

정규논문 (Regular Paper)

방송공학회논문지 제19권 제5호, 2014년 9월 (JBE Vol. 19, No. 5, September 2014)

<http://dx.doi.org/10.5909/JBE.2014.19.5.640>

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

## 깊이정보를 이용한 HEVC의 인코더 고속화 방법

이 윤 진<sup>a)</sup>, 배 동 인<sup>a)</sup>, 박 광 훈<sup>a)†</sup>

## HEVC Encoder Optimization using Depth Information

Yoon Jin Lee<sup>a)</sup>, Dong In Bae<sup>a)</sup>, and Gwang Hoon Park<sup>a)†</sup>

### 요 약

최근 영상시스템 환경은 2D 비디오카메라에 깊이 카메라가 부착되어 2D 및 3D 어플리케이션을 지원하는 형태로 보편화 되고 있다. 이러한 3차원 멀티미디어 시스템 환경으로의 변화는 비디오 시스템에서 깊이정보 획득을 용이하게 만들었다. 깊이정보는 객체 구분, 배경영역 인지 등에 이용할 수 있는데, 2D 부호화에 이를 이용한다면 높은 부호화 효율을 얻을 수 있다. 따라서, 본 논문에서는 차세대 2D 비디오 코덱인 HEVC 인코더에 반영한 깊이정보 이용 비디오 부호화 방법을 제안한다. 제안방법으로, 현재 부호화하려는 CU가 배경영역에 위치할 경우 1) 주변블록의 SKIP 모드를 참조하여 결정하는 CU 분할 조기 결정, 2) 시간적 위치의 CU 정보를 이용하여 수행하는 CU 분할 구조 제한, 3) 배경영역에 따른 움직임 예측 탐색 범위 제한이 있다. 실험은 HEVC 참조 소프트웨어인 HM 12.0에 적용하였고, 실험결과 40% 이상의 부호화 복잡도가 감소했으며, BD-Bitrate는 0.5% 손실되었다. 특히, 마이크로소프트사에서 개발한 키넥트를 통해 획득한 영상을 이용한 실험 결과에서는 영상 품질의 큰 열화 없이 기존대비 최대 53%의 부호화 복잡도가 감소하는 결과를 나타내어, 향후 실시간 화상통신, 모바일 또는 핸드헬드 환경에서의 비디오 서비스 등에서 광범위하게 적용할 수 있을 것으로 기대된다.

### Abstract

Many of today's video systems have additional depth camera to provide extra features such as 3D support. Thanks to these changes made in multimedia system, it is now much easier to obtain depth information of the video. Depth information can be used in various areas such as object classification, background area recognition, and so on. With depth information, we can achieve even higher coding efficiency compared to only using conventional method. Thus, in this paper, we propose the 2D video coding algorithm which uses depth information on top of the next generation 2D video codec HEVC. Background area can be recognized with depth information and by performing HEVC with it, coding complexity can be reduced. If current CU is background area, we propose the following three methods, 1) Earlier stop split structure of CU with PU SKIP mode, 2) Limiting split structure of CU with CU information in temporal position, 3) Limiting the range of motion searching. We implement our proposal using HEVC HM 12.0 reference software. With these methods results shows that encoding complexity is reduced more than 40% with only 0.5% BD-Bitrate loss. Especially, in case of video acquired through the Kinect developed by Microsoft Corp., encoding complexity is reduced by max 53% without a loss of quality. So, it is expected that these techniques can apply real-time online communication, mobile or handheld video service and so on.

Keyword : HEVC(High Efficiency Video Coding), Kinect, Depth Information, Encoder Optimization

## I. 서 론

최근 영상시스템은 일반 카메라와 깊이 카메라 함께 결합된 형태의 시스템으로 개발되고 있다. 이러한 개발 추세는 기존의 2D 멀티미디어 시스템 환경에서 3D 멀티미디어 시스템 환경으로 변화를 나타내고 있으며, 대표적인 제품으로는 마이크로소프트사가 개발한 동작 및 인식 장치인 키넥트센서<sup>[1]</sup>, 다양한 3D 기능을 제공하는 구글 탱고 스마트폰<sup>[2]</sup>, 실시간으로 피사체와 배경을 분리하는 인텔 리얼센스 3D 카메라<sup>[3]</sup> 등이 있다.

비디오 시스템 환경의 변화는 카메라를 이용한 촬영, 편집, 저장의 2차원 어플리케이션에서 깊이정보를 이용하여 동작 인식, 객체 추출, 3차원 스캔<sup>[4][8]</sup> 등의 3차원 어플리케이션으로의 확장을 지원하고 있어, 비디오 시스템 환경에서의 깊이정보를 이용한 어플리케이션의 개발은 보편화 되고 있다. 또한, 최근의 영상서비스는 HD 디지털 방송 서비스가 대중화 되었고, 초고해상, 초고화질의 UHD 서비스 환경으로 변화가 시작되고 있으며, 디스플레이 환경에서는 3D 텔레비전, 3D 스마트폰, 3D 게임기 등이 상용화되어 고해상도 3D 디스플레이 환경으로 보편화 되어가고 있다.

현재 동영상 표준 단체인 Moving Picture Experts Group (MPEG)<sup>[9]</sup>과 ITU-T 산하의 Video Coding Experts Group (VCEG)<sup>[10]</sup>은 공동으로 Joint Collaborative Team on Video Coding(JCT-VC)<sup>[11]</sup>와 The Joint Collaborative Team on 3D Video Coding Extension Development(JCT-3V)<sup>[12]</sup>를 결성하여 이러한 시대적 변화에 대응하여 차세대 비디오 압축 표준을 결정하고 있으며, 2D 비디오 코덱에서는 2013년 High Efficiency Video Coding(HEVC)<sup>[13]</sup> 표준화를 완료하였고, 3D 비디오 코덱에서는 2013년 H.264/AVC<sup>[14]</sup> 기

반 3차원 비디오 부호화 표준(3D-AVC<sup>[15]</sup>)을 완료하였고, 2015년 HEVC 기반 3차원 비디오 부호화 표준(3D-HEVC<sup>[16]</sup>) 완료를 목표로 진행 중에 있어, HD, UHD 영상 뿐 아니라 3D 방송 및 이동통신망에서도 현재보다 낮은 전송 대역폭으로 고화질 및 실감형 영상을 제공할 것으로 기대된다.

깊이정보는 카메라에서 사물간의 거리를 나타내는 정보로, 이를 이용해 객체정보 추출 또는 배경분리 등이 가능하다. 또한, 영상 내에서의 동일객체는 화소 및 움직임 정보가 유사하다는 특징이 있어 이를 객체정보와 함께 이용하는 경우, 공간적 또는 시간적 상관관계에 있는 동일 객체로부터 현재 객체의 예측정보로 이용할 수 있어 부호화 효율을 높일 수 있다. 이는 과거로부터 현재까지 객체기반 부호화 알고리즘<sup>[17][18]</sup> 등에 이용되고 있으며, 대표적인 비디오 코덱으로 MPEG-4<sup>[19]</sup>을 들 수 있다. 특히, 비디오 부호화에서 다양한 화소 값을 갖는 객체영역에 비해 배경영역은 높은 공간적 및 시간적 유사성으로 인해 움직임이 없거나 완만한 특성을 갖고 있어 이를 현재 예측에 이용한다면 높은 비디오 부호화 효율을 얻겠다<sup>[20]</sup>.

HEVC는 최신의 비디오 코덱 표준으로 기존의 H.264/AVC와 같은 블록 기반의 하이브리드 비디오 부호화 구조를 갖으며, H.264/AVC 대비 두 배의 압축 효율을 보인다<sup>[21]</sup>. HEVC는 최소 8x8에서 최대 64x64 까지의 크기를 갖는 최대코딩단위(Largest Coding Unit; LCU)들로 Slice를 구성하고, 각 LCU들은 계층적 쿼드트리 구조로 재귀적 분할되며 부호화를 수행한다. 이렇게 부호화를 수행하는 부호화 단위를 코딩단위(Coding Unit; CU)라 하며, 이 CU들은 추가적으로 예측단위(Prediction Unit; PU)와 변환단위(Transform Unit; TU)로 세분화 된다<sup>[13][22]</sup>.

