

특집논문 (Special Paper)

방송공학회논문지 제20권 제1호, 2015년 1월 (JBE Vol. 20, No. 1, January 2015)

<http://dx.doi.org/10.5909/JBE.2015.20.1.57>

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

## HEVC 스크린 콘텐츠 코딩 성능 향상을 위한 화면 내 블록 카피 기술 분석

마 중 현<sup>a)</sup>, 안 용 조<sup>a)</sup>, 심 동 규<sup>a)‡</sup>

### Intra Block Copy Analysis to Improve Coding Efficiency for HEVC Screen Content Coding

Jonghyun Ma<sup>a)</sup>, Yong-Jo Ahn<sup>a)</sup>, and Donggyu Sim<sup>a)‡</sup>

#### 요 약

본 논문에서는 HEVC (high efficiency video coding) SCC (screen content coding)의 화면 내 블록 카피 (IBC: intra block copy) 기술의 소개와 분석을 통해 성능향상 방법에 대해 알아본다. 분석을 위해 HEVC SCC 참조 소프트웨어인 SCM 2를 사용하며, HEVC SCC에 새롭게 채택된 예측 모드인 IBC의 선택 비율과 IBC의 블록 벡터 예측 방법 및 블록 벡터 부호화 방법에 대한 분석결과를 제시한다. 실험결과를 통해 IBC의 선택 비율은 I-Slice와 B-Slice에서 각각 평균 31.08%와 0.33%를 보이고 있으며, 이를 통해 IBC를 선택적으로 수행한다면 부호화 성능을 높일 수 있을 것으로 기대된다. 또한, 블록 벡터 예측 방법과 블록 벡터 부호화 방법에 대한 분석 실험을 통해 기존의 방법이 스크린 콘텐츠 특성이 강한 영상에 대해 효율적이지 못한 것을 확인하였으며 이를 개선하기 위한 방안을 제시한다.

#### Abstract

This paper describes and analyzes IBC (intra block copy) in HEVC (high efficiency video coding) SCC (screen content coding) to improve the coding efficiency of IBC. HEVC SCC reference software SCM 2 is employed to analyze the selection ratio of IBC which is newly adopted in HEVC SCC, and the tools for IBC such as the block vector prediction and block vector coding method are evaluated. Experimental results show the average IBC selection ratio is 31.08% and 0.33% in I-Slice and B-Slice, respectively. Based on this results, the coding efficiency of IBC could be improved by utilizing IBC selectively. In addition, analysis tests of block vector prediction and the block vector coding method show the current methods are not efficient to screen content videos, and the analysis results are presented to improve these methods.

Keyword : HEVC, Screen content coding, SCC, Intra block copy, IBC

---

a) 광운대학교 컴퓨터공학과(Dept. of Computer Engineering, Kwangwoon University)

‡ Corresponding Author : 심동규(Donggyu Sim)

E-mail: dgsim@kw.ac.kr

Tel: +82-2-941-6470

ORCID: <http://orcid.org/0000-0002-2794-9932>

※ 이 논문은 2014년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(NRF-2014R1A2A1A11052210)과 2014년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(NRF-2010-0025325)

· Manuscript received November 18, 2014 Revised January 12, 2015 Accepted January 22, 2015

## 1. 서론

최근 컴퓨터 그래픽 기술의 발달과 디지털 콘텐츠 제작 기술이 향상됨에 따라 방송 및 교육 분야 등에서 디지털 콘텐츠의 제작 및 공급이 활발히 이루어지고 있다. 또한, 다양한 멀티미디어 기기의 보급과 초고속 인터넷 등의 인프라 시설이 구축됨에 따라 디지털 콘텐츠를 이용할 수 있는 방법과 사용자 층이 더욱 확대되었다. 원격 화상회의, 방송콘텐츠, 교육, 애니메이션 등 다양한 분야에 걸쳐 스크린 콘텐츠 영상에 대한 수요가 증가하고 있는 가운데 HEVC (High Efficiency Video Coding)<sup>[1]</sup>를 비롯한 기존의 비디오 압축 표준 기술이 스크린 콘텐츠 영상에 대한 압축을 효율적으로 지원하지 못한다는 문제가 제기되었다. 기존의 비디오 압축 기술은 자연 영상의 압축에 초점을 두어 개발되었기 때문에 그래픽과 텍스트 등으로 이루어진 스크린 콘텐츠 영상의 특성을 제대로 반영하지 못했다.

그림 1은 HEVC SCC (Screen Content Coding) 표준 작업을 위해 사용되고 있는 공통실험조건 (CTC: common test condition)<sup>[2]</sup>의 실험영상의 예시이며, 그래픽과 텍스트 그리고 자연 영상이 혼합된 형태를 나타낸다. 그림 1의 스크린 콘텐츠 영상은 기존의 자연 영상에 비해 색과 형태는 단조롭지만 경계 부근에서 밝기 값의 차이가 두드러지는 특성을

보인다. 이러한 특성을 반영한 효율적인 스크린 콘텐츠 비디오 압축 표준 제정을 위해 2014년 1월 17일 ISO/IEC JTC1/SC29/WG11과 ITU-T SG16 Q6/16에서 CfP (Call for Proposal)<sup>[3]</sup>를 공동으로 발행하였다. 2014년 3월 표준화 회의에서 CfP에 대한 응답으로 일곱 건의 문서가 등록되었으며, 이 중 가장 좋은 성능을 보인 응답<sup>[4]</sup>을 기반으로 표준화가 진행되고 있다. 2014년 11월 현재까지 두 차례의 회의를 거쳐 표준화 작업이 이루어지고 있으며, 2016년 2월 FDIS (Final Draft International Standard)를 목표로 하고 있다.

HEVC SCC는 2014년 7월 14일에 제정된 HEVC RExt 표준 기술<sup>[5]</sup>을 포함하며, 2014년 10월 회의까지 채택된 대표적인 기술로는 화면 내 블록 카피 (IBC: intra block copy)와 팔레트 모드 그리고 적응적 칼라 변환 (ACT: adaptive colour transform)이 있다<sup>[6]</sup>. 이외에도 화면 내 라인 카피 (Intra Line Copy)와 화면 내 문자 카피(Intra String Copy) 등의 연구도 이루어지고 있으며, 2015년 2월 표준화 회의에서 IBC와 팔레트 모드 기술의 향상 실험과 함께 표준화 작업이 진행될 예정이다.

