

## HEVC의 GPB 슬라이스에서 양예측 모드의 동일 움직임 정보에 대한 성능 향상 방안

김 상 민<sup>a)</sup>, 김 경 용<sup>a)</sup>, 박 광 훈<sup>a)†</sup>, 김 휘 용<sup>b)</sup>, 임 성 창<sup>b)</sup>, 이 진 호<sup>b)</sup>

### Coding Efficiency Improvement for Identical Motion Information of Bi-prediction Mode within the GPB Sllice of HEVC

Sang Min Kim<sup>a)</sup>, Kyung Yong Kim<sup>a)</sup>, Gwang Hoon Park<sup>a)†</sup>, Hui Yong Kim<sup>b)</sup>,  
Sung Chang Lim<sup>b)</sup>, and Jin Ho Lee<sup>b)</sup>

#### 요 약

본 논문에서는 현재 표준화가 진행 중인 HEVC(High Efficiency Video Coding)에서 양예측(bi-predictive)모드에 존재하는 문제점을 거론하고 문제점에 대한 해결방안을 제시하여 부호화 효율을 증가시키고 계산 복잡도를 감소시키는 방법을 제안한다. 현재 HM 3.0에서는 양예측을 사용하는 블록에서 L0 움직임 정보와 L1 움직임 정보가 동일해지는 경우가 빈번히 발생한다. 본 논문에서는 이러한 현상이 발생하는 경우 L1의 움직임 벡터를 현재 블록의 주변 블록의 (0,0)이 아닌 L0 움직임 벡터로 대체 하고, 여전히 L0 움직임 벡터와 L1 움직임 벡터가 동일할 경우 예측모드를 단예측으로 변경하여 부호화 성능을 향상 시키고 계산 복잡도를 감소시키는 방법을 제안하였다. 실험 결과, LD(Low-Delay) 실험조건인 경우 기존 대비 복호화기의 수행시간을 2% ~ 5% 감소시키고 부호화 성능을 약 0.3% ~ 0.5% 향상 시켰다.

#### Abstract

This paper proposes the method which reduces complexity and improves coding efficiency by solving a problem of HEVC bi-prediction. In current HM 3.0, it is frequently occurred that L0 motion information and L1 motion information are identical in blocks which are bi-predicted. In this case, L1 motion vector is replaced by non-zero motion vector which belongs to first available neighbor block of current block. If they are still identical, prediction mode is replaced by uni-prediction. As an experimental result, in LD(Low-Delay) case, decoding time is reduced roughly 2%~5% and coding gain is roughly 0.3%~0.5% compared with the HM 3.0 anchor.

Keyword: HEVC(High Efficiency Video Coding), Bi-prediction, Merge

a) 경희대학교 전자정보대학 컴퓨터공학과

Media Lab., College of Electronics and Information, Kyung Hee University

b) 한국전자통신연구원

ETRI

† 교신저자 : 박광훈(ghpark@khu.ac.kr)

※ 본 연구는 방송통신위원회의 ETRI 연구지원 사업[KCA-2011-(11921-02001), 무안경 다시점 3D지원 UHD TV 방송 기술개발]과 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업(NIPA-2011-(C1090-1111-0001))의 연구결과로 수행되었음.

· 접수일(2011년10월10일), 수정일(2011년11월16일), 게재확정일(2011년11월16일)

## I. 서론

고화질, 고해상도 콘텐츠의 엄청난 데이터량을 처리하기 위해서 기존 H.264/AVC<sup>[1]</sup>를 대체할 새로운 동영상 부호화 표준의 필요성이 대두 되었다. 이러한 요구에 부합하기 위하여 ISO/IEC JTC1/SC29/WG11 MPEG과 ITU-T SG16 Q6 VCEG은 H.264/AVC 기술표준 대비 2배 정도의 압축 효율을 목표로 한 차세대 동영상 부호화 기술인 HEVC (High Efficiency Video Coding)<sup>[2]</sup>의 표준화를 진행하고 있다.