이러한 재귀적 유닛 구조에서 LCU 내의 최적의 CU 분할 정보를 찾기 위해 모든 깊이의 CU를 부호화하여 코딩효율이 최대가 되는 분할구조로 결정하기 때문에 HEVC의 복잡도를 크게 증가시키는 문제점을 갖고 있다. 이러한 문제점을 인지하여, 통계적 분석 방법을 이용한 분할구조 결정방법<sup>[23]</sup>, 주변블록 예측정보를 이용한 현재블록의 부호화 모드 조기결정 방법<sup>[24]</sup>, 시간적 이전 프레임을 이용한 현재 프레임 분할구조 결정방법<sup>[25]</sup>, 분할구조 조기결정 방법

a) 경희대학교 컴퓨터공학과 (Dept. of Computer Engineering, Kyung Hee University)

‡ Corresponding Author : 박광훈 (Gwang Hoon Park)  
E-mail: ghpark@khu.ac.kr  
Tel: +82-31-201-3680

※ 본 연구는 미래창조과학부 및 정보통신산업진흥원의 대학IT연구센터육성 지원사업/IT융합고급인력과정지원사업의 연구결과로 수행되었음. (NIPA-2014-H0301-14-1002)

· Manuscript received June 20, 2014 Revised August 20, 2014 Accepted August 29, 2014

[26][27] 등의 많은 연구들이 부호화 복잡도를 낮추는 방법으로 제시되었다.

본 논문에서는 부호화 복잡도 감소에 있어서 기존의 제안방법과 달리, 깊이정보 획득이 용이해진 영상시스템 환경으로의 변화에 대응하여 2D 비디오 부호화에 깊이정보를 이용하는 방법, 최근 표준화가 완료된 HEVC 비디오 코덱에 깊이정보를 이용한 부호화 방법을 제안한다. 이는 HEVC의 높은 압축률로 인해 증가된 부호화 복잡도를 감소시킬 수 있는 효과적인 방안이라 할 수 있다.

본 논문의 구성은 다음과 같다. II장에서는 HEVC의 계층적 CU 부호화 방법에 대해 설명하고 III장에서는 화면 간 예측 부호화 방법에 대해 설명하며, 그리고 IV장에서는 깊이정보 영상에서 객체정보를 추출하는 방법에 대해 기술한다. V장에서는 깊이정보를 이용한 HEVC 인코더의 계산 복잡도를 줄일 수 있는 움직임 예측방법과 분할구조 결정 방법에 대해 논의하고, VI에서는 제안하는 알고리즘에 대하여 기술하였다. VII장에서는 제안한 알고리즘에 대한 성능평가 실험을 수행하였으며 마지막으로 VIII장에서는 결론을 도출하였다.

## II. HEVC 계층적 부호화 방법

HEVC에서는 영상을 효율적으로 부호화하기 위해 부호화 트리 단위(Coding Tree Unit; CTU)로 영상을 겹치지 않게 나누고 각각의 CTU들은 쿼드트리 형태로 분할되어 부호화를 수행한다. 부호화 수행에 있어서 다양한 크기의 정사각형 모양으로 정의된 CU를 기본 부호화 단위로 사용하며, CU의 크기는 가장 큰 단위인 LCU부터 가장 작은 단위인 SCU(Smallest Coding Unit)까지 존재한다. CU는 가로 세로 각각 절반 크기의 4개의 CU로 분할할 수 있으며, 이때 분할된 CU는 재귀적으로 다시 가로 세로 각각 절반 크기의 4개의 CU로 분할될 수 있다. 이러한 CU의 재귀적분할은 인코더에서 정의한 회수까지 가능하며, 이를 통해 다양한 크기를 갖는 CU에 대해 그 크기에 따라 깊이(depth)로 표현된다. LCU의 경우 깊이 0으로 표현되며 분할 회수에 따라 깊이가 1식 증가한다<sup>[22]</sup>.

<그림 1>은 HEVC 참조 소프트웨어(reference software) HM 12.0에서 부호화를 수행한 ‘BasketballPass’ 영상의 예를 보여주는 그림이다. 그림에서 ‘BasketballPass’ 영상은

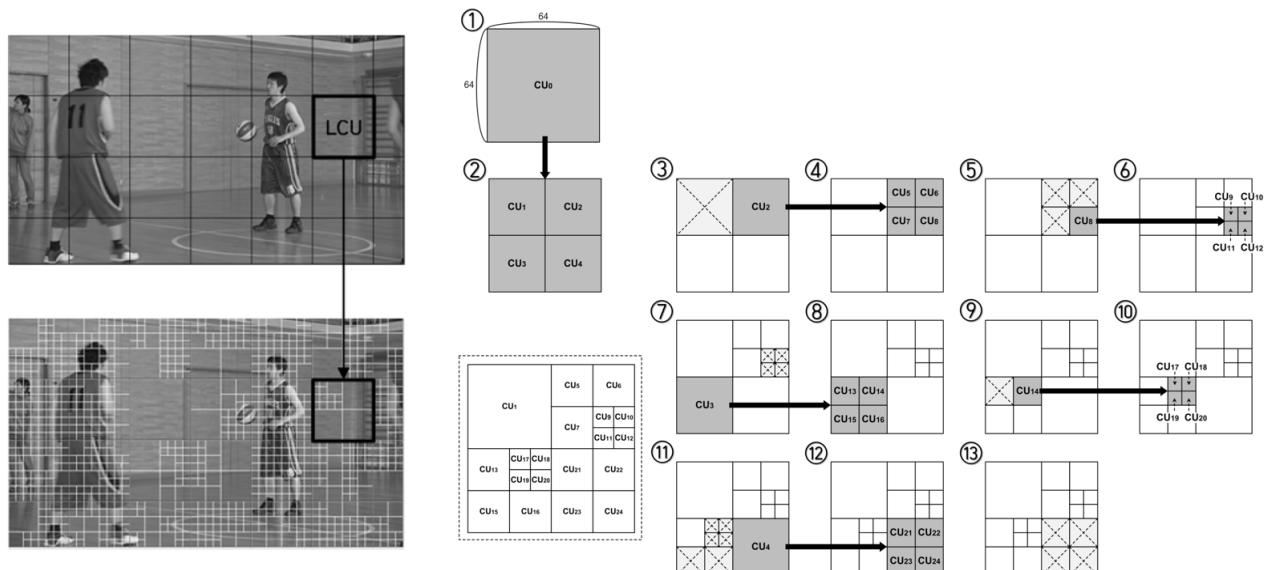


그림 1. LCU 단위로 CU를 분할하는 방법  
Fig. 1. CU splitting for each LCU in HEVC

HEVC 테스트 시퀀스(Test sequence) 영상 중 하나이며<sup>[28]</sup>, 그림에서 가로 세로 64x64 화소 단위의 부호화 단위(LCU)의 CU 분할을 수행하는 영상의 예를 나타내고 있다.

<그림 1>에서 영상을 LCU 단위로 순차적으로 분할하고, LCU 단위로 CU 분할구조를 결정하여 부호화하는 예를 나타내고 있다. 그림에서 부호화 기본이 되는 LCU의 크기를 64x64 화소로, 최대 깊이는 3으로 정의하여 깊이 별로 분할하지 않는 CU의 경우에는 2Nx2N 크기로, 분할하는 CU의 경우에는 NxN 크기의 CU 4개로 분할되는 것을 보여주고 있다. CU에 대한 분할정보는 SCU를 제외한 모든 CU에 포함되어 있으며, 해당 깊이의 CU를 분할하지 않는 경우에는 분할정보에 ‘0’을 저장하고, 분할할 경우에는 ‘1’을 저장한다.

HEVC 인코더에서 LCU에 대한 CU 단위의 분할구조 결정은 LCU 내의 영상 특성에 따라 결정된다. 인코더에서 LCU 부호화시 모든 깊이의 CU에 대해 부호화를 수행하고, 부호화 정보를 CU 후보로 저장하여, 코딩효율이 가장 높은 분할구조를 선택해 결정한다. <그림 2>는 HEVC 인코더에서의 LCU 단위의 분할구조 결정방법의 예를 보여주고 있다.

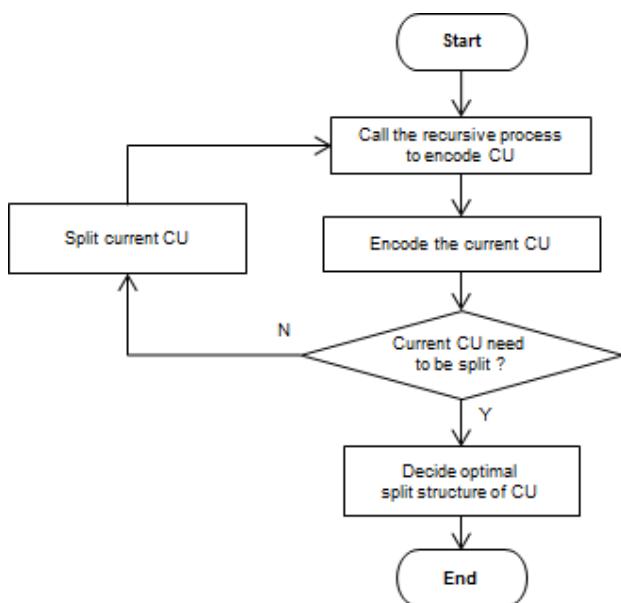


그림 2. LCU 단위 분할구조 결정 방법

Fig. 2. Method to determine the split structure for LCU

<그림 2>의 과정에서 부호화 정보를 갖는 CU 후보에 대한 선택은 율-왜곡 최적화(Rate-distortion Optimization) 방법<sup>[29]</sup>에 의해 결정되며, 이를 통해 가장 부호화 효율이 좋은 분할구조로 결정된다. <그림 1>에서 이러한 과정을 통해 결정된 LCU에 대한 CU 분할구조를 보여주고 있다. <그림 1>을 통해 LCU에 대한 부호화 결과, 복잡하거나 움직임이 많은 영역에서는 대체로 CU의 크기가 작으며 깊이가 큰 CU로 부호화가 수행되고, 단순하거나 움직임이 적은 영역에서는 대체로 CU의 크기가 크며 깊이가 작은 CU로 부호화가 수행됨을 알 수 있다.

