

향상된 해쉬 기법을 통한 HEVC 스크린 콘텐츠 코딩 성능 개선 기법

*허정환 **김일승 ***정제창

한양대학교 전자컴퓨터통신공학과

*hur881122@hanyang.ac.kr

Improved Hashing Method for HEVC Screen Content Coding

*Heo, Jeonghwan **Kim, Ilseung ***Jeong, Jechang

Department of Electronics and Computer Engineering, Hanyang University

요약

본 논문에서는 화면 내 블록 카피 (IntraBC: Intra Block Copy) 예측 기술의 압축 성능 분석과 향상된 해쉬 기법을 통한 HEVC (High Efficiency Video Coding) 스크린 콘텐츠 코딩 성능 기법을 제안한다. 현재 SCM (Screen Content Coding Test Model) 에 채택 된 화면 내 블록 카피 기술에서는 16x16 블록에는 1차원 탐색을 수행하고 8x8블록에서는 해쉬기반 전역 탐색을 수행하여 해쉬가 일치하는 블록들과 RD-Cost를 수행한다. 현재의 해쉬기반 전역탐색에는 기울기 (Gradient) 위주의 해쉬 구성으로 인해 해쉬가 고르게 분포하지 않아, RD-Cost 수행횟수가 과도하게 많아지는 문제가 있다. 제안하는 방법은 전역적 화면 내 블록 카피의 해쉬 구성 방법을 개선함으로써, 기존 SCM-6.1 대비 0.46%의 BDBR 향상을 확인하였다.

1. 서론

최근 인터넷 환경이 빠르게 발전함에 따라 클라우드 컴퓨팅, 게임 등의 분야에서 다양한 디지털 콘텐츠들이 제작되어 배포되고 있다. 또한 다양한 멀티미디어 기기의 보급과 무선 인터넷 망의 보급으로 기존의 콘텐츠 제작, 배포과정과 다른 다양한 사용자층이 형성되었다. 원격 화상회의, 게임, 애니메이션 등 스크린 콘텐츠의 수요가 증가하는 가운데 HEVC (High Efficiency Video Coding)의 비디오 압축 표준 기술이 자연 영상 압축에 맞춰 개발되었기 때문에, 컴퓨터상에서 생성된 그래픽과 텍스트 등으로 이루어진 스크린 콘텐츠 영상을 효율적으로 압축하지 못한다는 문제가 제기되었다[1].

이러한 문제 때문에 2014년 1월 스크린 콘텐츠를 고려한 비디오 압축 표준 제정을 위해 ISO/IEC JTCl/SC29/WC11과 ITU-T SG16 Q6/16에서 CfP (Call for Proposal) 를 공동으로 발행하였다[2].

본 논문은 HEVC SCC (Screen Contents Coding) 에 채택된 기술 중 화면 내 블록 카피 기술 (IntraBC: Intra Block Copy) 을 분석하고 개선된 알고리즘을 제시한다[3]. 화면 내 블록 카피 기술은 그림 1 과 같이 화면 내 복호화 된 영역 중에서 동일한 패턴의 블록을 찾아 현재 블록의 예측으로 사용하는 기술이다.

기존 SCM의 화면 내 블록 카피 방법에는 지역적 블록 벡터 탐색과 전역적 블록 벡터 탐색이 있다. 그 중 전역적 블록 벡터 탐색에서는 16x16 블록에는 1차원 탐색을 수행하고 8x8블록에서는 해쉬기반 전역 탐색을 수행하여 해쉬가 일치하는 블록들에 대해 RD-Cost를 수행한다. 8x8 전역적 해쉬 기반 탐색 알고리즘은 기울기 (Gradient) 위주의 해쉬 구성을 사용하기 때문에 블록의 기울기 값이 낮은 쪽에 해쉬의 분포가 집중되게 되어, 동일 해쉬를 가지는 블록이 많아질수록 RD-Cost 수행횟수가 급격하게 늘어나는 단점이 있다. 따라서 본 논문

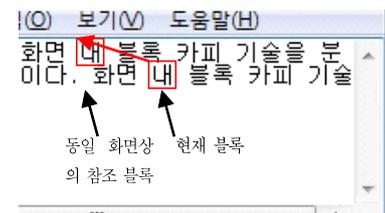


그림 1. 스크린 콘텐츠 화면 내 중복 블록 탐색

에서는 개선된 해쉬 탐색 기법을 제안한다.