HEVC에서는 화면 간 예측 시 단예측을 사용하여 부호화된 슬라이스를 P 슬라이스, 양예측을 사용하여 부호화된 슬라이스를 B 슬라이스라 한다. 양예측을 사용할 경우 해당 블록은 최대 두 개의 움직임 정보(움직임 벡터, 참조 픽처 번호)를 가질 수 있다. 참조 픽처는 참조픽처 리스트에 의해 관리되며, 일반적으로 순방향 참조(과거영상)는 리스트0(list 0; L0)에, 역방향 참조(미래영상)는 리스트1(list 1; L1)에 할당된다.

HEVC의 저지연(Lowdelay)환경에서는 화면 간 예측 효율을 높이기 위해 P 슬라이스 대신 GPB(Generalized P and B)<sup>[3]</sup> 슬라이스를 사용한다. GPB 슬라이스는 일반적인 B 슬라이스와 마찬가지로 하나의 블록 당 최대 2개의 움직임 정보를 가질 수 있다. 하지만 L0 참조 픽처 리스트와 L1 참조 픽처 리스트는 항상 동일해야 하며, 참조 픽처 리스트에 저장되는 영상들은 현재 영상보다 시간적으로 과거 영상으로만 이루어져야한다. 본 논문에서는 GPB 슬라이스처럼 순방향 예측만이 가능한 B 슬라이스를 순방향 B 슬라이스(forward B slice)라 정의 한다. 이러한 순방향 B 슬라이스에서 L0 움직임 정보와 L1 움직임 정보가 동일하게 되는 경우가 주로 발생하고, 그 결과 똑같은 움직임 보상 과정을 불필요하게 반복하여 수행하는 결과를 야기 시킨다. 따라서 본 논문에서는 이러한 문제에 대한 원인을 분석하고 이를 해결하여 계산 복잡도를 감소시키기 위한 방안을 제시한다.

본 논문의 구성은 다음과 같다. II장에서는 기존 양예측의 문제점에 대하여 설명하고, III장에서는 문제점을 해결하기 위한 방법을 제안한다. IV장에서는 제안하는 방법의 성능을 측정하고, 마지막으로 V장에서 결론을 기술한다.

## II. 기존 양예측의 문제점

[그림 1]은 표준화가 진행 중인 HEVC의 참조 모델 HM3.0<sup>[2]</sup>을 이용해서 저지연 환경으로 부호화한 영상이다. 저지연 환경으로 부호화를 수행할 경우, 단예측과 양예측이 모두 사용되며, 양예측의 경우 순방향 양예측을 이용하여 부호화되기 때문에 각 PU(Prediction Unit)<sup>[2]</sup>마다 최대 두 개의 움직임 정보가 존재할 수 있다. [그림 1]에서 “11-24,-23|0,0”는 각각 “POC|움직임벡터|움직임벡터의 차이”를 나타내며, 위쪽은 L0 움직임 정보를 나타내고 아래 쪽은 L1 움직임 정보를 나타낸다. 그리고 “n.a”는 움직임 정보가 없음을 의미한다. Merge<sup>[2]</sup> 모드로 부호화 되었을 경우 “Merge : merge Index”가, AMVP<sup>[2]</sup> 모드 부호화 되었을 경우 “AMVP : |L0 참조픽처번호, L1 참조픽처번호”가 표시된다. [그림 1]을 보면, L0움직임 정보와 L1움직임 정보가 동일한 블록이 다수 존재한다는 것을 알 수 있다. L0 움직임 정보와 L1 움직임 정보가 동일하다는 의미는 L0의 움직임 벡터와 L1의 움직임 벡터가 같고 동일한 POC(Picture Order Count)의 영상을 참조영상으로 사용한다는 것이다.