### III. HEVC의 화면 간 예측 부호화 방법

HEVC에서 화면 간 예측 방법은 기존 코덱에서와 같이 영상의 중복성 제거를 위한 ME/MC 개념을 기본으로 PU 단위로의 움직임 예측을 수행한다. 움직임 예측은 제한된 탐색 범위 내에서 현재블록과 가장 근접한 정보를 갖는 위치를 찾아 움직임 벡터를 가져오는 것으로써, HEVC에서는 크게 Advanced Motion Vector Prediction(AMVP)<sup>[30]</sup>와 Merge<sup>[31]</sup>가 있다. <그림 3>에서 AMVP에서의 탐색 방법을 나타내고 있다.

<그림 3>에서 AMVP 탐색 방법은 참조 프레임으로부터 움직임 탐색 영역 안의 위치들에 다이아몬드 탐색 방법을 이용하여 1차로 현재 PU와 제일 근접한 블록이 존재하는 위치를 찾고 2차로 2포인트 탐색 방법을 이용하여 해당 위치를 중심으로 세밀하게 주변 위치들과 비교하여 최적의 움직임 벡터를 구한다. <그림 3>에서 현재 위치의 1번 위치에서 다이아몬드 탐색 방법을 통해 4번 위치를 찾고, 해당 지점으로부터 2차로 2포인트 탐색 방법을 이용하여 5번 위치에서 최적의 움직임 벡터를 구하는 예를 보여주고 있다.

또한 Merge 방법에서 SKIP 모드로 부호화 할 수 있는데, SKIP 모드는 해당 PU의 잔여신호를 제외한 움직임 정보만을 부호화하여, 현재 PU와 참조블록의 화소 값이 모두 동일하며 어떠한 화소 정보도 추가되지 않고 현재 PU 그대로 가져오도록 한다. SKIP 모드는 2Nx2N의 PU에만 적용되

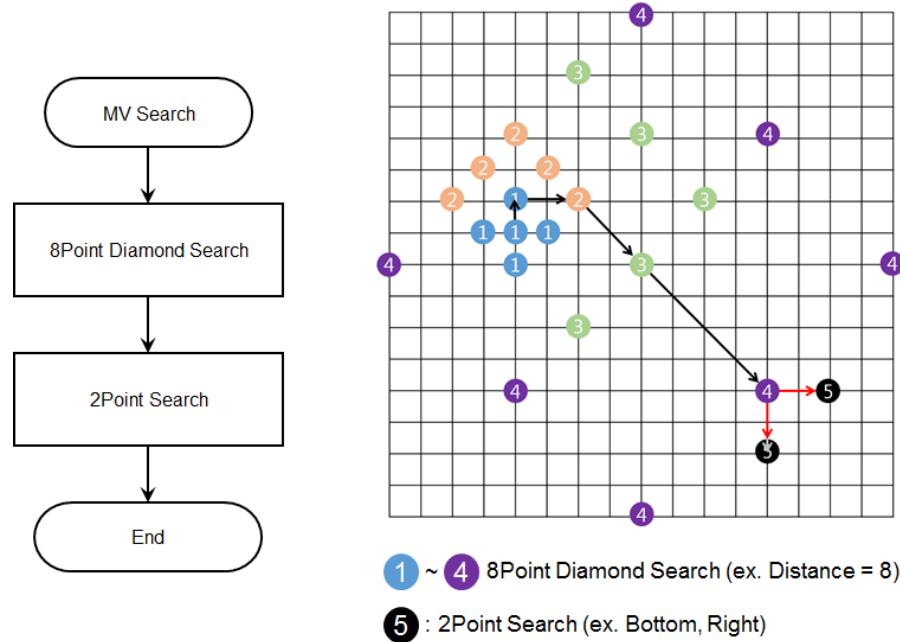


그림 3. AMVP 예  
Fig. 3. Example of AMVP

며, 해당 PU가 SKIP 모드로 부호화 되었는지의 확인 여부는 SKIP\_FLAG를 사용하여 판단한다.

#### IV. 깊이정보를 이용한 영상의 객체정보 추출 방법

깊이정보는 카메라와 물체간의 거리를 나타내는 정보로

서, <그림 4>의 'Kendo' 영상에 대한 일반 영상과 그것의 깊이정보 영상을 통해 확인할 수 있다. 그림에서 (c)는 깊이정보 영상 (b)에 대해서 깊이정보 값의 평균값을 이용하여 이진화를 수행한 영상이다.

<그림 4>에서의 (b)와 같이 동일 객체 내에는 대체로 거리 간격이 유사하기 때문에 깊이정보 값 역시 동일하게 나타난다. 이러한 점을 통해, 깊이정보 영상에서 라벨링 알고리즘 등의 방법을 적용하면 유사한 깊이정보 값을 갖는 영

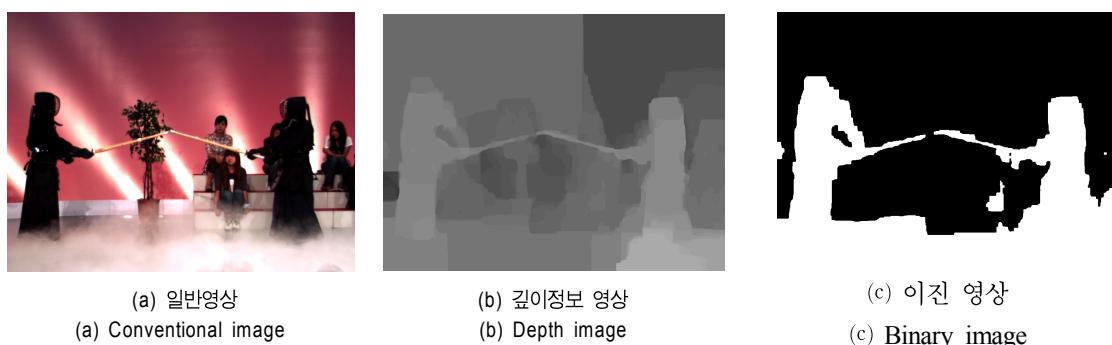


그림 4. 일반영상과 깊이정보 영상  
Fig. 4. Images taken from Kendo

역끼리 뮤이게 되어 객체를 추출할 수 있어 객체와 배경을 분리할 수 있게 한다. 라벨링 알고리즘을 통해 객체정보를 추출하는 방법은 대표적으로 Gaussian Mixture Model (GMM) 방법을 들 수 있으며<sup>[32]</sup>, GMM은 가우시안 분포를 이용한 픽셀기반 배경분리 방법으로 K개의 배경분포 모델을 가지고 배경을 분리하는 방식이다. 그 외 라벨링 방법으로 Kernel Density Estimator, Mean-shift based estimation, Sequential Kernel Density Approximation, Eigenbackground 방법 등이 있다<sup>[33]</sup>. 또한, 깊이정보 영상에서 객체는 대체로 배경보다 카메라 가까이 위치하기 때문에 <그림 4>에서의 (c)와 같이 식 (1)의 방법으로 깊이정보 영상 프레임에서 깊이정보 값의 평균을 문턱값(Threshold)으로하여 이진화를 수행할 경우 객체와 배경을 쉽게 분리할 수 있다.

$$T = \text{Average}(\sum f_{(x,y)})$$

$$d_{(x,y)} \begin{cases} 0 & \text{if } f_{(x,y)} < T \\ 1 & \text{otherwise} \end{cases} \quad (1)$$

이외 <그림 5>에서와 같이 Microsoft사가 개발한 키넥트 (Kinect)를 이용하는 경우 깊이정보 영상으로부터 라벨링된 영상을 얻을 수 있어, 객체와 배경을 분리할 수 있다. 그림에서 (c)는 사람을 인지하여 라벨링한 영상으로 키넥트를 통해 실시간으로 제공되는 영상이다.