본 논문의 구성은 다음과 같다: 2장에서는 기존 방법에 대해 설명하고 문제점에 대해 분석한다. 3장에서는 제안하는 해쉬기반 블록탐색 방법에 대해 설명하고, 4장에서는 제안하는 방법의 성능을 평가한다. 마지막으로 5장에서는 결론 및 분석과 향후 연구 방향을 기술한다.

2. 기존의 화면 내 블록 카피 방법

화면 내 블록 카피 기술은 현재 화면의 복호화 된 내용 중에서 동일한 패턴의 블록을 찾아 블록 벡터 (BV: Block Vector) 로 기록한다. 현재 화면의 복호화 된 영역에서 지역적, 전역적 과정을 통해 최적 블록을 각각 구하고, 서로의 RD-Cost 비교를 통해 최종 블록의 블록 벡터를 구한다.

A. 지역적 블록 벡터 탐색

그림 2 와 같이 지역적 블록 벡터 탐색 (Local Block Vector Search) 은 현재와 좌측 CTU (coding tree unit) 의 복호화 된 영역을 탐색 범위로 한다. 16x16 과 8x8 크기의 CU에서 2Nx2N, 2NxN, Nx2N, NxN의 네 가지 PU 분할 모드를 사용 가능하며 현재의 CTU와

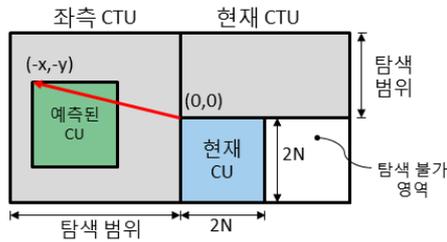


그림 2. 지역적 블록 벡터 탐색의 참조 가능 영역

좌측의 CTU의 복호화 된 화면 영역에서 지역적 수직, 혹은 수평 방향으로 2차원 탐색을 수행한다.

탐색에는 RD-Cost측정을 통해 최적의 블록을 구한다. 먼저 휘도 성분에 대해 RD_Cost_{luma} 를 계산한다:

$$RD_Cost_{luma} = SAD_{luma} + \lambda \times BV_{bits} \quad (1)$$

, 이때 SAD_{luma} 는 휘도성분 차이의 절대 합을 의미하고 λ 는 라그랑지안 승수, BV_{bits} 는 압축된 블록 비트길이를 의미한다.

휘도성분의 4개의 최적 후보 블록을 선정 한 뒤, 각각의 블록에 대해 식 (2)를 사용하여 색차 성분도 고려한 RD-Cost를 구하고, 하나의 지역적 최적 블록을 선정한다:

$$RD_Cost = SAD_{luma} + SAD_{chroma} + \lambda \times BV_{bits} \quad (2)$$

, SAD_{chroma} 는 색차성분 차이의 절대 합을 의미한다.

B. 전역적 블록 벡터 탐색

전역적 블록 벡터 탐색 (Global Block Vector Search) 에서는 CU 사이즈 마다 압축 과정이 다르다. 16x16 크기의 CU에서는 2Nx2N 크기의 PU로만 탐색이 가능하며 수직, 혹은 수평 방향으로의 1픽셀 선단 위의 1차원 방향으로 탐색하는 방법으로 현재 블록과 일치하는 블록을 찾는다. 8x8 크기의 CU에서는 2Nx2N 크기의 PU만 사용이 가능하며 해쉬 기반 탐색을 사용하여 탐색의 복잡성을 줄인다. 그림 3 과 같이 해쉬는 록업테이블과 연결리스트 구조로 이루어져 있어 해쉬가 일치하는 블록을 빠르게 찾는다.