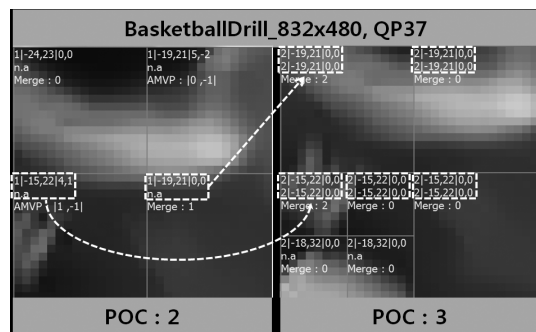


그림 1. 부호화된 영상의 움직임 정보  
Fig 1. Motion information of encoded picture

[표 1]은 HEVC 공통실험조건<sup>[4]</sup>에서 사용하는 실험 영상을 이용하여 순방향 B 슬라이스에서 L0 움직임 정보와 L1 움직임 정보가 같아지는 블록의 비율을 QP(Quantization Parameter)별로 나타낸 표이다.

[표 1]에서 LDHE, LDLC의 실험 조건의 경우 L0 움직임

정보와 L1움직임 정보가 동일한 경우가 전체의 약 30%를 차지한다. 이러한 동일 움직임 정보는 merge/skip 부호화 모드<sup>[2]</sup>에서 시간적 움직임 벡터를 결정하는 과정에서 발생된다. [그림 1]의 표시된 부분과 같이, 참조 영상에서 현재 블록과 대응하는 블록에 L0 움직임 정보가 존재하고 L1 움직임 정보가 존재하지 않으면, 현재 블록의 시간적 움직임 벡터의 L0 움직임 정보와 L1 움직임 정보 모두 참조 영상에서 현재 블록과 대응하는 블록의 L0 움직임 정보로 설정된다. 이는 merge/skip 부호화모드에 의해서 공간적으로 확산된다. 이러한 L0 움직임 정보과 L1 움직임 정보의 동일성은 움직임 보상 단계의 중복으로 인한 계산 복잡도의 증가를 유발한다.

표 1. 순방향 B 슬라이스에서 L0 움직임 정보와 L1움직임 정보가 같아지는 블록의 비율  
Table 1. Ratio of blocks which have identical motion informations in forward B slice

	QP	LD HE(%)	LD LC(%)
Class A	37		
	32		
	27		
	22		
Class B	37	38.3%	38.6%
	32	28.8%	28.4%
	27	21.0%	19.8%
	22	11.6%	9.2%
Class C	37	31.3%	33.6%
	32	21.0%	23.2%
	27	14.2%	16.2%
	22	9.3%	10.1%
Class D	37	32.9%	35.0%
	32	22.6%	23.7%
	27	15.4%	16.3%
	22	10.0%	10.2%
Class E	37	71.0%	70.2%
	32	55.1%	57.7%
	27	41.0%	44.8%
	22	29.6%	34.4%
Average		28.3%	29.5%

### III. 문제점을 해결하기 위해 제안하는 방법

본 논문에서는 II장에서 밝힌 문제점을 해결하기 위해서 다음과 같은 세 가지 방법을 제시한다.

- 1) L0 움직임 정보와 L1 움직임 정보가 동일하면, 단예측으로 변경하는 방법
- 2) 중복된 L1 움직임 벡터를 주변 블록의 움직임 벡터로 대체하는 방법
- 3) 중복된 L1 움직임 벡터를 주변 블록의 움직임 벡터로 대체하고, 그 이후에도 L0 움직임 정보와 L1 움직임 정보가 동일하면 단예측으로 변경하는 방법

1) 첫 번째 방법은 merge/skip 부호화 모드의 시간적 움직임 정보를 결정할 때 L0 움직임 정보와 L1 움직임 정보를 유추한 후 동일 움직임 정보 여부를 판단하여 동일한 움직임 정보일 경우, 해당 블록의 움직임 정보 예측 모드를 양예측에서 단예측으로 변경하는 방법이다.