현재 HEVC SCC에 채택된 기술 중 가장 높은 부호화 성능을 보이는 것은 IBC 기술이며, 최대 69.39%의 BD-rate<sup>[7]</sup> 성능 향상을 효과를 보이고 있다<sup>[8]</sup>. 해당 기술은 같은 프레임 내의 복원된 영역에서 부호화 하려는 블록과 가장

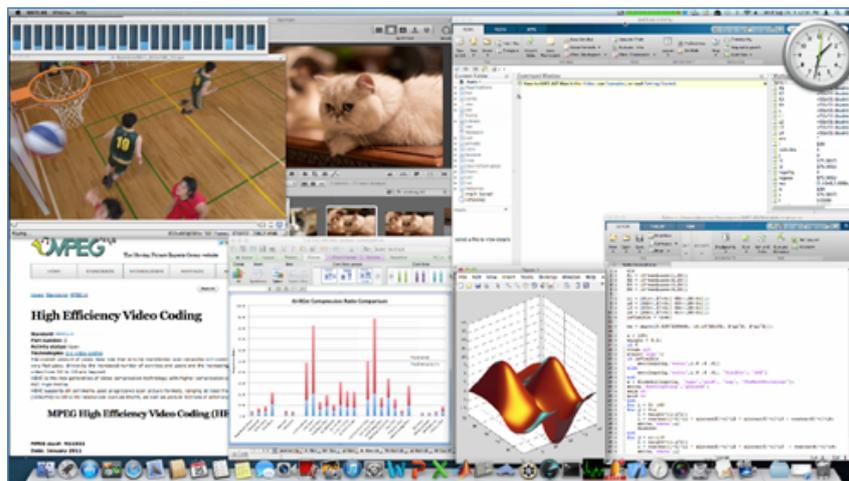


그림 1. HEVC SCC CTC의 'Basketball\_Screen\_2560x1440\_60p\_8b444' 실험 영상  
 Fig. 1. Test sequence 'Basketball\_Screen\_2560x1440\_60p\_8b444' of CTC for HEVC SCC

유사한 블록을 예측하는 기술이다. 특히, IBC로 인한 부호화 효율 향상은 주로 스크린 콘텐츠 특성이 강한 영상이나 I-Slice에서 나타나기 때문에 IBC 예측 모드의 선택 비율에 기반한 적응적인 수행에 대한 연구가 이루어지고 있다<sup>9)</sup>. 또한, IBC의 성능 향상을 위해서는 IBC의 블록 벡터 예측 방법과 블록 벡터 코딩 방법에 대한 분석이 선행되어야 한다.

본 논문에서는 HEVC SCC에 채택된 기술 중 가장 높은 부호화 성능을 보이는 IBC 기술의 소개와 성능 향상을 위한 분석 결과를 제시한다. IBC의 분석을 위해 HEVC SCC 참조 소프트웨어인 SCM 2<sup>10)</sup>를 사용하였으며, SCC의 CTC 하에서 IBC 모드의 선택 비율 분석을 수행하였다. 또한, 분석한 선택 비율을 기반으로 슬라이스 또는 타일별로 IBC 모드의 선택적 수행 방법에 대하여 제안하고 이에 대한 효과에 대해 알아본다. 이와 더불어, IBC 기술 중 블록 벡터 예측 기술의 성능 향상을 위하여 예측 블록 벡터를 구하는 방법에 대한 분석과 각각의 블록 벡터와 예측 블록 벡터 사이의 상관도 분석을 통해 개선 방법을 제안한다. 마지막으로, 블록 벡터와 예측 블록 벡터의 블록 벡터 차분 (BVD: block vector difference)값을 부호화하는 방법에 대해 살펴보고 이를 향상시키기 위해 블록 벡터 차분 값에 대한 분포도 조사를 통해 개선점을 살펴본다. 이와 같은 실험과 분석 결과를 기반으로 제안하는 방법들을 통해 IBC 기술의 부호화 성능이 향상될 것으로 기대된다.

본 논문의 구성은 다음과 같다. II장에서는 HEVC SCC의 IBC 기술에 대한 전반적인 내용 및 세부적인 기술에 대해 알아보고, III장에서는 IBC의 기술에 대한 분석을 통해 부호화 성능을 향상시킬 수 있는 방법에 대해 제시한다. 마지막으로, IV장에서는 앞으로의 연구 방향을 알아보고 본 논문에 대한 결론을 맺는다.

## II. 화면 내 블록 카피 (IBC: intra block copy) 기술

IBC는 현재 부호화 하려는 PU (Prediction Unit)와 동일한 프레임 내의 복원된 영역 중 가장 정합이 잘 되는 부분을 예측하는 기술이다. IBC는 그림 2와 같이 기존의 화면 간 예측, 화면 내 예측, 그리고 이번에 추가된 팔레트 모드와의 경쟁을 통해 CU (Coding Unit) 단위로 결정되는 부호화 모드이다. 또한, IBC는 CU 내에서는 2N×2N, 2N×N, N×2N, N×N의 네 가지 PU 분할모드를 지원한다.

HEVC SCC 인코더의 IBC 기술은 크게 블록 벡터 탐색 (Block vector search), 블록 벡터 예측 (Block vector prediction), 블록 벡터 부호화 (Block vector coding)로 이루어져 있다. 우선, 블록 벡터 탐색에서는 그림 3과 같이 현재 부호화 하는 PU에 대해 복원된 영역 중 예측 성능이 가장