전역적 블록 벡터 탐색 기술은 현재 부호화 중인 8x8 크기의 블록 해쉬를 구하고, 복호화 가능한 이전 블록들의 해쉬 값과 일치하는 블록을 찾는다. 해쉬 값이 일치하는 블록에 대해 RD-Cost 비교를 통해 전역적 최적의 블록을 선택한다.

해쉬는 블록에 대한 평균(DC)과 기울기로 이루어져 있으며 해쉬(H)는 다음과 같다:

$$H = MSB(DC0,3) \ll 13 + MSB(DC1,3) \ll 10 + MSB(DC2,3) \ll 7 + MSB(DC3,3) \ll 4 + MSB(Grad,4) \quad (3)$$

식 (3)에서 $MSB(X,n)$ 는 X의 최상위 비트 n개를 취하는 함수이다. 평균값 $DC0, DC1, DC2, DC3$ 은 그림 4 와 같이 각 위치의 4x4서브블록 평균값이며, 기울기($Grad$)는 다음과 같다:

$$Grad = \sum_{y=1}^4 \sum_{x=1}^4 \frac{|p(x,y) - p(x-1,y)| + |p(x,y) - p(x,y-1)|}{2} \quad (4)$$

, $p(x,y)$ 는 해당 좌표의 원본 픽셀 값을 의미한다.

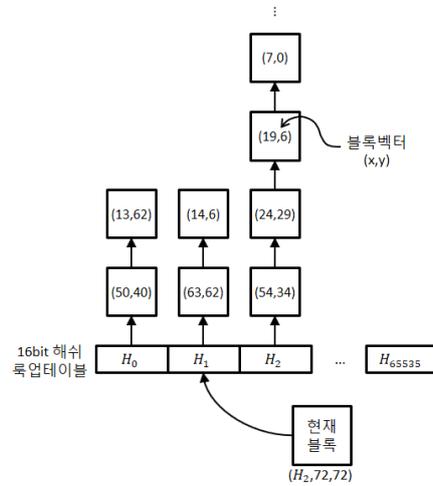


그림 3. 연결리스트로 구현된 16비트 해쉬

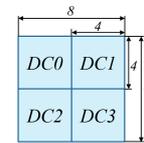
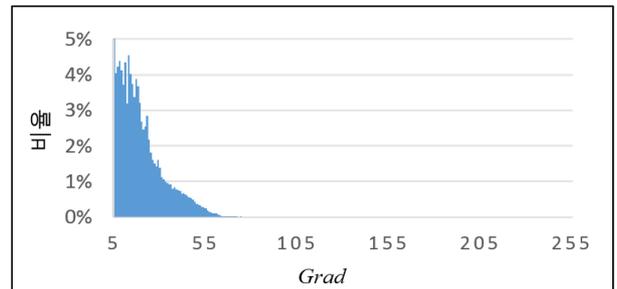
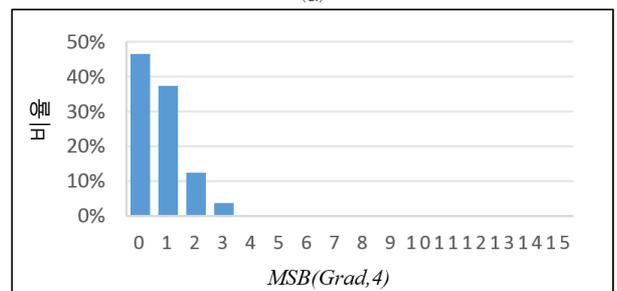


그림 4. 8x8 블록의 DC 값 구성



(a)



(b)