이상의 방법들로 깊이정보 영상을 이용하는 경우에 있어서, 추출된 객체정보로부터 객체 및 배경영역에 대해 구별할 수 있어, 각 영역 특성을 고려한 부호화 방법이 가능할

수 있겠다. 또한, 현재 HEVC 코덱은 H.264/AVC 대비 2배 이상의 압축 성능을 보이지만, 이에 따른 연산 복잡도가 H.264/AVC에 비해 크게 증가하여<sup>[34][35]</sup>, HEVC 상용화, 실시간 부호화를 위해서는 최적화 및 부호화 복잡도 감소를 위한 연구가 요구되고 있다.

깊이정보 획득이 용이해진 최근의 영상 시스템 추세에 대응한다면 깊이정보를 이용한 영상 부호화 방법은 부호화 효율을 높이는 효율적인 방법이라 할 수 있다. 따라서, 본 논문에서는 <그림 4>, <그림 5>에서와 같이 깊이정보 영상으로부터 객체영역과 배경영역을 분리하여 영역 특성에 맞는 부호화 방법을 통해 부호화 복잡도를 낮추는 깊이정보를 이용한 HEVC 인코더를 제시하고자 한다.

## V. 객체정보와 CU와의 상관관계

### 1. CU 분할 구조의 특성

CU 분할 구조는 영상의 특성에 따라 결정되는 특성을 보여주고 있다. 이를 <그림 6>을 통해 객체정보와 비교해 보면 확인할 수 있는데, 사람이나 사물 또는 그것의 경계와 같이 움직임이 많거나 복잡한 영상에서는 LCU 단위의 분할구조가 배경영역에서의 분할구조와 비교해 극명한 차이를 이루는 것을 알 수 있다. 그림에서 HEVC 참조 소프트웨어인 HM 12.0 부호화를 ‘Kendo’ 영상에 수행한 분할구조를 나타내고 있으며, 객체정보 영상은 깊이정보 영상에 Threshold를 수행한 수 행한 영상이다.

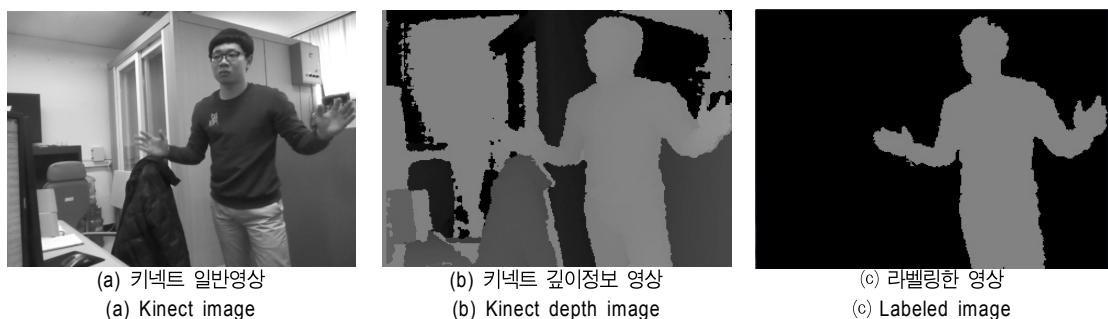


그림 5. Microsoft 키넥트를 통해 추출한 영상의 예  
 Fig. 5. Example of extracted images from Microsoft Kinect

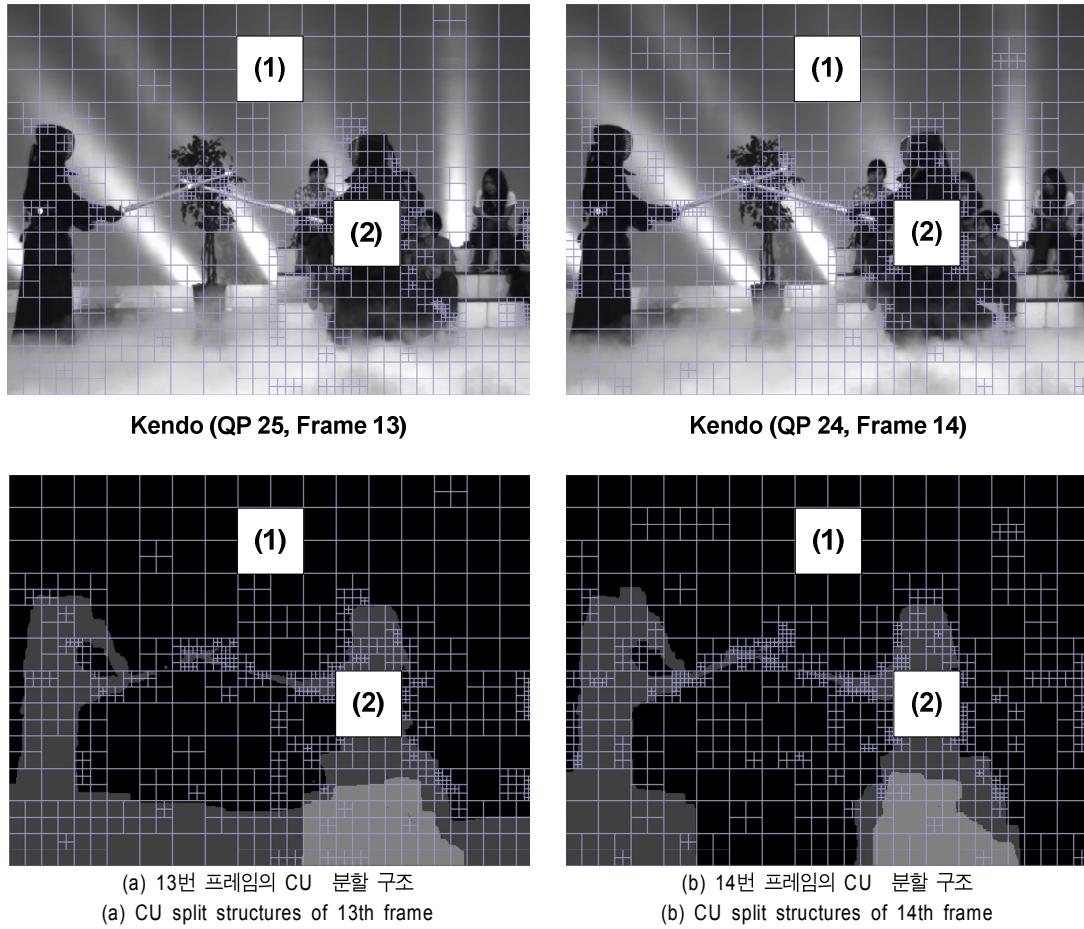


그림 6. 객체정보에 따른 CU 분할 구조 차이의 예  
Fig. 6. Example of different CU split structures depending on object information

<그림 6>에서 (1)과 같이 배경에 위치하는 LCU의 경우 사람에 위치하는 (2)의 LCU와 달리 화소 변화가 적어 상대적으로 큰 크기의 CU로 분할되어 부호화가 수행되고 있다. 또한 <그림 6>의 (a)와 (b)의 (1)과 (2)의 LCU 비교를 통해 시간적으로 동일한 위치의 LCU에 대한 분할구조가 상당한 유사성을 갖는 것을 알 수 있다. 이러한 점들은 CU 분할구조를 결정하는데 있어서 분할구조의 형태를 미리 예측할 수 있는 좋은 정보가 돼, 객체정보를 이용하는 경우 배경영역에 위치하는 LCU는 대체로 깊이가 크지 않으며 CU 크기가 큰 형태로 분할구조를 갖는 것을 예상할 수 있고, 시간적으로 동일한 위치의 LCU에 대해서 객체정보를 참조하면 분할구조가 유사하게 될 것임을 예상할 수 있다.

따라서, 객체정보를 이용할 수 있다면, 모든 깊이의 CU에 대해 부호화를 수행하여 분할구조를 결정하는 기존의 방법에서 객체 및 배경 영역에 따른 분할구조를 예측할 수 있어 부호화 복잡도를 감소시킬 수 있겠다.

## 2. SKIP 모드 특성

<그림 7>은 ‘Kendo’ 영상을 HEVC 참조 소프트웨어인 HM 12.0을 이용한 부호화 결과를 PU 단위로 보여주고 있고, (a)에서 SKIP 모드로 부호화된 PU에 대해서는 흰색 영역으로 표시하였다.