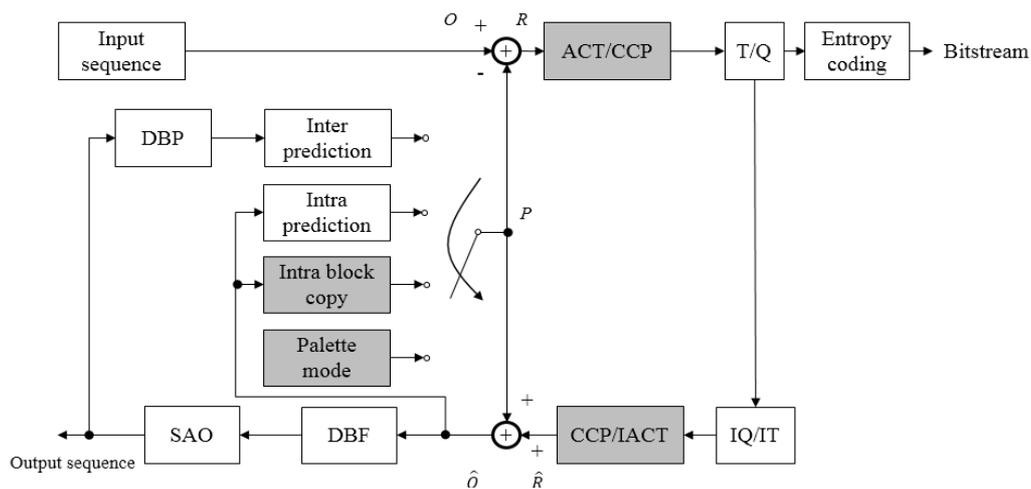


그림 2. HEVC SCC 인코더의 블록도  
 Fig. 2. HEVC SCC encoder diagram

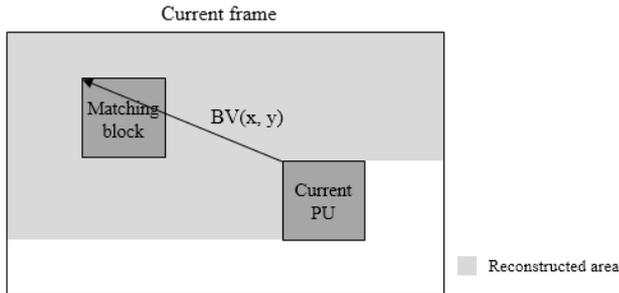


그림 3. IBC 블록 벡터 탐색의 예  
Fig. 3. An example of IBC block vector search

좋은 하나의 블록을 탐색하며, 해당 블록의 위치는 블록 벡터 (BV: block vector)로 기술된다. 이후 블록 벡터 예측을 통해 PU 단위로 전송되는 BV 데이터를 줄이는 과정을 수행한다. 마지막으로 블록 벡터 부호화를 통해 BV와 예측 블록 벡터 (BVP: block vector predictor)의 차분 값 (BVD: block vector difference)을 부호화 하는 과정을 거친다. 또한, IBC의 수행 여부는 SPS (Sequence Parameter Set) RBSP (Raw Byte Sequence Payload)의 'intra\_block\_copy\_enabled\_flag' syntax를 통해 전송할 수 있으며, 해당 syntax가 1인 경우, CU 단위로 'intra\_bc\_flag' syntax를 통해 해당 CU가 IBC로 부호화 되었는지를 전송할 수 있다. 디코더에서는 복호화 하는 비트스트림에서 SPS RBSP의 'intra\_block\_copy\_enabled\_flag' syntax가 1이라면, CU 단위로 'intra\_bc\_flag' syntax 읽어 해당 CU가 IBC로 부호화 되었는지를 확인한다. 해당 syntax가 1이라면, BVD정보를 복호화 한 후, 블록 벡터 예측을 통해 획득한 BVP와의 합산을 통해 BV 정보를 계산한다. 마지막으로, BV에 해당하는 위치의 블록을 사용하여 현재 복호화 하고자 하는 블록을 복원한다.

### 1. 블록 벡터 탐색 (Block vector search)

IBC의 탐색 범위는 기본적으로 같은 프레임 내의 복원된 전체 영역이고 이 영역 내에서 블록을 탐색하는 방법은 비표준화 이슈이다. SCM 2 인코더에서는 인코딩 속도를 고려하여 크게 1D 탐색과 해쉬 탐색 방법을 사용하고 있다.

1D 탐색은 현재 PU에 대해서 수평 방향과 수직 방향으로 순차적인 탐색을 진행하며 휘도성분 블록을 기준으로 네 개의 후보 블록들을 찾고 이 중 색차성분까지 고려하여 하나의 블록을 선택하게 된다. 해쉬 탐색은  $8 \times 8$  CU의  $2N \times 2N$  PU 모드에서 수행되며, IBC로 부호화된 이전 CU 블록들에 대한 BV 정보와 해쉬 값이 기록되어있는 해쉬 테이블에서 현재 PU의 해쉬 값과 동일한 블록들에 대해서만 탐색을 수행한다. 해쉬 테이블의 각 엔트리는 16bits로 이루어져 있으며, 해쉬 값은 원본 영상의 픽셀 값을 이용하여 식 (1)과 같이 계산한다.

식 (1)에서  $DC_0, DC_1, DC_2, DC_3$ 은  $8 \times 8$  블록의 서브블록인  $4 \times 4$  블록 각각에 대한 DC성분들이며,  $MSB(x, n)$ 는  $x$ 에 대한  $n$ 개의 최상위 비트를 의미한다. 마지막으로, Grad는  $8 \times 8$  블록에 대한 gradient 이다. SCM 2 인코더에서는 식 (1)을 이용하여 IBC로 부호화된 블록들에 대한 해쉬 값을 구하여 이를 기준으로 블록들을 분류하고 현재 블록의 해쉬 값과 동일한 블록들에 대해서만 탐색을 수행함으로써 부호화 속도를 개선하고 있다. 이외에도 부호화 속도를 개선하기 위해 CU 크기의 제한, PU 분할 모드의 제한 등의 기법들이 SCM 2 표준 초안 문서<sup>[6]</sup>와 SCM 2 인코더 관련 기고서<sup>[11]</sup>에 기술되어 있다.

### 2. 블록 벡터 예측 (Block vector prediction)

본 절에서는 II-1장의 블록 벡터 탐색을 수행한 후 PU 단위의 참조 블록에 대한 위치 정보인 BV의 정보를 줄이는 블록 벡터 예측에 대하여 살펴본다. IBC의 탐색 범위는 프레임 내의 복원된 전체 영역이며, Full HD 영상 ( $1920 \times 1080$ ) 기준으로 BV가 최대 (1926, -1076)까지 발생할 수 있다. 이러한 BV 정보를 각 PU마다 그대로 전송하는 것은 비효율적이며, 해당 정보를 줄이기 위해 HEVC SCC에서는 두 개의 BVP 후보를 선정하여 이 중 BV와의 차분 값이 적게 발생하는 BVP에 대한 인덱스와 그 차분 값 BVD를 부호화한다. HEVC SCC의 블록 벡터 예측 방법 중 부호화 하려는 PU에 대한 두 개의 BVP 후보선정 과정은 다음과

$$H = MSB(DC_0, 3) \ll 13 + MSB(DC_1, 3) \ll 10 + MSB(DC_2, 3) \ll 7 + MSB(DC_3, 3) \ll 4 + MSB(Grad, 4) \quad (1)$$

같다<sup>[12]</sup>.

i) AMVP (Advanced motion vector predictor) 사용  
 BVP를 선정하는 첫 번째 과정으로, 그림 4와 같이 기존 화면 간 예측의 AMVP 공간적 후보 중  $a_1$ 과  $b_1$  위치의 BV를 사용한다. 해당 위치의 블록이 IBC로 코딩된 경우 해당 BV를 BVP로 사용하며,  $b_1$ 의 경우에는 메모리 낭비를 막기 위해 동일한 CTU 내의 블록인 경우에만 사용한다.