2) 두 번째 방법은 L0 움직임 정보와 L1움직임 정보가 동일할 경우, 현재 블록의 주변블록 중 L0 움직임 벡터가 (0,0)이 아닌 블록을 [그림 2]의 A(왼쪽), B(위쪽), E(왼쪽 위)의 순서대로 탐색하여 가장 먼저 찾아지는 블록의 L0 움직임 벡터를 현재 블록의 L1 움직임 벡터로 사용한다.

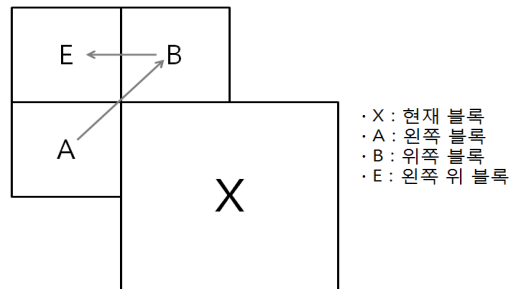


그림 2. 주변 블록의 탐색 순서  
Fig 2. Search order of neighbor blocks

3) 세 번째 방법은 첫 번째 방법과 두 번째 방법을 함께 수행하는 방법이다. L0 움직임 정보와 L1 움직임 정보의 동일성 여부를 판단하고, 동일한 움직임 정보일 경우 주변 블록에서 (0,0)이 아닌 움직임 벡터를 가지고 있는 블록을 찾아 그 블록의 L0 움직임 벡터를 현재 블록의 L1움직임 벡터로 사용한다. 그 후 L0 움직임 정보와 L1 움직임 정보의 동일성 여부를 다시 한번 판단하여 여전히 동일할 경우 예측 모드를 양예측에서 단예측으로 변경한다.

### IV. 제안하는 방법의 성능 평가

본 논문에서 제안하는 방법의 우수성을 검증하기 위해 HEVC HM 3.0<sup>[2]</sup> 참조 소프트웨어에 실제 구현하여 HM 3.0과의 비교 실험을 수행하였다. 실험 영상 및 조건은 HEVC의 공통실험조건<sup>[4]</sup>을 준수하였다. 제안하는 방법의 계산 복잡도를 측정하기 위해 실험 영상을 인코더로 부호화하여 비트스트림을 생성하는 과정의 평균시간과 비트스트림을 복호화 하는 과정의 평균시간을 측정하였다. 또한 부호화 효율을 측정하기 위해 평균적인 bit-rate 감소량을 나타내는 BD-Bitrate 방법<sup>[5]</sup>을 사용하였다.

표 2. 첫 번째 방법의 실험 결과

Table 2. Experimental result of 1st method

	Low delay B HE			Low delay B LC		
	Y	U	V	Y	U	V
Class A						
Class B	-0.1	0.2	0.0	0.0	-0.1	-0.3
Class C	0.0	0.1	0.0	0.0	-0.1	-0.1
Class D	0.0	0.1	0.3	0.0	-0.3	-0.3
Class E	0.0	0.3	-0.2	0.0	0.0	-0.6
<b>Overall</b>	<b>0.0</b>	<b>0.2</b>	<b>0.0</b>	<b>0.0</b>	<b>-0.1</b>	<b>-0.3</b>
Enc Time[%]	100%			100%		
Dec Time[%]	96%			96%		

표 3. 두 번째 방법의 실험 결과

Table 3. Experimental result of 2nd method

	Low delay B HE			Low delay B LC		
	Y	U	V	Y	U	V
Class A						
Class B	-0.3	-0.5	-0.7	-0.6	-1.0	-1.2
Class C	-0.3	-0.6	-0.6	-0.3	-0.6	-0.6
Class D	-0.1	-0.1	-0.1	-0.2	-0.2	-0.3
Class E	-0.2	-0.7	-0.9	-0.6	-1.1	-1.6
<b>Overall</b>	<b>-0.2</b>	<b>-0.5</b>	<b>-0.5</b>	<b>-0.4</b>	<b>-0.7</b>	<b>-0.9</b>
Enc Time[%]	100%			100%		
Dec Time[%]	94%			95%		