<그림 7>을 보면 객체들로 구성된 영역과 배경을 구성하

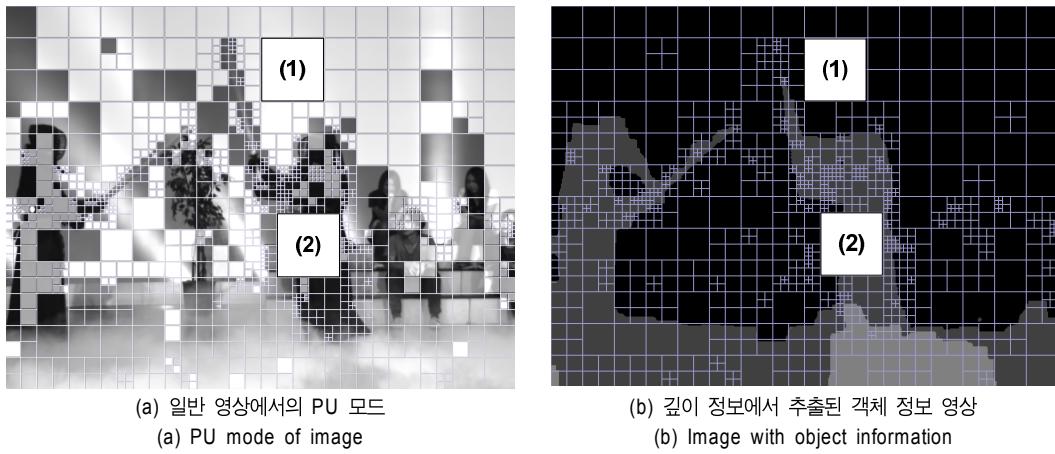


그림 7. 객체 정보에 따른 PU 모드 정보 차이의 예  
 Fig. 7. Example of different PU mode depending on object information

는 영역의 PU 정보를 살펴보면 차이를 보이고 있다. 그림에서 (1)과 같이 배경영역에 위치하는 PU는 (2)와 같이 객체영역에 위치하는 PU에 비해서 화소 변화가 적고 현재블록과 인접블록 간의 유사성이 높아 대체로 Merge 방법의 SKIP 모드 부호화가 많이 수행돼, PU의 깊이가 작고 크기가 큰 형태로 존재한다.

<그림 7>을 통해 배경영역에 위치한 PU의 경우 대체로 큰 크기를 갖으며 SKIP 모드로 부호화가 될 것임을 예상할

수 있다. 객체정보를 이용할 수 있다면, 배경영역에 한하여 PU 부호화 시에 작은 깊이를 갖는 PU에 대한 SKIP 모드 부호화가 선택될 확률이 높아, 이를 통해 PU 예측을 제한할 수 있어서 부호화 복잡도를 감소시킬 수 있겠다.

### 3. 움직임 예측 특성

<그림 8>은 HEVC 참조 소프트웨어인 HM 12.0을 통해

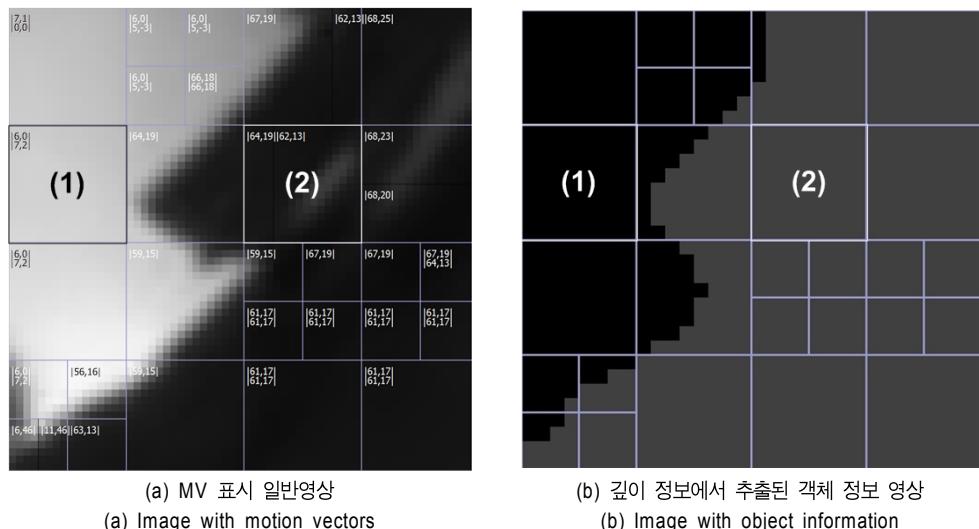


그림 8. 객체 정보에 따른 움직임 벡터 차이의 예  
 Fig. 8. Example of different motion vectors depending on object information

부호화 수행 결과를 보여주고 있다. 그림에서 (a) 일반영상에 표시되는 숫자는 해당 PU의 움직임 벡터 값을 int-pel 단위로 나타내고 있다.

<그림 8>의 (a)와 (b)를 보면 객체 경계를 중심으로 움직임 벡터 값에 차이가 남을 알 수 있는데, (1)이 위치한 배경 영역의 움직임 벡터 값이 (2)가 위치한 객체영역의 움직임 벡터 값 (64, 19), (62, 13)에 비해 (6, 0), (7, 2)의 값으로 대체로 0에 가깝게 분포되고 있다. 이를 통해, 배경영역에서는 움직임 탐색 범위를 객체영역과 달리 제한을 두어 작은 범위로도 가능함을 유추할 수 있다. 현재 2차원 동영상 코덱에서는 단순히 화소의 집합으로만 구성되어 있기 때문에 정확한 객체정보를 기반으로 하는 탐색을 수행할 수 없어 해당 PU의 움직임 벡터를 구하기 위한 많은 계산량이 필요하다.

그러므로 움직임 예측을 하는데 있어서 깊이정보 영상을 이용할 수 있다면, 객체 정보를 통해 그에 따라 탐색 범위를 조정하는 방법을 고려해볼 수 있으며, 이러한 방법은 객체정보에 따라 탐색을 제한하는 방법으로 탐색이 필요하지 않는 영역을 구분하여 제거함으로써 부호화 복잡도를 감소시킬 수 있겠다.

## VI. 제안 방법

HEVC의 향상된 압축성능으로 인한 증가된 계산 복잡도를 낮추려는 연구는 HEVC의 상용화 및 실시간 부호화를 위한 매우 유용한 연구라 할 수 있다. 특히, HEVC 인코더 측면에서 CTU 단위로의 모든 깊이에 대한 CU 부호화를 통한 분할구조 결정방법은 부호화 복잡도의 큰 비중을 차지하고 있다 [36]. 이에, 분할구조 결정과정에서 분할구조에 대한 예측이 가능해 모든 깊이의 CU에 대한 부호화 과정을 통하지 않고 결정할 수 있다면 부호화 복잡도의 큰 감소를 이룰 수 있겠다.

부호화 복잡도 감소를 위한 방안으로써 깊이정보를 이용한 부호화 알고리즘의 적용은 좋은 제안이 될 수 있다. 특히, 본 논문의 II, III, IV, V 장의 내용들을 통해 깊이정보로부터 객체정보를 분리하여 객체 또는 배경 각 영역을 고려한 예측방법을 제안해 볼 수 있다.

현재 최신의 영상 제품들의 2차원 비디오 부호화 환경 아래에서는 깊이 카메라의 대중화로 인해 깊이정보를 이용

하는 것이 가능한 상황으로 변하고 있다. 이러한 점은 최신의 2차원 비디오 코덱인 HEVC 코덱의 인코딩 환경에서 깊이정보 이용이 가능해져, 향후 2차원 비디오 코덱에서 화소집합 단위의 블록기반 부호화 알고리즘에서 객체기반 부호화 알고리즘이 적용 가능해, 객체정보를 이용한 HEVC 부호화 방법을 제안할 수 있겠다.

또한, 깊이정보 영상을 획득할 수 있는 최신의 영상 제품들은 대체로 핸드헬드(Handheld) 제품과 같은 저전력 제품들이기 때문에, 계산 복잡도를 고려하여, 완벽하게 객체를 분리하는 알고리즘을 이용하기보다는 식 (1)과 같은 문턱값(Threshold)을 이용한 이진화 방법으로의 객체 분리 방법을 본 논문에서 적용하였다. 본 논문에서 제안되는 방법은 깊이정보를 이용하여 객체와 배경 각 영역을 대략적으로 구한 후, 이를 이용하여 주변블록의 SKIP 모드를 참조한 CU 분할구조 조기결정(방법 1), 시간적 위치의 LCU 정보를 이용한 CU 분할구조 결정(방법 2), 배경영역에 따른 움직임 벡터 탐색 범위 제한(방법 3)으로 구성된다.

