

하드웨어 구현에 적합한 HEVC 의 CU 단위 율 및 왜곡 예측 방법

이범식, 김문철
한국과학기술원

lbs@kaist.ac.kr, mkim@ee.kaist.ac.kr

A Rate and Distortion Estimation Scheme for HEVC Hardware Implementation

Busmhik Lee, Munchurl Kim

Korea Advanced Institute of Science and Technology (KAIST)

요 약

본 논문에서는 하드웨어의 제한된 자원을 이용하여 HEVC 코덱을 구현할 때 DCT 와 엔트로피 부호화를 사용하지 않고 율 및 왜곡값을 예측하여 고효율의 부호화를 수행하는 방법에 대하여 제안한다. HEVC 는 기존의 부호화기에 비하여 계층적 부호화 구조와 함께 큰 블록 크기를 갖는 DCT 와 엔트로피 부호화를 반복적으로 수행하기 때문에 하드웨어 구현 시 그 복잡도가 매우 크게 증가한다. 먼저 DCT 는 하다마드변환 행렬과 또 다른 정규 직교 변환 행렬의 곱으로 표현될 수 있는 성질을 이용하여 부호화 변환 시 생성된 하다마드변환 행렬에 저 복잡도의 정규 직교 변환 행렬을 곱하여 DCT 변환 계수를 생성한 후 변환 및 양자화를 수행한다. 왜곡값의 경우, 이 때 생성된 양자화 계수와 변환 계수 간의 차이를 변환도메인에서 제곱합을 이용하여 계산하여 역변환을 생략함으로써 복잡도를 감소시킬 수 있다. 또한 텍스처에 대한 비트율 예측은 각 CU 블록내의 양자화 계수의 수를 더하여 계산하여 엔트로피를 수행하지 않고 예측할 수 있다. 그리고 비 텍스처에 대한 비트율 예측의 경우 움직임벡터의 비트에 대한 Pseudo CABAC 코드를 수행하여 예측할 수 있다. 이러한 저 복잡도의 텍스처 및 비 텍스처 비트와 왜곡을 예측함으로써 하다마드변환만을 이용하여 부호화하였을 때에 비해 최대 33%의 비트율 감소를 얻을 수 있었다.

1. 서론

최근에 표준 제정을 완료한 HEVC 는 기존의 비디오 부호화 표준과 달리 일정한 블록 크기를 이용하여 부호화하지 않고 다양한 크기를 갖는 부호화 블록과 변환 블록 크기를 사용하여 계층적 구조를 가지고 부호화를 수행하기 때문에 HD 나 UHD 뿐만 아니라 저해상도 영상까지도 고효율로 부호화할 수 있는 장점이 있다. HEVC 의 CU 와 TU 는 최종적으로 계층적 형태의 쿼드트리 형태를 가지도록 부호화되며, 이러한 형태의 쿼드트리가 결정되기 위해서는 각 노드에서 현재 블록과 하위 4 개의 블록간의 율-왜곡 비용을 비교하여 그 비용이 작은 값이 최적의 블록의 되도록 결정한다. 그러기 위해서는 각 노드에서 반복적으로 율 및 왜곡 값을 계산하여 수행하여야 하기 때문에 하드웨어의 한정된 자원에서는 반복적인 계산으로 인한 복잡도가 급격히 증가한다. TU 도 기존 부호화에서 단일 크기의 변환 커널을 사용하지 않고 4x4 부터 32x32 블록 크기까지 다양한 변환 커널을 사용하기 때문에 RDO 시 변환 커널에 해당하는 DCT 연산을 반복적으로 수행하여야 한다. 또한 변환 및 양자화 이후 수행되는 엔트로피 부호화를 통해 비트율도 계산해야 하기 때문에 CU 의 블록 크기를 결정하기 위하여 수행되는 연산은 매우 복잡해 줄 수 밖에 없다. 본 논문에서는 최적의 CU 깊이를 결정하기 위해 DCT 연산을 크게 감소 시키고 엔트로피 부호화를 수행하지 않으면서 율 및 왜곡 값을 예측하는 저 복잡도 율-왜곡 최적화

