 이용하여 intel core i7에서 측정하였다. 표 4의 경우 CU의 분할에 대해 유의수준 0.1로 하여 early splitting 및 early pruning을 적용하였을 때의 BDRate[9] 및 시간감소를 나타내는 average time saving(ATS)를 측정된 결과에 해당하며, 2장의 방법론을 단계적으로 적용한 결과를 표 5에 표현하였다. 이 때 ATS는 (1)과 같은 수식으로 표현된다.

$$ATS(\%) = \frac{enc T(HM-10.0) - enc T(prop.)}{enc T(HM-10.0)} \quad (1)$$

표 4. CU분할에 대해 early splitting 및 early pruning을 적용

사용한 이미지	BDRate			ATS
	Y	U	V	
BasketballDrive	5.5%	7.4%	7.0%	50.2%
Cactus	3.8%	2.5%	3.0%	51.7%
Kimono	6.6%	5.1%	4.7%	45.1%
BQSquare	1.6%	0.6%	0.8%	53.8%
average	4.4%	3.9%	3.9%	50.2%

가설검정을 이용한 CU분할의 early splitting 및 early pruning의 경우 Sequence에 따른 BDRate의 변화가 큰 편이며, 전체 인코딩 시간의 절반가량을 줄였음을 알 수 있다.

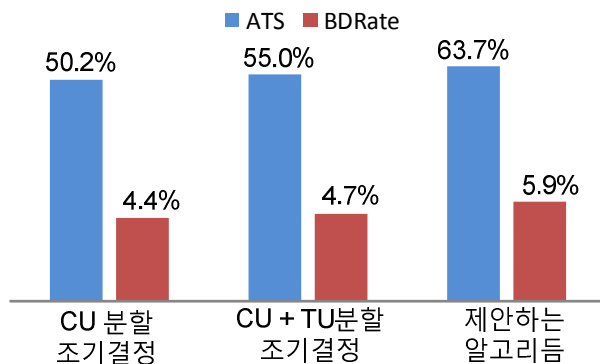


그림 4. CU분할의 조기결정, TU분할의 조기결정을 추가, 제안하는 알고리즘간의 BDRate 및 ATS의 관계

첫 번째 그래프는 표 4에 해당하는 ATS 및 BDRate이며, CU 분할의 조기결정 및 TU분할의 조기결정을 적용하였을 때의 BDRate 및 ATS를 2번째 그래프에 표현하였고, CU분할의 조기결정방법에 TU분할의 조기결정, FRDO연산을 수행하는 예측모드의 축소, 예측모드 결정에 있어서의 RDOQ 연산을 배제하는 방법을 모두 적용한 결과값을 3번째 그래프에 표현하였다. CU분할의 조기결정을 기준으로 2장의 제안하는 알고리즘을 추가할 경우 BDRate의 증가와 ATS간의 trade-off가 발생하였으며, 2장의 내용을 전부 적용할 경우 5.9%의 luma의 BDRate증가가 있었으나 63.7%의 인코더의 속도향상이 있었다. 이는 CU분할을 조기에 결정하는 방법만을 적용했을 때에 비해 1.5%의 luma BDRate 증가가 있었으나, 13%의 추가적인 인코딩 속도 증가가 존재함을 확인할 수 있었다.

4. 결론 및 향후 연구방향

본 논문에서는 인트라 인코더에서 CU의 분할을 조기에 결정하기 위하여 가설검정을 이용한 early splitting 및 early pruning방법을 사용하는 방법론[1]을 발전시켜, 예측모드를 결정할 때 RDOQ의 연산을 배제하여 RDOQ의 연산을 줄이고, FRDO를 수행하는 예측모드의 개수를 줄임으로써 FRDO의 연산을 줄였으며, CBF 및 임계값을 이용한 TU분할의 early pruning방법을 사용하여 인코더의 성

능을 향상시키는 알고리즘을 제안하였다. 제안한 알고리즘은 참조코드인 HM-10.0에 비해 5.9%정도의 luma의 BDRate 증가가 있었으나, 63.7%의 인코딩 시간을 줄일 수 있었다.

참고문헌

[1] S. Cho. M. Kim, "Fast CU Splitting and Pruning for Suboptimal CU Partitioning in HEVC Intra Coding", Circuits and Systems for Video Technology, Vol PP, issue 99

[2] HEVC reference software 10.0

[3] M. Karczewicz, P. Chen, Y. Ye, "R-D Based Quantization in H.264", SPIE vol.7443

[4] R. H Gweon, Y.L Lee, "Early Termination of CU Encoding to Reduce HEVC Complexity", IEICE TRANSACTIONS, vol E95-A No.7, July 2012

[5] Jiang, W., Ma, H., & Chen, Y. (2012, April). Gradient based fast mode decision algorithm for intra prediction in HEVC. In Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on (pp. 1836-1840). IEEE.

[6] da Silva, T. L., Agostini, L. V., & da Silva Cruz, L. A. (2012, August). Fast HEVC intra prediction mode decision based on EDGE direction information. In Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European (pp. 1214-1218). IEEE.

[7] Zhang, M., Zhao, C., & Xu, J. (2012, September). An adaptive fast intra mode decision in HEVC. In Image Processing (ICIP), 2012 19th IEEE International Conference on (pp. 221-224). IEEE.

[8] F. Bossen, "Common test conditions", JCTVC-L1100, Aug. 2012. Stockholm, Jan 2013

[9] G. Bjontegaard, "Calculation of average PSNR differences between RD-Curves", ITU-T SG16 Q.6 Document VCEG-M33, Austin, April 2001.