그림 5. (a) SlideEditing Sequence의 8x8 블록의 초기 Grad 분포 (b) MSB(Grad,4)를 수행한 결과

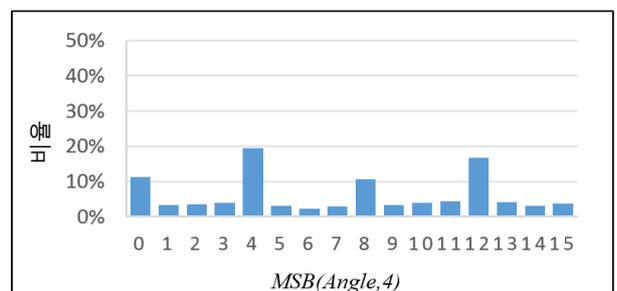


그림 6. SlideEditing Sequence의 8x8 블록 Angle 분포

그림 5의 (a)는 SlideEditing Sequence의 식 (4) Grad분포이고 (b)는 $MSB(Grad,4)$ 를 수행한 결과이다. 해쉬의 Grad의 분포가 0에 가까운 값으로 밀집한걸 알 수 있다. Grad가 0인 블록들은 Grad가 0인 다른 모든 블록과 RD-Cost 계산을 수행하므로 계산 량이 증가한다. 이를 해결하기 위해 추가적인 해쉬 구성을 하고자 한다.

3. 제안하는 방법

본 논문에서 제안하는 알고리즘은 크게 세 가지 방법으로 나눌 수 있다. 먼저 해쉬 구성을 변경하는 방법과 인근 해쉬를 추가적으로 탐색하는 방법 및 빠른 탐색 종료 방법이다.

A. 해쉬의 구성

기존의 해쉬 구성은 식 (3) 과 같이 구성되어 있고 해쉬가 일치하는 블록 간 RD-Cost 비교를 수행한다. 하지만 Grad 정보는 4비트에 불과하며, Grad의 분포는 그림 5의 (b)와 같이 낮은 값에 밀집한 특성이 있어 RD-Cost 비교 횟수가 많다.

따라서 그림 6 과 같이 블록의 각도를 기반으로 한 해쉬를 제안한다. 블록의 각도를 기반으로 한 해쉬는 기존의 Grad보다 고르게 분포해 있어 불필요한 RD-Cost 수행 횟수를 줄일 수 있음을 알 수 있다. 블록의 각도(Angle)는 다음과 같이 계산한다:

$$Angle = \tan^{-1} \left(\frac{w(x,y) \times (p(x,y) - p(x,y-1))}{w(x,y) \times (p(x,y) - p(x-1,y))} \right) \quad (5)$$

, $w(x,y)$ 는 σ 가 1.5인 가우시안 정사각행렬이다. 블록의 중앙에 가중치를 더 줌으로써 사각형 모양의 블록을 원형에 가깝게 처리가능하다. 최종적으로 식 (6) 과 같이 8x8블록의 해쉬(H)를 다음과 같이 설정한다:

$$H = MSB(Angle,4) \ll 12 + MSB(DC,6) \ll 7 + MSB(Grad,6) \ll 2. \quad (6)$$

B. 인근 해쉬의 탐색

기존의 SCM은 블록의 해쉬가 완전히 일치하는 블록들에 대해서만 RD-Cost 비교를 수행하였다. 이 경우 비슷한 블록이지만 해쉬가 일치하지 않는 경우가 발생 할 수 있다. 예로 그림 7 과 같이 비슷한 DC1 값을 가지더라도 $MSB(DC1,3)$ 이후에는 해쉬가 다른 경우가 발생할 수 있다. 이러한 해쉬 불일치를 해결하기 위해 해쉬간 거리가 비슷한 인근 해쉬에 대해서도 모두 RD-Cost 비교를 수행한다. 현재 블록 해쉬의 Angle, DC, Grad 값에 대해 ± 1 의 해쉬를 생성하여 해쉬가 일치하는 블록을 추가로 탐색한다.

C. 빠른 해쉬 탐색 종료

본 절에서는 빠른 해쉬 탐색 종료 방법을 제시한다. 기존 SCM의 전역적 해쉬 탐색의 경우 그림 8의 벽면과 같이 동일 무늬가 반복될 때 해쉬가 모두 일치하여 RD-Cost 수행 횟수가 급격하게 증가한다.