방법을 제안한다. 제안 방법에서는, 먼저 DCT 변환 행렬이 하다마드변환과 또 다른 직교 변환의 곱으로 표현할 수 있는 성질을 이용하여 부호소 움직임 벡터 예측 시 생성된 하다마드 변환 계수와 제안하는 직교 변환 커널을 이용하여 DCT 변환 계수를 생성하여 변환 및 양자화를 수행한다. CU 단위 왜곡값을 얻기 위하여 변환도메인 내에 변환 계수와 역양자화계수를 이용하여 왜곡값을 계산함으로써 역변환을 생략함으로써 저 복잡도로 왜곡값을 예측할 수 있다. 또한 텍스처 비트의 경우 CU 내에 0 이 아닌 양자화 계수를 모두 카운트함으로써 CU 단위 비트를 예측할 수 있고, 비 텍스처비트의 경우 각 CU 깊이에서 움직임 벡터가 차지하는 비트의 양이 매우 절대적인 성질을 이용하여 각 CU 깊이에서 예측된 움직임 벡터를 엔트로피 부호화하는 Pseudo 코드를 이용하여 예측한다.

본 논문의 2 장에서는 DCT 가 생성되는 하다마드변환과 직교행렬에 대하여 설명하고, 3 장에서는 제안하는 CU 단위 율 및 왜곡 예측방법을, 4 장에서는 실험결과를 제시하고 5 장에서 결론을 맺는다.

2. 하다마드변환과 DCT

DCT는 HT와 또 다른 직교행렬의 곱으로 이루어 질 수 있다. [3] 즉 이 것은 (1)과 같이 표현할 수 있다.

$$C_M = \frac{1}{\sqrt{M}} A_M H_{w,M} \quad (1)$$

여기서 변환 커널의 $1/\sqrt{M}$ 은 정규화를 위한 과정이고 C 는 DCT 변환 커널, H 는 윌시하다마드 커널이고 A 는 (2)와 같이 표현할 수 있다.

$$A = BQB^T \quad (2)$$

여기서 B 는 비트전환행렬(bit reversal matrix)이고 Q 는 항등행렬과 회전행렬로 이루어진 행렬로 표현할 수 있다. 먼저 $M=8$ 인 경우 Q 는 (3)과 같이 표현할 수 있다.

$$Q_8 = \begin{bmatrix} I_2 & & \mathbf{0} \\ & I_2 & \\ \mathbf{0} & & U_2 \\ & & & U_4 \end{bmatrix} \quad (3)$$

여기서 U_2 는 $-\pi/8$ 회전 행렬이고 U_4 는 (4)와 같이 $7\pi/16$, $3\pi/16$ 의 회전에 이루어진 행렬이다.

$$U_4 = \begin{bmatrix} \cos\left(\frac{7\pi}{16}\right) & 0 & 0 & -\sin\left(\frac{7\pi}{16}\right) \\ 0 & \cos\left(\frac{3\pi}{16}\right) & -\sin\left(\frac{3\pi}{16}\right) & 0 \\ 0 & \sin\left(\frac{3\pi}{16}\right) & \cos\left(\frac{7\pi}{16}\right) & 0 \\ \sin\left(\frac{7\pi}{16}\right) & 0 & 0 & \cos\left(\frac{7\pi}{16}\right) \end{bmatrix} \times$$

$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} \times \begin{bmatrix} \cos\left(\frac{3\pi}{16}\right) & 0 & -\sin\left(\frac{7\pi}{16}\right) & 0 \\ 0 & \cos\left(\frac{3\pi}{8}\right) & 0 & -\sin\left(\frac{3\pi}{8}\right) \\ \sin\left(\frac{7\pi}{16}\right) & 0 & \cos\left(\frac{7\pi}{16}\right) & 0 \\ 0 & \sin\left(\frac{3\pi}{8}\right) & 0 & \cos\left(\frac{3\pi}{8}\right) \end{bmatrix} \quad (4)$$

(1) 은 DCT 변환 커널이 하다마드변환 커널과 또 다른 직교변환 커널의 곱으로 표현할 수 있음을 나타낸다. (2)와같이 행렬 A 는 비트전환행렬과 회전행렬의 곱으로 표현할 수 있다. 행렬 A 를 얻기 위하여 소수점으로 표준되는 회전 계수를 정수로 근사화하고 하드웨어 설계가 용이하도록 하는 U_4 행렬을 (5)와 같이 제안한다.

$$U_4 = \begin{pmatrix} 4 & 0 & 0 & -16 \\ 0 & 12 & -8 & 0 \\ 0 & 8 & 12 & 0 \\ 16 & 0 & 0 & 4 \end{pmatrix} \begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} 12 & 0 & -28 & 0 \\ 0 & 12 & 0 & -28 \\ 28 & 0 & 12 & 0 \\ 0 & 28 & 0 & 12 \end{pmatrix} \quad (5)$$

이 것을 하드웨어 버터플라이 구조를 이용하여 구현하면 그림 4 와 같이 표현할 수 있다.