ii) i)에서 두 개의 BVP를 선정하지 못한 경우, 최근 부호화된 IBC BV 두 개를 사용  
 최근 부호화된 IBC BV는 이전에 IBC로 부호화된 블록에서 사용된 BV를 의미한다. 중복되지 않는 값으로 최근의 부호화된 순서로 최대 두 개를 유지하며 CTU (Coding Tree Unit)마다 초기화 된다.

iii) 위의 과정을 통해 채워지지 않은 부분에 대해서는 기본 벡터 값,  $(-2 \times \text{CU\_width}, 0)$ 과  $(-\text{CU\_width}, 0)$ 을 사용

위의 블록 벡터 예측을 통해 PU 단위로 부호화되는 BV의 데이터를 줄일 수 있으며 BV와 BVP의 차분 값인 BVD는 II-3장의 블록 벡터 부호화 방법을 통해 부호화가 이루어진다.

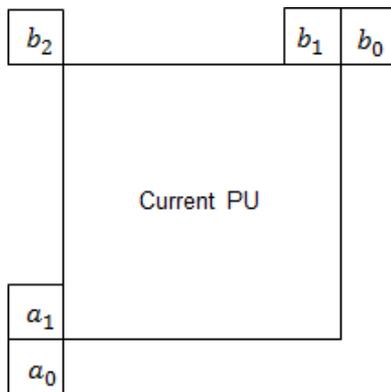


그림 4. 화면 간 예측의 AMVP 공간적 후보  
 Fig. 4. AMVP spatial candidate list for inter prediction

### 3. 블록 벡터 부호화 (Block vector coding)

II-2장을 통해 구한 BVD의 각 원소  $(x, y)$ 는 그림 5와 같은 부호화 과정을 거친다<sup>[13]</sup>. 먼저 BVD의 x원소  $BVD_x$ 가 0이면, 'zero\_flag'를 1로 부호화되고 부호화 과정은 종료된다.  $BVD_x$ 가 0이 아닌 경우에는, 'zero\_flag'를 0으로 부호화되며  $BVD_x$ 의 절대 값에서 1을 뺀 값을 3차 Exponential-Golomb을 통해 부호화한다. 마지막으로,  $BVD_x$  값에 대한 부호 정보를 'sign\_flag'를 통해 부호화하며,  $BVD_y$  값에 대한 부호화 과정도 동일하게 수행된다.

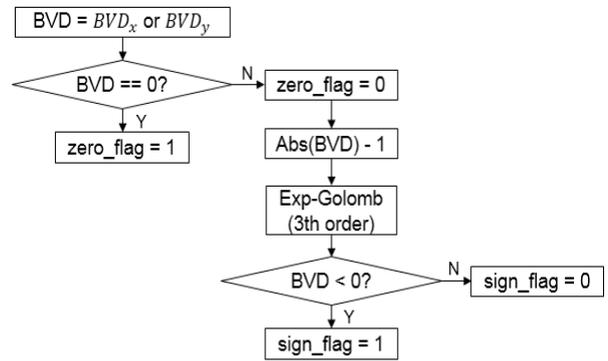


그림 5. IBC의 BVD 부호화 블록도  
 Fig. 5. Block vector coding diagram of IBC

## III. 화면 내 블록 카피 기술의 분석

본 장에서는 화면 내 블록 카피 모드에 대한 기존의 연구 [14]에 기반 하여 화면 내 블록 카피 모드의 선택 비율과 블록 벡터의 예측 성능, 블록 벡터 차분 값의 분포에 대한 분석을 수행한다. 또한, 전술한 화면 내 블록 카피의 대표적 세 가지 기술들에 대한 부호화 효율 향상 방법에 대하여 제시한다.

### 1. 화면 내 블록 카피 모드의 선택 비율

HEVC version 1에서는 화면 내 예측 모드와 화면 간 예측 모드의 두 가지 예측 모드가 존재했던 것에 비하여

HEVC SCC에는 IBC 모드와 팔레트 모드가 추가되었다. IBC 모드는 영상 내에서 단순한 모양이나 텍스처 반복 등의 스크린 콘텐츠 특성이 강한 부분에 대하여 예측성능을 높이기 위한 모드이다<sup>8)</sup>. 스크린 콘텐츠 코딩 성능 향상 연구를 위해서는 기존의 화면 간 예측 및 화면 내 예측과 더불어 화면 내 블록 카피 모드의 선택 비율의 분석이 선행되어야 한다.

본 논문에서는 새롭게 추가된 모드 중 화면 내 블록 카피 모드의 선택 비율을 알아보기 위해 HEVC SCC의 참조 소

프트웨어인 SCM 2를 이용하여 CTC 하에 실험을 진행하였다. 표 1~3은 각각 AI (All intra), RA (Random access), LB (Low-delay B) 실험 조건에서의 CU 모드 선택 비율을 나타낸다. 표에서 Resolution은 해당 영상의 해상도, Sequence는 영상의 이름, Category는 영상의 특성, Intra는 화면 내 예측 모드의 선택 비율, IBC는 IBC 모드의 선택 비율, 그리고 Inter는 화면 간 예측 모드의 선택 비율을 의미한다. 추가적으로 팔레트 모드의 경우에는 syntax 상의 분류와 동일하게 화면 내 예측 모드 비율에 포함하였다. Category에서 TGM (Text and graphics with motion)은 그래픽과 텍스트가 혼합된 영상으로 스크린 콘텐츠의 특성을 가장 잘 나타내고 있는 영상이며, M (Mixed content)은 스크린 콘텐츠 영상의 특성과 자연 영상의 특성이 혼합된 영상이다. 또한, A (Animation)은 애니메이션 영상이며 CC (Camera-captured content)는 자연 영상을 의미한다. 표 1~3에서 CU 모드의 선택 비율은 각각의 모드로 선택된 CU 모드의 수에 대한 비율이 아닌 CU 크기에 대한 면적 비율을 의미한다.