이를 해결하기 위해 현재 영상의 전역적 해쉬 탐색 범위의 면적에 비례한 임계치 $th1$ 과 $th2$ 를 설정한다. 전체 RD-Cost 수행 횟수가 $th1$ 높을 경우 해쉬 탐색을 중단하고 다음 블록의 부호화 과정으로 진

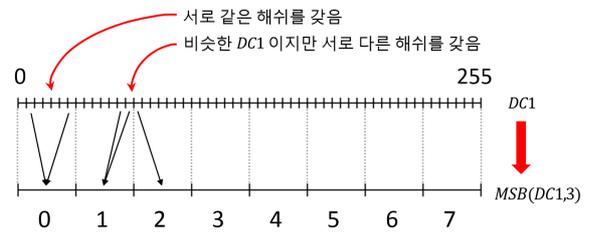


그림 7. DC1을 예로 든 해쉬 간 불일치 예

행한다. 또한 동일 해쉬 내에서의 RD-Cost 수행 횟수가 $th2$ 를 넘을 경우에는 다음 인근 거리의 블록 해쉬로 이동하고 해쉬 탐색을 계속



그림 8. 동일 해쉬를 가지는 반복적인 패턴의 구조물 BQTerrace sequence의 일부

수행하여 다양한 블록을 탐색한다. $th1$ 과 $th2$ 는 다음과 같이 구한다:

$$\begin{aligned} th1 &= (\Phi_{global} + \Phi_{total}) \times 4 \\ th2 &= (\Phi_{global} + \Phi_{total}) \end{aligned} \quad (7)$$

, 식 (7)에서 Φ_{global} 는 전역적 해쉬 탐색이 가능한 범위에 비례한 상수를 의미하고 Φ_{total} 는 전체 영상 범위에 비례한 상수이다. 전역적 해쉬 탐색은 영상의 부호화 진행 정도에 따라 참조 가능한 해쉬의 개수가 다르다. 따라서 Φ_{global} 은 부호화 진행 정도에 따라 임계치를 점진적으로 증가시키기 위해 실험적으로 구한 값이다. Φ_{total} 은 영상의 해상도에 비례한 임계치를 설정하기 위한 값이다.

전체 알고리즘 도식은 그림 9 와 같다.

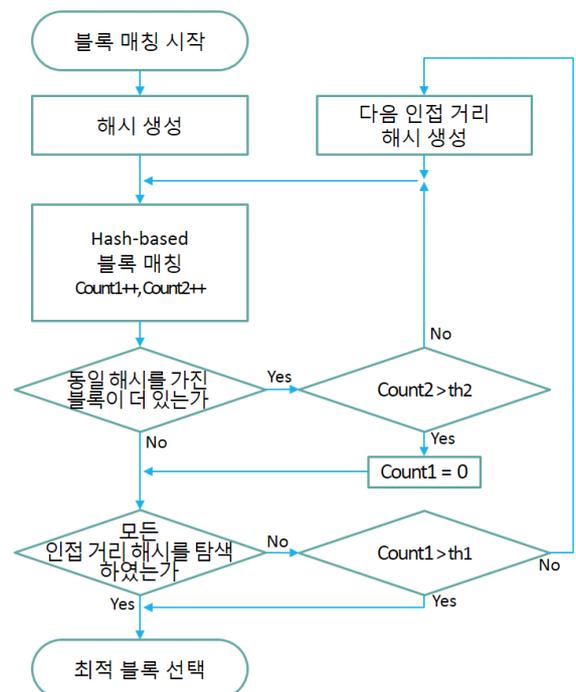


그림 9. 임계치를 통한 빠른 탐색 종료 도식

표 1. HM-RExt와 SCM, 그리고 제안하는 방법의 BDBR, ATS 비교

Category	Sequences	Resolution	실험 1		실험 2	
			BDBR	ATS(%)	BDBR	ATS(%)
Natural	BQTerrace	1920x1080	-1.09	-308.88	-0.04	-16.68
	Kimono1	1920x1080	3.13	-138.49	-0.02	9.05
	PartyScene	832x480	1.16	-238.28	-0.06	2.12
	RaceHorses	832x480	0.27	-340.72	0.01	-0.07
Natural Average			0.87	-256.59	-0.03	-1.40
Screen Content	MissionControlClip3_8bit_444	1920x1080	-67.74	-175.45	-0.78	-7.24
	sc_desktop_8bit_444	1920x1080	-83.70	-188.95	-0.42	-12.15
	sc_web_browsing_8bit_444	1280x720	-76.68	-149.61	-0.56	-3.00
	SlideEditing	1280x720	-47.82	-189.22	-1.81	-9.12
Screen Content Average			-68.99	-175.81	-0.89	-7.88
Total Average			-34.06	-216.20	-0.46	-4.64