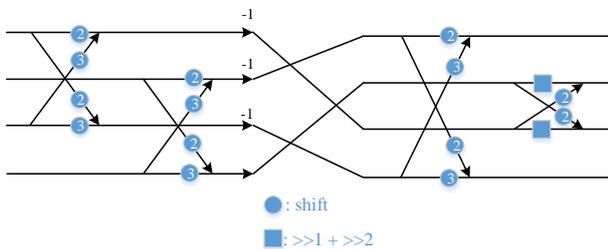


그림 2. Q 의 U_4 행렬의 버터플라이 구조

그림 2 에서 보는 것처럼 덧셈과 시프트 연산만으로 행렬을 구현할 수 있으면 U_4 를 이용하여 A 행렬을 구하면 (6)과 같다.

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 64 & 0 & 32 & 0 & -8 & 0 & 16 \\ 0 & 0 & 12 & 0 & 0 & 0 & 4 & 0 \\ 0 & -24 & 0 & 48 & 0 & 32 & 0 & 16 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 16 & 0 & -32 & 0 & 48 & 0 & 24 \\ 0 & 0 & -4 & 0 & 0 & 0 & 12 & 0 \\ 0 & -16 & 0 & -8 & 0 & -32 & 0 & 64 \end{bmatrix} \quad (6)$$

8×8 변환의 경우 (6)을 수행하기 위하여 그림 2 의 분할 방법을 이용하였을 경우 덧셈 12 회 시프트 17 회만으로 DCT 연산을 수행할 수 있다. 이 것은 8×8 DCT 구현 시 Chen' DCT 를 수행하기 위해 필요한 곱셈 16 회, 덧셈 26 회에 비해 매우 크게 감소한 수치이다.

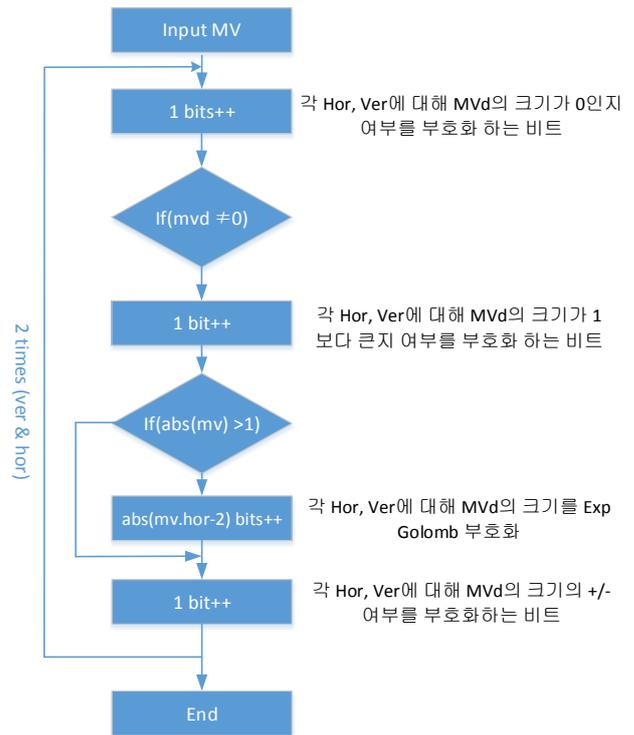


그림 1. 비텍스처 정보를 예측하는 블록다이어그램

3. CU 단위 율 및 왜곡의 예측

2 장에서 DCT 는 하다마드변환(H)과 직교변환행렬(A)의 곱으로 표현할 수 있음을 보였다. 하드웨어에 적합하도록 행렬 A 를 구현하기 위하여 A 의 직교 성질을 제거하고 각 행의 크기를 서로 다른 norm 을 사용하여 구현하였다. 서로 다른 크기의 norm 은 양자화 시에 양자화 행렬을 이용하여 정규화 한다. CU 단위의 텍스처 비트율을 예측하기 위하여 (7)같이 양자화를 수행한다.

$$Y_q(u, v) = (F(i, j) \cdot Q_{k \% 6}(i, j) + f) \gg (N_0 + \text{floor}(QP/6)) \quad (7)$$