표 1에서는 TGM 카테고리에서 IBC 모드가 상대적으로 많이 선택되는 것을 볼 수 있으며 특히 해상도가 큰 영상에 대해서 IBC 모드 선택의 비율이 높은 것을 알 수 있다. 표 2와 표 3의 I-Slice 에서도 표 1과 비슷한 경향을 보이고 있다. 하지만, 표 2와 표 3의 B-Slice에서는 'sc\_flyingGraphics' 영상을 제외하고는 IBC 모드의 선택 비율이 평균 0.33%로

표 1. CU 모드 선택 비율 (All intra configuration)  
Table 1. CU mode selection ratio (All intra configuration)

Resolution	Sequence	Category	Intra	IBC
1920x1080	sc_flyingGraphics	TGM	33.90%	66.10%
	sc_desktop	TGM	38.41%	61.59%
	sc_console	TGM	33.41%	66.59%
	MissionControlClip3	M	72.13%	27.87%
	EBURainFruits	CC	98.86%	1.14%
	Kimono1	CC	99.78%	0.22%
1280x720	sc_web_browsing	TGM	52.81%	47.19%
	sc_map	TGM	87.32%	12.68%
	sc_programming	TGM	62.48%	37.52%
	sc_SlideShow	TGM	62.48%	37.52%
	sc_robot	A	95.81%	4.19%
2560x1440	Basketball_Screen	M	69.43%	30.57%
	MissionControlClip2	M	79.68%	20.32%

표 2. CU 모드 선택 비율 (Random access configuration)  
Table 2. CU mode selection ratio (Random access configuration)

Resolution	Sequence	Category	I-Slice		B-Slice		
			Intra	IBC	Inter	Intra	IBC
1920x1080	sc_flyingGraphics	TGM	34.45%	65.55%	83.41%	6.03%	10.56%
	sc_desktop	TGM	38.80%	61.20%	99.15%	0.31%	0.54%
	sc_console	TGM	33.93%	66.07%	97.56%	1.20%	1.23%
	MissionControlClip3	M	70.53%	29.47%	99.46%	0.36%	0.18%
	EBURainFruits	CC	98.82%	1.18%	99.54%	0.45%	0.01%
	Kimono1	CC	99.77%	0.23%	90.77%	9.21%	0.02%
1280x720	sc_web_browsing	TGM	52.79%	47.21%	99.57%	0.30%	0.13%
	sc_map	TGM	87.63%	12.37%	98.76%	0.74%	0.50%
	sc_programming	TGM	60.93%	39.07%	97.02%	2.42%	0.55%
	sc_SlideShow	TGM	94.30%	5.70%	97.03%	2.80%	0.16%
	sc_robot	A	95.33%	4.67%	99.04%	0.92%	0.04%
2560x1440	Basketball_Screen	M	66.20%	33.80%	99.47%	0.51%	0.02%
	MissionControlClip2	M	77.15%	22.85%	98.05%	1.37%	0.57%

표 3 CU 모드 선택 비율 (Low-delay B configuration)  
 Table 3 CU mode selection ratio (Low-delay B configuration)

Resolution	Sequence	Category	I-Slice		B-Slice		
			Intra	IBC	Inter	Intra	IBC
1920x1080	sc_flyingGraphics	TGM	31.84%	68.16%	81.15%	8.65%	10.20%
	sc_desktop	TGM	40.08%	59.92%	99.13%	0.31%	0.55%
	sc_console	TGM	36.96%	63.04%	97.64%	1.12%	1.23%
	MissionControlClip3	M	65.82%	34.18%	99.54%	0.28%	0.18%
	EBURainFruits	CC	98.85%	1.15%	99.84%	0.15%	0.00%
	Kimono1	CC	99.83%	0.17%	95.79%	4.20%	0.02%
1280x720	sc_web_browsing	TGM	63.55%	36.45%	99.17%	0.58%	0.25%
	sc_map	TGM	85.36%	14.64%	99.31%	0.52%	0.17%
	sc_programming	TGM	53.89%	46.11%	96.76%	2.61%	0.62%
	sc_SlideShow	TGM	82.97%	17.03%	95.45%	4.28%	0.28%
	sc_robot	A	95.47%	4.53%	98.90%	1.05%	0.04%
2560x1440	Basketball_Screen	M	69.30%	30.70%	99.58%	0.41%	0.01%
	MissionControlClip2	M	66.86%	33.14%	98.12%	1.30%	0.58%

거의 선택되지 않는 것을 볼 수 있다.

IBC 모드의 선택비율이 위와 같이 나타나는 이유는 다음과 같이 분석해볼 수 있다. 우선, I-Slice에서는 IBC 모드와의 경쟁 모드로 화면 내 예측 모드만 존재한다. 화면 내 예측 모드에서는 현재 부호화 하려는 블록의 주변 샘플들에 필터링을 취하여 변환과 양자화로 인한 열화현상을 줄인 값을 참조한다. 이러한 이유로 일반적인 자연 영상에서는 부호화 효율이 증가하지만, 예지 정보가 많고 반복되는 패턴들이 산발적으로 일어나는 TGM 카테고리의 영상에서는 필터링된 화면 내 참조 샘플이 오히려 예측 에러를 확산시킬 수 있다. 따라서 스크린 콘텐츠 특성이 강한 TGM 카테고리 영상들에 대해서는 블록 대 블록 카피를 수행하는 IBC 모드의 선택 비율이 높게 나타난다. 또한, 참조 가능 범위가 화면 내 예측의 경우 공간적으로 인접한 샘플들인데 비해 IBC는 복원된 영역 전체이기 때문에 반복되는 패턴들에 대한 정밀한 예측이 가능하다. 다음으로 B-Slice에서는 IBC 모드의 경쟁 모드로 화면 내 예측뿐만 아니라 화면 간 예측 모드가 추가적으로 존재한다. 화면 간 예측 모드에서는 시간적 유사성에 기반하여 복원된 여러 장의 픽처를 참조할 수 있으므로 IBC 모드에 비해 정확한 예측이 가능하다.

본 논문에서는 이러한 CU 모드 선택 비율에 기반하여 슬라이스에 따라 선택적인 IBC 모드 수행을 제안한다. 제안하는 방법을 적용할 경우, 싱글 슬라이스 구조에서는 일

정시간 동안 스크린 콘텐츠 영상과 자연 영상이 flipping되는 영상에서 부호화 효율을 얻을 수 있으며, 멀티 슬라이스 구조에서는 한 화면 내에 스크린 콘텐츠 영상과 자연 영상이 혼합된 경우에도 부호화 효율을 얻을 수 있다. 제안하는 방법의 추가적인 기대효과는 자연 영상의 특성이 강한 부분이나 B-Slice에서 IBC 모드로 부호화를 시도하지 않음으로써 인코더의 속도 증가부분과, CU 단위로 전송하는 'intra\_bc\_flag' syntax를 줄임으로써 발생하는 비트량 감소 등이 있다<sup>[11]</sup>. 또한, 슬라이스 단위뿐 아니라, 타일로 분할하고 분할 구조에 따라 IBC 모드를 선택적으로 수행한다면, 다양한 형태의 영상에 대해서도 효율적인 부호화가 가능할 것으로 기대된다.