### 1. 깊이정보와 SKIP 모드를 이용한 CU 분할 구조 조기 종료

본 논문에서 제안된 첫 번째 방법은 <그림 9>과 같이 깊이정보를 통해 얻어진 배경정보와 PU의 SKIP 모드를 이용하여 LCU 내부의 CU 분할구조를 조기 결정하는 방법이다. 그림에서 LCU 크기를  $64 \times 64$ 로 설정하고 'Kendo' 영상에 부호화를 수행해 얻은 분할구조와 깊이정보 영상에 이진화를 수행해 얻은 객체정보 영상을 나타내고 있다.

<그림 9>와 같이 부호화 결과 배경영역에 위치한 CU는 대체로 Merge의 SKIP 모드를 이용한 부호화가 수행되는 경우가 많고, 움직임이 있는 특정객체에 위치한 CU는 그밖의 화면 간 예측 부호화 또는 화면 내 예측 부호화가 수행되는 경우가 많다. 이를 통해 배경영역에 위치한 CU의 경우는 움직임의 변화가 크지 않아 대체로 SKIP 모드 부호화가 수행되는 점을 유추할 수 있어, 제안하는 방법은 현재 CU가 배경에 해당할 경우 부호화 수행과정에서 최적의 CU가  $2Nx2N$  크기의 SKIP 모드로 결정이 될 경우에는 더 이상의 CU 분할을 수행하지 않고 종료하는 방법으로 기

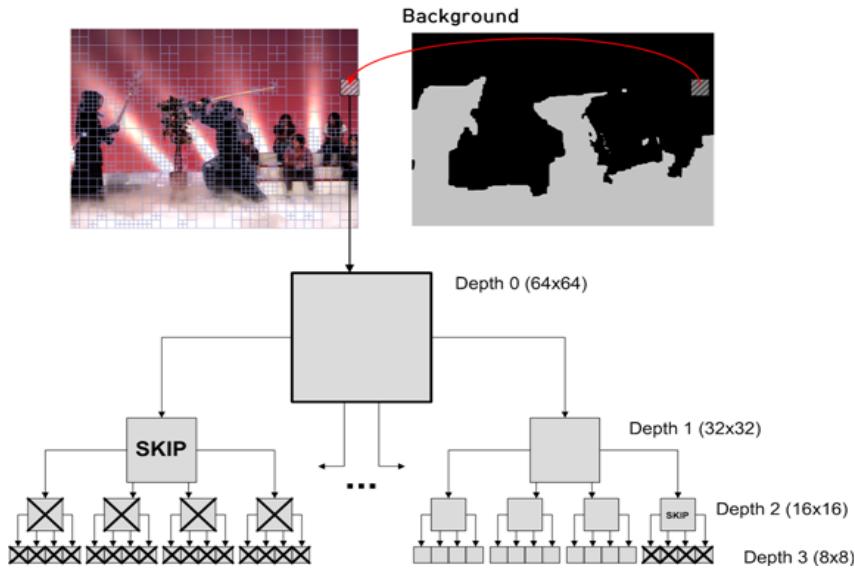


그림 9. 제안하는 방법 1  
 Fig. 9. Proposed method 1

준보다 계산 복잡도를 줄이고자 한다. <그림 9>는 LCU에 대한 부호화 수행과정에서 SKIP 모드가 최적의 CU로 결정되고 배경영역에 있는 경우 해당 CU를 더 이상 분할하지 않고 부호화를 종료하는 과정을 나타내고 있으며, 수행과정은 다음과 같다.

- (1 단계) 현재 CU의 객체정보를 얻고 최적 PU 모드를 결정한 뒤 다음 단계를 수행한다.
- (2 단계) 현재 CU가 가장 작은 크기의 CU인지 판단한다. 가장 작은 크기의 CU일 경우 6단계를 수행한다. 가장 작은 크기의 CU가 아닐 경우 다음 단계를 수행한다.
- (3 단계) 현재 CU가 배경영역에 속하는 CU인지 판단한다. 배경영역에 속하는 CU일 경우 다음 단계를 수행한다. 그렇지 않을 경우에는 5단계를 수행한다.
- (4 단계) 현재 CU가 SKIP 모드로 결정되었는지 판단한다. SKIP 모드로 결정되었다면 6단계를 수행한다.
- (5 단계) 현재 CU의 가로 절반, 세로 절반의 크기를 갖는 4개의 CU로 분할하고 1단계를 수행한다.
- (6 단계) 현재 크기의 CU를 부호화하여 CU 분할 구조 후보에 저장하고 다음 단계를 수행한다.
- (7 단계) 현재 LCU에 저장되어 있는 CU 후보들 중 율-왜곡 최적화 방법(Rate-distortion Optimization)을 통하여 화질 및 비트량 측면에서 가장 효율적인 CU의 분할 구조를 선택하고 종료한다.

## 2. 깊이정보와 시간적 위치의 CU 분할정보를 이용한 현재 CU의 분할구조 제한

본 논문에서 제안된 두 번째 방법은 <그림 10>과 같이 깊이정보와 시간적 위치의 CU 분할 정보를 이용하여 LCU 내부의 CU 분할 구조를 결정하는 방법이다. 그림에서 LCU 크기를  $64 \times 64$ 로 설정하여 'Kendo' 영상에 부호화를 수행해 얻은 분할구조와 깊이정보 영상에 이진화를 수행해 얻은 객체정보 영상을 나타내고 있다.

시간적 위치의 CU 분할정보를 이용해 현재 CU의 분할구조를 예측하는 방법은 HEVC의 고속화 방법<sup>[25]</sup>으로 제안되었고, [25]에서는 시간적 위치의 CU를 분할구조에 따라 복잡한 분할구조를 갖는 CU와 단순한 분할구조를 갖는 CU로 나누어 현재 CU를 예측하는 방법을 사용하였다. 또한, 단순히 시간적 위치의 CU 분할구조의 복잡 유무에 따라 현재 CU의 분할구조를 예측할 수는 없어, 현재 CU의 분할구조 예측을 위해서는 여러 조건을 추가하여 예측을 정교함을 더하였다. 이와 달리 본 논문에서는 배경영역에 기반한 보다 단순한 방법을 제안한다. 제안된 방법은 배경영역에서의 CU의 분할구조가 대체로 단순하며, 배경영역에 있는 시간적 위치의 CU들은 대체로 비슷한 분할구조를 갖고 있어, 이러한 점에

착안하여 현재 CU가 배경역영에 있고 시간적 위치의 CU가 단순한 분할구조를 갖는 경우 현재 CU의 분할구조를 단순한 분할구조를 갖도록 제한하는 방법이다.

<그림 10>은 현재 CU가 배경역영에 해당할 경우 시간적 상관관계에 있는 LCU의 최소 크기 CU가 제안된 크기 이하가 아니라면, 현재의 LCU도 해당 범위 이하의 CU 분할을 수행하지 않고 종료하는 방법을 나타내고 있다. 본 논문에서는 제안 크기를  $32 \times 32$ 으로 설정하여 제안방법 2를 수행하도록 하였는데, 3DTV 9개의 테스트 영상에 대해 제안방법 2의 CU 제안 크기를  $64 \times 64$ ,  $32 \times 32$ ,  $16 \times 16$ 으로 놓고 실험한 결과  $32 \times 32$ 로 설정하는 경우의 성능이 가장 우수함을 실험적으로 확인할 수 있었다. <그림 10>에 대한 수행과정은 다음과 같다.

- (4 단계) 현재 CU의 크기가  $32 \times 32$ 보다 작으면 판단한다.  $32 \times 32$ 보다 작다면 다음 단계를 수행한다.  $32 \times 32$ 보다 크거나 같다면 6단계를 수행한다.
- (5 단계) 참조 프레임으로부터 대응되는(Co-located) LCU의 최소 CU 크기가  $32 \times 32$ 보다 크거나 같은지 판단한다.  $32 \times 32$ 보다 크거나 같으면 7단계를 수행한다.  $32 \times 32$ 보다 작다면 다음 단계를 수행한다.
- (6 단계) 현재 CU의 가로 절반, 세로 절반의 크기를 갖는 4개의 CU로 분할하고 1단계를 수행한다.
- (7 단계) 현재 크기의 CU를 부호화하여 CU 분할 구조 후보에 저장하고 다음 단계를 수행한다.
- (8 단계) 현재 LCU에 저장되어 있는 CU 후보들 중 율-왜곡 최적화 방법(Rate-distortion Optimization)을 통하여 화질 및 비트량 측면에서 가장 효율적인 CU의 분할 구조를 선택하고 종료한다.

- (1 단계) 현재 CU의 객체정보를 얻고 최적 PU 모드를 결정한 뒤 다음 단계를 수행한다.
- (2 단계) 현재 CU가 가장 작은 크기의 CU인지 판단한다. 가장 작은 크기의 CU일 경우 7단계를 수행한다. 가장 작은 크기의 CU가 아닐 경우 다음 단계를 수행한다.
- (3 단계) 현재 CU가 배경에 속하는 CU인지 판단한다. 배경에 속하는 CU일 경우 다음 단계를 수행한다. 그렇지 않을 경우에는 5단계를 수행한다.