표 4. 세 번째 방법의 실험 결과

Table 4. Experimental result of 3rd method

	Low delay B HE			Low delay B LC		
	Y	U	V	Y	U	V
Class A						
Class B	-0.3	-0.4	-0.4	-0.7	-0.9	-1.0
Class C	-0.3	-0.6	-0.7	-0.4	-0.6	-0.8
Class D	-0.2	-0.3	-0.1	-0.2	-0.7	0.0
Class E	-0.3	-0.9	-1.1	-0.7	-1.8	-1.5
<b>Overall</b>	<b>-0.3</b>	<b>-0.5</b>	<b>-0.5</b>	<b>-0.5</b>	<b>-0.9</b>	<b>-0.8</b>
Enc Time[%]	101%			101%		
Dec Time[%]	95%			98%		

[표 2], [표 3], [표 4]는 각각 첫 번째, 두 번째, 세 번째 해결방법의 부호화/복호화 시간과 BD-Bitrate를 나타낸 것이다. 첫 번째 방법에서는 복호화 시간이 감소하였고, 두 번째 방법에서는 부호화 효율이 개선된 것을 알 수 있다. 첫 번째 방법과 두 번째 방법이 함께 사용된 세 번째 방법에서는 복호

화 시간과 부호화 효율이 모두 개선된 것을 알 수 있다.

LDHE 조건에서는 복호화기의 수행시간을 4%~6% 감소시키고 부호화 성능을 약 0.0% ~ 0.3% 향상시켰으며, LDLC 조건에서는 복호화기의 수행 시간을 2%~5% 감소시켰고 부호화 성능을 약 0% ~0.5% 향상시켰다. 특히 L1 움직임 정보를 주변 블록의 움직임 벡터로 대체하고 여전히 동일한 움직임 정보일 경우 단예측으로 변경하는 세 번째 해결 방법의 경우, LDHE 조건에서는 복호화기의 수행 시간을 5% 감소시키고 부호화 성능을 약 0.3% 향상시켰으며, LDLC 조건에서는 복호화기의 수행 시간을 2% 감소시켰고 부호화 성능을 약 0.5% 향상시켰다.

### V. 결론

HEVC에서 양예측을 수행할 때 L0 움직임 정보와 L1 움직임 정보가 동일해지는 블록이 다수 발생한다. 이는 똑같은 움직임 보상 과정을 불필요하게 두 번 반복함으로써 계산 복잡도의 증가를 야기시킨다. 따라서 본 논문에서는 이러한 문제를 해결하기 하기 위한 방안으로 양예측을 단예측으로 변경하거나 L1 움직임 벡터를 주변의 인접한 블록의 움직임 벡터로 변경하는 방법을 제안하였다. 그 결과 L0 움직임 정보와 L1 움직임 정보의 동일성을 제거할 수 있었으며, HM 3.0대비 복호화기의 수행시간을 2% ~ 5% 감소시키고 부호화 성능을 약 0.3% ~ 0.5% 향상시켰다.

### 참고 문헌

- [1] ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 Part 10 AVC), "Advanced Video Coding for Generic Audiovisual Services," Version 1: March 2003, Version 2: May 2004, Version 3: March 2005, Version 4: September 2005, Version 5 and Version 6: June 2006, Version 7: April 2007, Version 8: July 2007.
- [2] Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "WD3: Working Draft 3 of High-Efficiency Video Coding", JCTVC-E603, Geneva, Switzerland, March 2011.
- [3] Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "HM3: High-Efficiency Video Coding(HEVC) Test Model3 Encoder Description", JCTVC-E603, Geneva, Switzerland, March 2011.
- [4] Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "Common test conditions and software reference configurations", JCTVC-E700, Geneva, Switzerland, March 2011.
- [5] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," ITU-T SG16 Q.6, VCEG-M33, Texas, USA, April 2001.