제안하는 방법의 성능을 평가하기 위하여 RA 코딩 환경에서 B-Slice내의 IBC 모드 수행 및 관련 syntax에 대한 부호화를 수행하지 않는 경우의 부호화 효율 및 속도를 측정하였다. IBC 모드의 선택 비율이 상대적으로 높은 TGM 카테고리의 영상들에 대해서는 BD-rate의 손실이 평균 4.88%이었지만, IBC 모드의 선택 비율이 낮은 CC 카테고리의 영상들에 대해서는 BD-rate 손실이 평균 0.16%정도였으며 부호화 속도는 평균 9.22% 개선됨을 볼 수 있었다.

## 2. 블록 벡터 예측 방법

II-2장에서 IBC의 블록 벡터 예측 방법에서 BVP를 구하

는 과정은 세 단계로 구성되어 있으며, 마지막 단계에서는 기본 벡터 값을 사용하는 것을 확인하였다. 일반적으로 기본 벡터 값을 사용하는 단계가 가장 마지막인 이유는 인접한 블록의 BV를 사용하는 것에 비해 예측 성능이 떨어지기 때문이며, 이는 현재 블록의 BV와 인접한 블록의 BV간의 유사성이 기본 벡터 값과의 유사성보다 높다는 것을 의미한다.

본 논문에서는 현재 IBC의 블록 벡터 예측 방법에서 기본 벡터 값의 선택 비율을 알아보기 위하여 IBC 모드 선택 비율이 가장 높게 나타나는 AI 코딩 환경에서 실험을 진행하였다.

표 4는 IBC 블록 벡터 예측을 통한 BVP 구성 상태 및 향상 가능성을 나타낸다. 표 4에서 Empty space는 II-2장에서 BVP를 구하는 과정 중 두 번째 단계 수행 이후 BVP 두 개 중 하나라도 채워지지 않은 경우에 대한 비율을 나타낸다. 최소 10.92%에서 최대 61.24%까지 BVP가 채워지지 않는 것으로 나타났다. 이 경우 기본 벡터 값을 사용하게 되며, 본 논문에서는 기본 벡터를 사용하기 전 주변 블록에서 BVP로 사용할 수 있는 BV가 존재하는지에 대한 실험을 진행하였다. 표 4의 Availability는 Empty space 상황에서 주변 블록에서 BVP로 사용할 수 있는 BV가 존재하는 경우에 대한 비율로, 본 실험에서 주변 블록의 검사는 그림 4와 같이 화면 간 예측의 AMVP 후보 중 IBC를 위한 블록 벡터

예측에서 검사되었던  $a_1$  과  $b_1$  을 제외한  $b_0$ ,  $a_0$ ,  $b_2$  를 대상으로 수행하였다. 또한, 메모리 효율을 위해 CTU를 넘어가는  $b_1$  에 대해서는 후보로 사용하지 않았던 것과 동일하게  $b_0$ ,  $b_2$  의 경우에 CTU를 넘어가는 블록에 대해서는 이용할 수 있는 BV에서 제외하였다.

표 4의 Availability는 최소 0.88%에서 최대 23.30%까지 나타났으며, Availability가 낮은 영상들은 CC 카테고리에 해당하는 자연 영상들로써 표 1~3에서 보았던 것처럼 IBC 모드의 선택 비율이 낮은 영상들이다. 반면, TGM 카테고리에 해당하는 스크린 콘텐츠 특성이 강한 영상에 대해서는 Availability가 높게 나타나며, 이는 IBC 모드의 선택 비율이 높은 TGM 카테고리의 영상들에서는 현재 블록 주변에 IBC로 부호화된 블록이 존재할 확률이 상대적으로 높기 때문이다. 이러한 이유에 기반하여 두 가지 위치 ( $a_1$  과  $b_1$ ) 에 대해서만 검사하던 기존의 방법에 비해  $b_0$ ,  $a_0$  및  $b_2$  의 위치에 대해서도 추가적인 검사를 수행하므로 Availability가 높게 나타난다. 또한, TGM 카테고리의 영상에서 Availability가 높다는 것은 기존의 IBC의 블록 벡터 예측 방법이 주변 블록의 BV를 효과적으로 이용하지 못하고 있음을 반증한다.

본 논문에서는 위의 분석 결과를 기반으로 기본 벡터를 사용하기 전 BVP 후보에 대한 검사를  $b_0$ ,  $a_0$  및  $b_2$  의 위치에 대하여 추가로 실시하였다. 그 결과 TGM 카테고리의 영상들에 대해 부호화 속도 저하 없이 평균 0.1%의 BD-rate 성능 향상이 있음을 확인하였으며, 주변 블록의 탐색 범위를 늘리거나 BVP 후보를 고르는 순서를 조절하는 방법 등을 통해 추가적인 부호화 성능 향상이 있을 것으로 기대된다.

### 3. 블록 벡터 차분 값 부호화 방법

HEVC SCC 인코더에서는 IBC의 블록 벡터 탐색 및 블록 벡터 예측을 수행한 후 디코더로 BV와 BVP의 차인 BVD정보를 디코더로 전송한다. 이때, PU 단위로 전송하는 BVD의 데이터를 줄이기 위해서는 블록 벡터 예측 방법도 중요하지만, BVD를 부호화 하는 방법 역시 중요하다. 특

표 4. IBC 블록 벡터 예측을 통한 BVP 구성 상태 및 향상 가능성  
Table 4. BVP construction state after employing block vector prediction and the improvement possibility

Resolution	Sequence	Category	Empty space	Availability
1920x1080	sc_flyingGraphics	TGM	13.72%	14.46%
	sc_desktop	TGM	10.92%	22.60%
	sc_console	TGM	14.13%	21.69%
	MissionControlClip3	M	11.16%	16.02%
	EBURainFruits	CC	32.80%	1.32%
	Kimono1	CC	61.24%	0.88%
1280x720	sc_web_browsing	TGM	11.05%	23.30%
	sc_map	TGM	11.32%	8.58%
	sc_programming	TGM	12.70%	14.53%
	sc_SlideShow	TGM	21.10%	7.53%
	sc_robot	A	22.09%	3.18%
2560x1440	Basketball_Screen	M	14.09%	14.43%
	MissionControlClip2	M	18.70%	9.93%

히, IBC는 화면 내 예측 모드와의 율-왜곡 (RD: Rate-Distortion) 경쟁에서 원본 영상과의 왜곡 (Distortion)이 화면 내 예측 모드보다 더 적은 경우에도 추가적으로 부호화하는 BVD 관련 syntax 정보로 인하여 IBC 모드의 선택 기회를 잃는 경우가 발생한다. 따라서 IBC의 성능 개선을 위해서는 BVD를 효율적으로 감소시키기 위한 연구가 필요하며, 본 논문에서는 해당 연구에 앞서 기존의 블록 벡터 예측 방법을 통한 BV와 BVP간의 상관도 실험 및 BVD 분포에 대한 실험을 진행하였다.