### 3. 깊이정보를 이용한 움직임 예측의 탐색범위 제한

본 논문에서 제안된, 세 번째 방법은 <그림 11>과 같이 깊이정보를 이용하여 광범위한 움직임 예측을 수행하는 HEVC 인코더의 움직임 예측 탐색범위를 설정값의 1/2로 제한하는 방법이다. 그림에서 LCU 크기를  $64 \times 64$ 로 설정하여 ‘Kendo’ 영상에 부호화를 수행해 얻은 분할구조와 깊

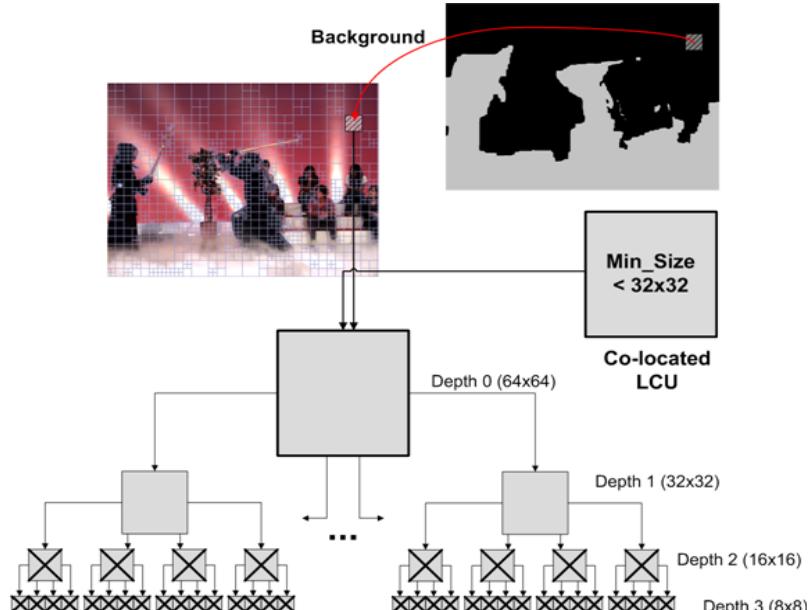


그림 10. 제안하는 방법 2  
Fig. 10. Proposed method 2

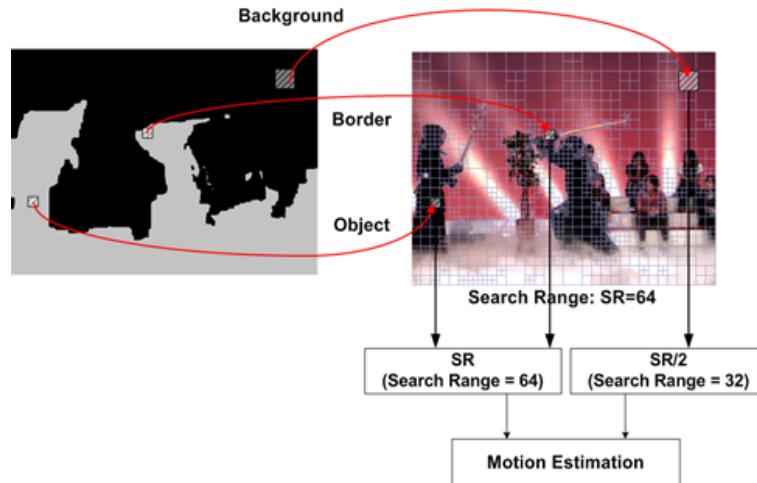


그림 11. 제안하는 방법 3  
 Fig. 11. Proposed method 3

이정보 영상에 이진화를 수행해 얻은 객체정보 영상을 나 타내고 있다.

<그림 11>과 같이 현재 PU가 움직임 탐색을 수행하는 경우에 움직임 예측 범위를 기준보다 1/2의 값으로 설정하여 기준보다 계산 복잡도를 줄이고자 한다. <그림 11>에서 현재 CU가 배경영역에 해당할 경우 움직임 예측 탐색범위

를 설정 값 64에서 1/2인 32로 설정하고, 그 외의 경우에는 기존방법을 따르게 된다.

#### 4. 깊이정보를 이용한 HEVC 인코더에서의 부호화 방법

본 논문에서 제안된 방법을 <그림 12>의 블록도로 나타

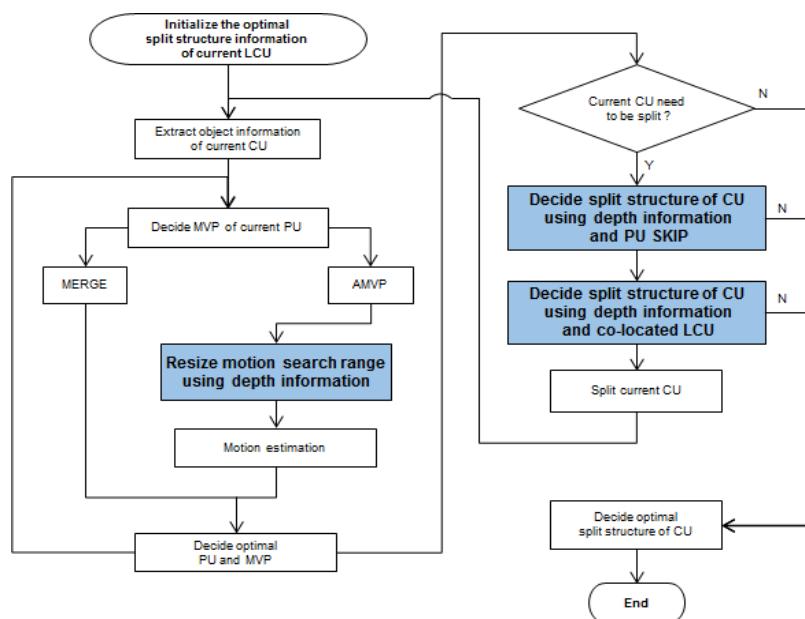


그림 12. 제안하는 방법  
 Fig. 12. Flow chart of proposed methods

내고 있다.

## VII. 성능 평가

본 논문에서 제안한 알고리즘의 우수성을 확인하기 위하여 HEVC 참조 소프트웨어인 HM 12.0에 구현하여 HM 12.0과의 비교실험을 수행하였다. 실험은 Intel(R) Xeon(R) CPU E5-2620 @ 2.0GHz, 32.0GB 메모리, Windows 7 64비트 O/S 환경에서 진행하였다.

실험영상은 키넥트를 통해 추출한 일반(Texture) 영상과 깊이(Depth) 정보 영상, JCT-3V에서 제공하는 일반영상과 깊이정보 영상을 사용하였다. 본 논문에서 사용된 <표 1>

의 실험영상은 3DV Test Sequence로 JCT-3V에서 제공하는 실험영상이며, Kinect Sequence는 키넥트에서 추출한 영상을 나타낸다. <그림 13>과 같이 키넥트 영상의 깊이정보 라벨링은 사전에 키넥트 영상에서 라벨링된 객체정보 영상을 추출하여 실험에 적용하였다. 3DV 영상의 깊이정보 라벨링은 배경이 카메라로부터 멀리 떨어져 있고, 객체가 카메라로부터 가까이 있다는 점을 이용하여 각 프레임에 해당하는 깊이정보 화소 전체의 평균값을 적용하여 객체정보가 배경과 객체로 구분되도록 처리하여 실험에 적용하였다.