4. 실험 결과

A. 실험 조건

실험 평가를 위해 스크린 콘텐츠 영상뿐만 아니라 여러 가지 특성을 가진 영상들 또한 실험 하였다. 실험 환경은 HM-16.8+SCM-6.1 All intra 프로파일을 사용하였고[4], 객관적 부호화 성능 평가를 위해 Bjøntegaard-Delta 측정방법을 사용하였다[5]. 실험 1 은 HM-RExt와 SCM과의 BDBR, ATS(Average Time Saving)를 비교하였고, 실험 2는 SCM과 제안하는 방법으로 나누어 표 1 에 나타내었다. 실험 영상은 모두 0-49 화면을 부호화 하여 실험하였으며 평균 시간 감소율 ATS는 다음 수식을 사용한다:

$$ATS(\%) = \frac{Enc.time(anchor) - Enc.time(proposed)}{Enc.time(anchor)} \cdot (8)$$

B. 실험 결과

실험 1은 HM-RExt과 SCM 간의 비교를 나타낸 것이다. 스크린 콘텐츠 특성이 강한 영상에서 높은 부호화 효율을 보였다. 전체 평균 34.06% 만큼 BDBR이 감소했으나 자연영상에서는 0.87% BDBR 증가했다. 이는 SCM의 시그널 비트가 증가했기 때문이다.

실험 2는 SCM과 제안하는 방법 간의 비교를 나타낸 것이다. 자연 영상에서 부호화 효율 향상을 예상했으나 기존의 SCM과 0.03%만큼 만의 BDBR 감소를 보였다. 특히 BQTerrace 영상의 부호화 속도가 16.68% 감소했고, 이는 반복적인 구조물의 영향으로 RD-Cost 수행 횟수가 증가했기 때문이다. Screen Content 영상에서는 평균적으로 0.89% 의 추가 BDBR 감소를 보였고, SlideEditing 영상이 가장 높은 부호화 효율을 보였다. 전체 영상의 BDBR은 평균 0.46% 감소하였다.

5. 결론

본 논문에서는 SCM의 화면 내 블록 카피 기술의 전역적 블록 벡터 탐색 방법의 해쉬 구성과 탐색 방법을 제안했다. 기존의 SCM에는 해쉬기반 전역탐색이 기술기 위주의 해쉬 구성으로 해쉬의 분포가 고르지 않아 RD-Cost 횟수가 많아지는 문제가 있었고, 해쉬의 분포를 균일화하기 위해 블록의 각도를 추가 하여 해쉬를 구성했다. 제안하는 알고리즘은 이전의 알고리즘에 비해 평균 0.46%의 BDBR 향상을 이루었다.

감사의 글

“이 연구는 방위사업청 및 국방과학연구소의 재원에 의해 설립된 신호 정보 특화연구센터 사업의 지원을 받아 수행되었음.”

참 고 문 헌

- [1] G. Sullivan, J. 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] ITU-T Q6/16 and ISO/IEC JTC1/SC29/WG11 document N14175, “Joint Call for Proposals for Coding of Screen Content,” San Jose, USA, Jan. 2014.
- [3] M. Budagavi, and D.-K. Kwon, “Intra motion compensation and entropy coding improvements for HEVC screen content coding,” in Proc. of Picture Coding Symposium (PCS), pp. 365-368, San Jose, U.S.A., December 2013.
- [4] HM-10.1+RExt-3.0 software, Developed by JCT-VC, https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.7+SCM-6.1
- [5] G. Bjøntegaard, “Calculation of average PSNR differences between RD curves,” *Video Coding Experts Group (VCEG), VCEG-M33*, Austin, Texas, U.S.A., April, 2001.