우선, AI 코딩 환경에서 IBC 모드가 가장 많이 선택되는 'sc\_console' 영상을 이용하여 BV와 BVP간의 상관도를 측정하였다. 표 5에서 Correlation\_x와 Correlation\_y는 BV의 원소 (x, y)에 대하여 BVP의 원소 (x', y')와의 상관관계를 각각 구한 값으로 y에 대한 상관계수가 x에 대한 상관계수에 비하여 상대적으로 높은 것을 확인할 수 있다. 이는 영상

의 크기가 세로축 방향으로 더 짧기 때문에 블록 벡터 예측이 실패했을 경우 발생할 수 있는 BVD의 값이 가로축 방향에 비해 상대적으로 더 작기 때문이다.

다음으로 BVD의 분포를 알아보기 위한 실험을 진행하였다. 실험 결과 BVD(BVD<sub>x</sub>, BVD<sub>y</sub>)에 대하여 전체 BVD<sub>x</sub> 중 0이 차지하는 비율은 68.07%로 나타났으며, 전체 BVD<sub>y</sub> 중 0이 차지하는 비율은 82.57%로 나타났다. 실험 결과에서 나타나는 바와 같이 BVD<sub>y</sub>에서 0이 차지하는 비율이 높은 것은 y축 방향으로 블록 벡터 예측 방법이 더 정확하다는 것을 의미하며, 이는 표 5의 BV와 BVP간의 상관관계 실험 결과 및 분석 내용을 통해 알 수 있다.

또한, 기존의 블록 벡터 차분값 부호화 방법인 3차 Exp-Golomb 부호화 방법이 적합한지에 대한 분석을 위해 위 실험에서 0을 제외한 나머지 BVD들에 대한 비율을 조사하였다. 그림 6의 가로 축은 BVD<sub>x</sub> 또는 BVD<sub>y</sub> 값을 의미

표 5. BV와 BVP의 상관관계  
 Table 5. The correlation between BV and BVP

Resolution	Sequence	Category	QP	Correlation_x	Correlation_y
1920x1080	sc_console	TGM	22	77.90%	84.90%
			27	78.08%	85.76%
			32	78.73%	86.05%
			37	81.01%	86.72%
Average				78.93%	85.86%

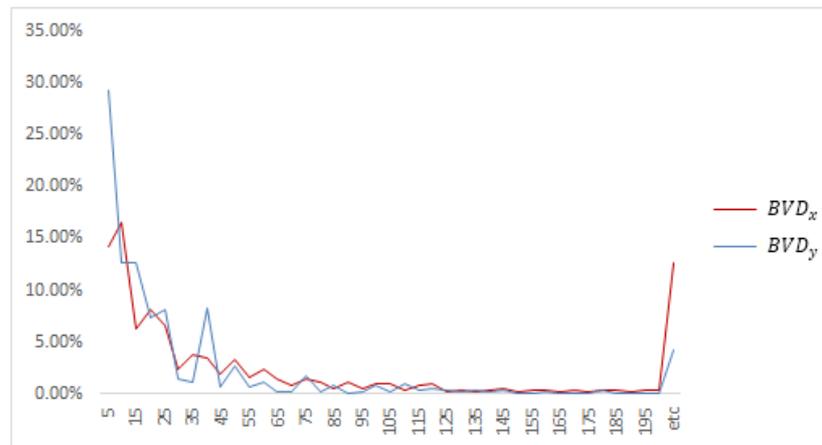


그림 6. BVD<sub>x</sub>와 BVD<sub>y</sub>의 분포  
 Fig. 6. The distribution of BVD<sub>x</sub> and BVD<sub>y</sub>

하며 etc는 블록 벡터 차분 값이 200을 초과하는 값을 의미한다. 세로축은 해당 BVD 값이 차지하는 비율을 나타내며 빨간 선과 파란 선은 각각  $BVD_x$ 와  $BVD_y$ 의 분포를 보여주고 있다.  $BVD_y$ 의 분포가  $BVD_x$ 의 분포에 비해 상대적으로 BVD가 0에 가까운 곳에 많이 나타나는 것을 확인할 수 있다.

위의 실험결과를 바탕으로  $BVD_x$ 와  $BVD_y$ 의 분포가 상이함을 확인하였으며, 이러한 결과에 기반하여 본 논문에서는 기존의  $BVD_x$ 와  $BVD_y$ 의 부호화 성능 향상을 위해서 서로 다른 부호화 방법을 적용하여 부호화 효율 향상을 측정하였다. 본 실험을 위하여 블록 벡터 부호화 방법은  $BVD_x$ 와  $BVD_y$ 에 같은 방식의 블록 벡터 부호화 방법을 적용하는 대신, BVD의 값이 작은 쪽에 분포율이 높은  $BVD_y$ 에는 차수가 낮은 Exp-Golomb 부호화 방법을 적용하고 BVD의 값이 큰 쪽에 분포율이 높은  $BVD_x$ 에는 차수가 높은 Exp-Golomb 부호화 방법을 적용하였다. 이에 대한 구현의 예로  $BVD_x$ 의 부호화를 위하여 4차 Exp-Golomb 부호화 방법을 적용시키면  $BVD_x$ 에 대한 비트량이 평균 1.00% 감소하였다. 또한,  $BVD_x$ 가 4보다 작은 경우는 3차 Exp-Golomb 부호화 방식을 사용하고 4보다 큰 경우는 4차 Exp-Golomb 부호화 방식을 사용하면 비트량이 평균 2.53% 감소하는 것을 확인하였다. 이는 작은 수를 부호화할 때는 차수가 낮은 것이 효율적이고 큰 수를 부호화할 때는 차수가 높은 것이 효율적인 Exp-Golomb 부호화 성질에 의해 나타난 것이다. 이에 기반하여  $BVD_x$ 와  $BVD_y$ 의 분포율에 따라 부호화 방식을 적용적으로 차등하게 적용시킨다면 추가적인 비트량 감소를 기대할 수 있다.