여기서 F 는 하드마드 변환계수에 변환 행렬 곱한 계수의 값 (AH)이고, Q 는 양자화 행렬로서 변환 행렬 A 에 대한 정규화

및 양자화를 위한 행렬이다. 양자화된 계수에 대하여 해당 CU 블록내의 모든 0 이 아닌 양자화 계수의 합을 구하여 현재 CU 내의 텍스처 비트를 예측할 수 있다. 비 텍스처 비트의 경우, [4]에서 밝힌 것처럼, 각 CU 내에서 움직임 벡터의 양이 차지하는 비율이 가장 크므로 움직임 벡터의 비트양으로 비 텍스처 비트의 양을 예측한다. 그러나 CABAC 부호화를 수행하지 않고 Pseudo 코드만을 이용하여 비 텍스처의 양을 예측한다. 그림 (1)은 비 텍스처 비트의 양을 예측하는 블록도를 보여준다. 또한 각 CU 에 대한 왜곡은 변환 영역에서 계산되며 (2)를 수행한 후 계수와 복원된 계수간의 제곱합을 계산함으로써 구할 수 있다.

4. 실험 결과

제안하는 윌-왜곡 최적화 방법을 검증하기 위해 HEVC 참조 소프트웨어인 HM12.0 [5]을 사용하였다. Anchor는 HM12.0원본을 사용하였고 비교 대상은 하다마드 변환만을 이용하여 윌-왜곡 최적화를 수행하는 방법이다. 표 1에서 보는 것처럼 제안하는 윌-왜곡 최적화 방법은 하다마드 변환만을 이용하는 것에 비해 월등히 개선된 RD 성능을 보여주고 있다. 모든 DCT 변환을 사용하는 HM12.0 원본에 비해서는 평균 약 10% 정도 성능 하락이 보이지만 SATD만을 사용하는 방법에 비해서는 최대 20% 이상의 성능 개선 효과를 보였다.

표 1. RD 성능개선

Sequence	HM12.0 vs. SATD Only [%]	HM12.0 vs. Prop. [%]
Traffic	28.7	6.8
Kimonol	42.3	16.5
Cactus	39.5	15.7
BasketDrive	37.1	15.0
FourPeople	41.8	8.6
KristenSara	40.8	8.5
BQMall	21.9	10.3
BasketDrill	23.6	7.9
PartyScene	16.5	8.5
BQSquare	20.8	7.4
평균	31.3	10.5

5. 결론

DCT 변환 커널을 도출하기 위하여 하마마드 변환 계수와 저 복잡도 변환 행렬을 이용하였고 이를 이용하여 각 CU 에 대한 윌과 왜곡값을 예측함으로써 HEVC 저 복잡도 윌-왜곡 최적화하는 방법에 대해 제안하였다. 본 방법에서는 부화소 예측 시 생성된 하다마드 변환 계수를 이용하여 DCT 를 수행기 때문에 복잡한 DCT 연산을 피할 수 있을 뿐만 아니라 엔트로피 부호화와 역변환 과정을 생략함으로써 하드웨어 HEVC 부호화기에 적합한 윌 및 왜곡 예측방법이라고 할 수 있다. 본 방법을 사용하면 SATD 만을 이용하였을 때에 비하여 약 20%의 성능 개선을 볼 수 있다.

감사의 글

이 논문은 2014 년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2014R1A1A1007363)

참고문헌

- [1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "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] W. Chen and C. H. Smith, "Adaptive coding of monochrome and color images," *IEEE Trans. Commun.*, vol. COMM-25, pp. 1285-1292, Nov. 1977.
- [3] K. R. Rao and P. Yip, "Discrete Cosine Transform, Algorithm, Advantages, Applications," Academic Press, 1990.
- [4] B. Lee and M. Kim, "Modeling Rates and Distortions Based on a Mixture of Laplacian Distributions for Inter-Predicted Residues in Quadtree Coding of HEVC," *IEEE Signal Processing Letters*, vol. 18, no. 10, pp. 571 - 574, Oct. 2011.
- [5] B. Lee, M. Kim and T. Nguyen, "A Frame-level Rate Control Scheme based on Texture and Non-texture Rate Models for High Efficiency Video Coding" *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 3, pp. 465- 479, Mar. 2014.
- [6] HM reference software 12.0, [Online] Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-12.0