#### IV. 결론

본 논문에서는 현재 표준화 작업이 진행 중인 HEVC SCC 기술 중 IBC의 세부적인 기술에 대한 소개와 분석 및 기술 개선 방법에 대해 다루고 있다. 스크린 콘텐츠 영상의 특성이 강한 TGM 클래스에서는 IBC 모드의 선택비율이 높았던 반면, 자연 영상이나 애니메이션 영상에 대해서는 IBC 모드의 선택 비율이 낮게 나타났다. 특히, B-Slice에서

는 'sc\_flyingGraphics' 영상을 제외한 모든 영상에서 IBC 모드의 선택 비율이 평균 0.33%에 불과한 것을 확인할 수 있었다. 이 분석 결과를 통해 영상의 종류나 슬라이스, 타일 별로 IBC의 수행 여부를 선택한다면, 부호화 효율이나 속도 면에서 성능 향상을 보일 수 있을 것으로 기대된다. 또한, 블록 벡터 예측 방법에 대한 분석을 통해 IBC 모드의 선택비율이 높은 TGM 클래스 영상에 대해서는 현재 적용되어 있는 블록 벡터 예측 방법이 효율적이지 못한 것을 확인할 수 있었다. 블록 벡터 예측 방법 수행 시 주변 블록 탐색 범위를 확대하면 기본 벡터 대신 주변의 BV를 BVP로 사용할 수 있는 확률이 최대 23.30%까지 증가할 수 있는 것을 확인하였다. 마지막으로, BV와 BVP간의 상관관계 분석과 BVD의 분포도 조사를 통해  $BVD_x$ 와  $BVD_y$ 의 특성이 다른 것을 확인하였다. 이를 개선하기 위하여 현재  $BVD_x$ 와  $BVD_y$ 에 동일한 방식의 블록 벡터 부호화방법을 사용하는 대신 서로 다른 알고리즘을 적용할 경우 비트량을 감소시킬 수 있을 것으로 기대된다.

향후에는 IBC 모드의 부호화 속도 개선을 위해 본 논문에서 분석한 결과를 기반으로 영상의 특성에 따라 IBC 모드를 적용적으로 수행하는 알고리즘에 대한 연구가 필요하다. 또한, IBC 모드의 부호화 효율 개선을 위해 블록 벡터 예측 기술에 대한 향상 방안과 블록 벡터 차분값을 부호화 방법에 대한 추가적인 연구도 필요할 것이다.

#### 참고 문헌 (References)

- [1] High Efficiency Video Coding, Rec. ITU-T H.265 and ISO/IEC 23008-2, Jan. 2013.
- [2] H. Yu, et al., "Common conditions for screen content coding tests," JCTVC-R1015, 18th JCT-VC meeting, Sapporo, JP, June 2014.
- [3] 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.
- [4] J. Chen, et al., "Description of screen content coding technology proposal by Qualcomm," JCTVC-Q0031, 17th JCT-VC meeting, Valencia, ES, March 2014.
- [5] J. Boyce, et al., "Draft high efficiency video coding (HEVC) version 2, combined format range extensions (RExt), scalability (SHVC), and multi-view (MV-HEVC) extensions," JCTVC-R1013, 18th JCT-VC meeting, Sapporo, JP, June 2014.
- [6] R. Joshi, and J. Xu, "High Efficiency Video Coding (HEVC) Screen

Content Coding: Draft 1," JCTVC-R1005, 18th JCT-VC meeting, Sapporo, JP, June 2014.

[7] G. Bjontegaard, "Calculation of Average PSNR Differences Between RD-curves," VCEG-M33, ITU-T Video Coding Experts Group (VCEG) Meeting, Austin, TX, April 2001.

[8] C. Pang, et al., "SCCE1: Test 1.1 - Intra block copy with different search areas," JCTVC-R0184, 18th JCT-VC meeting, Sapporo, JP, June 2014.

[9] W. Lim, et al., "Slice-level Intra block copy enabling," JCTVC-S0056, 19th JCT-VC meeting, Strasbourg, FR, Oct. 2014.

[10] HM-15.0\_REExt-8.0\_SCM-2.0 (SCM-2.0), [https://hevc.hhi.fraunhofer.de/svn/svn\\_HEVCSoftware/tags/HM-15.0+REExt-8.0+SCM-2.0](https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-15.0+REExt-8.0+SCM-2.0)

[11] R. Joshi, et al., "Screen content coding test model 2 (SCM 2)," JCTVC-R1014, 18th JCT-VC meeting, Sapporo, JP, June 2014.

[12] C. Pang, et al., "Non-SCCE1: Combination of JCTVC-R0185 and JCTVC-R0203," JCTVC-R0309, 18th JCT-VC meeting, Sapporo, JP, June 2014.

[13] C. Pang, et al., "SCCE1: Test 3.6 - Block vector coding for Intra block copy," JCTVC-R0186, 18th JCT-VC meeting, Sapporo, JP, June 2014.

[14] J. Ma, et al., "Analysis of Intra block copy technology to improve screen content coding", 2014 Korean Society of Broadcast Engineers Fall Conference, Nov. 2014.

---

저 자 소 개

---



마 중 현

- 2013년 8월 : 광운대학교 컴퓨터공학과 학사
- 2013년 9월 ~ 현재 : 광운대학교 컴퓨터공학과 석사과정
- ORCID : <http://orcid.org/0000-0002-5347-1223>
- 주관심분야 : 영상압축, 컴퓨터비전



안 용 조

- 2010년 2월 : 광운대학교 컴퓨터공학과 학사
- 2012년 2월 : 광운대학교 컴퓨터공학과 석사
- 2012년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 박사과정
- ORCID : <https://orcid.org/0000-0002-0012-0905>
- 주관심분야 : 영상압축, 최적화 및 병렬화



심 동 규

- 1993년 2월 : 서강대학교 전자공학과 공학사
- 1995년 2월 : 서강대학교 전자공학과 공학석사
- 1999년 2월 : 서강대학교 전자공학과 공학박사
- 1999년 3월 ~ 2000년 8월 : 현대전자 선임연구원
- 2000년 9월 ~ 2002년 3월 : 바로비전 선임연구원
- 2002년 4월 ~ 2005년 2월 : University of Washington Senior research engineer
- 2005년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 교수
- ORCID : <http://orcid.org/0000-0002-2794-9932>
- 주관심분야 : 영상신호처리, 영상압축, 컴퓨터비전