실험조건은 HEVC의 공통 실험 조건의 저지연 환경(Low-delay)과 임의 접근(Random-access) 환경, 그리고 Main 프로파일을 준수하였다<sup>[28]</sup>. 제안하는 방법의 계산 복



(a) Kinect 1라벨 일반영상과 이진영상  
(a) Image and 1 labeled binary image taken from Kinect

(b) Kinect 3라벨 일반영상과 이진영상  
(b) Image and 3 labeled binary image taken from Kinect

그림 13. 키넥트로 획득한 영상 및 이진영상  
Fig. 13. Image and binary image taken from Kinect

표 1. 실험 결과

Table. 1. Experiment result

		Low-delay Main				Random-access Main			
		Y BD-rate	U BD-rate	V BD-rate	Encoding Time	Y BD-rate	U BD-rate	V BD-rate	Encoding Time
3DV Test Sequences	Balloons	0.5%	1.1%	0.9%	69.2%	0.3%	0.2%	0.1%	64%
	Kendo	0.4%	0.7%	0.7%	64.3%	0.4%	0.4%	0.2%	60%
	Newspaper_CC	0.2%	-0.2%	-0.3%	59.3%	0.2%	-0.2%	-0.1%	55%
	Poznan_Hall2	0.4%	-0.1%	0.2%	46.9%	0.3%	-0.9%	-1.1%	43%
	Poznan_Street	0.6%	-1.2%	-0.7%	41.3%	0.5%	-0.7%	-0.7%	39%
	GT_Fly	0.4%	-0.1%	0.5%	60.7%	0.7%	0.1%	0.2%	58%
	Undo_Dancer	0.3%	-0.8%	0.3%	73.7%	0.4%	-0.4%	-0.3%	69%
	Shark	0.3%	0.5%	0.7%	63.1%	0.5%	0.0%	0.6%	59%
Kinect Sequences	1Label	0.9%	1.0%	0.8%	50.2%	0.4%	-0.4%	-0.1%	47%
	3Labels	0.8%	0.1%	0.3%	68.1%	0.5%	-0.2%	0.2%	65%
3DV Average		0.4%	0.0%	0.3%	59.8%	0.4%	-0.2%	-0.1%	55.9%
Kinect Average		0.9%	0.5%	0.6%	59.2%	0.4%	-0.3%	0.0%	55.9%
Average		0.5%	0.1%	0.3%	59.7%	0.4%	-0.2%	-0.1%	55.9%

잡도를 측정하기 위해 실험영상을 인코더로 부호화하여 비트스트림을 생성하는 과정의 평균시간을 측정하였다. 또한 부호화 효율을 측정하기 위해 평균적인 bit-rate 감소량을 나타내는 BD-Bitrate 방법<sup>[37]</sup>을 사용하였다. 실험에 사용한 3DTV 실험영상은 좌, 우, 중앙의 3개의 시점은 가지고 있어 그 중 중앙시점만을 실험에 사용하였다.

<표 1>는 VII장에서 제안한 방법을 이용하여 부호화한 시간과 BD-Bitrate를 나타낸 것이다. 실험결과는 저지연 환경(Low-delay)에서는 기존 대비 평균 40%, 임의 접근 환경(Random-access)에서는 기존 대비 평균 45%의 부호화 복잡도 감소가 있었으며, BD-Bitrate는 각각 평균 0.5%, 0.4%의 손실이 있었다. 이는 본 논문에서 제안한 배경영역에서의 제안방법의 유효함을 입증한 결과라 할 수 있다. 또한, 평균적으로 저지연 환경보다는 임의접근 환경에서의 성능향상이 뛰어난데, 이는 일반적으로 임의접근 환경의 조건인 계층적 B 예측 구조에서는 이전 프레임 정보뿐만 아니라 이후 프레임 정보까지 고려하고 있어 저지연 환경의 조건인 GPB 구조보다 부호화 효율이 뛰어나 큰 크기의 CU, SKIP 모드를 이용한 부호화가 더 많이 이루어지고 있기 때문이다.

실험을 통해 배경에서는 화소 및 움직임의 변화량이 많지 않아 제안하는 방법 모두를 적용하더라도 복잡도 측면에서는 큰 향상이 있지만 부호화 효율 성능에서는 큰 차이

가 없는 것으로 분석된다.

## VIII. 결 론

본 논문에서는 비디오 코딩 환경의 변화로 깊이정보 획득이 용이해지는 추세에 대응하여, 2D 비디오 환경에서 깊이정보로부터 객체정보를 유도하고 이를 이용한 비디오 부호화 방안에 다루었다. 제안한 연구는 기존의 2D 비디오 코딩 환경의 제약으로 블록기반 비디오 코딩 알고리즘만이 시장에서 사용되는 상황에서 벗어나 최신의 영상관련 제품들에서 객체기반 비디오 코딩 알고리즘이 실제 사용가능해질 수 있음을 보여주는 연구라 하겠다.

실험 결과를 통해 주목할 점으로, 가정용 게임 단말기에 이용되는 키넥트를 이용한 결과에서 화상통신의 상황과 유사한 1 라벨의 영상에서 화질의 큰 손실없이 복잡도가 절반으로 줄어들었음을 확인할 수 있는데, 이는 HEVC가 주로 사용될 최신의 영상 관련 제품, 특히 깊이 카메라가 결합된 스마트폰, 태블릿 PC, 웹캠 등의 사용자 중심의 비디오 코딩 환경을 갖는 핸드헬드 제품에서 저복잡도의 비디오 코덱으로 유용하게 활용될 것으로 예상된다. 또한, 본 논문의 연구가 기존의 코덱들에도 적용되어 기존대비 저복잡도 코덱으로 사용이 가능할 것으로 예상된다.

본 논문에서 얻어진 결과를 통해, 향후 인코더 측면에서의 부호화 복잡도 감소만이 아닌 디코더 측면에서의 코딩 효율 향상까지 이어지는 완전한 객체기반 비디오 코덱으로의 연구가 진행되어야 하겠다.

### 참 고 문 현 (References)

- [1] Kinect camera, <http://www.xbox.com/en-US/kinect/default.htm>
- [2] Project Tango, <http://www.google.com/atap/projecttango/>
- [3] Intel RealSense 3D Camera, <http://www.intel.com/content/www/us/en/architecture-and-technology/realsense-depth-technologies.html>
- [4] Francois, E., Chuapeau, B., "Depth-based segmentation", IEEE Transactions On CSVT, vol. 7, no. 1, pp.237-240, Feb. 1997.
- [5] Jungong Han, Ling Shao, Dong Xu, Shotton, J., "Enhanced Computer Vision With Microsoft Kinect Sensor: A Review", IEEE Transactions on Cybernetics, vol. 43, no. 5, pp.1318-1334, Oct. 2013.
- [6] Suarez, J., Murphy, R.R., "Hand gesture recognition with depth images: A review", 2012 IEEE Conference on RO-MAIN, pp.411-417, Sep. 2012.
- [7] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, Andrew Fitzgibbon, "KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera", UIST'11, pp.559-568, Oct. 2011.
- [8] Yan Cui, Schuon, S., Chan, D., Thrun, S., Theobalt, C., "3D Shape Scanning with a Time-of-Flight Camera", 2010 IEEE Conference on CVPR, pp.1173-1180, Jun. 2010.
- [9] MPEG, <http://mpeg.chiaiglione.org>
- [10] VCEG, <http://www.itu.int/en/ITU-T/studygroups/com16/video/Pages/default.aspx>
- [11] JCT-VC, <http://www.itu.int/en/ITU-T/studygroups/com16/video/Pages/jctvc.aspx>
- [12] JCT-3V, <http://www.itu.int/en/ITU-T/studygroups/2013-2016/16/Pages/video/jct3v.aspx>
- [13] Sullivan, G.J., Ohm, J., Woo-Jin Han, Wiegand, T, "Overview of the High Efficiency Video Coding (HEVC) Standard", IEEE Transactions on CSVT, vol. 22, no. 12, pp.1649-1668, Dec. 2012
- [14] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, A. Luthra, "Overview of the H.264/AVC video coding standard", IEEE Transactions on CSVT, vol. 13, no. 7, pp.560-576, Jul. 2003.
- [15] JCT-3V of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "CE1.h: Backward View Synthesis Prediction using Neighbouring Blocks", JCT3V-C0152, Geneva, CH, Jan. 2013.
- [16] JCT-3V of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "CE3: Results on Depth-based Block Partitioning (DBBP)", JCT3V-G0106, San Jose, US, Jan. 2014.
- [17] Jorn Ostermann, Euee S. Jang, Jae-Seob Shin, Tsuhan Chen, "Coding of Arbitrarily Shaped Video Objects in MPEG-4", International Conference of Image Processing, pp.496-499, 1997.
- [18] Kaup, A., "Object-based texture coding of moving video in MPEG-4", IEEE Transactions on CSVT, vol. 9, no. 1, pp.5-15, 1999.
- [19] ISO/IEC 14496-2 (MPEG-4 Visual Version 1), "Coding of audio-visual objects-Part 2: Visual", Apr. 1999.
- [20] Rong Ding, Qionghai Dai, Wenli Xu, Dongdong Zhu, Hao Yin, "Background-frame based motion compensation for video compression", 2004 IEEE ICME, vol. 2, pp.1487-1490, 2014.
- [21] Ohm, J., Sullivan, G.J., Schwarz, H., Thio Keng Tan, Wiegand, T., "Comparison of the Coding Efficiency of Video Coding Standards-Including High Efficiency Video Coding (HEVC)", IEEE Transactions on CSVT, vol. 22, no. 12, pp.1669-1684, Dec. 2012.
- [22] Il-Koo Kim, Junghye Min, Tammy Lee, Woo-Jin Han, JeongHoon Park, "Block Partitioning Structure in the HEVC Standard", IEEE Transactions on CSVT, vol. 22, no. 12, pp.1697-2012, Dec. 2012
- [23] Woo-Jin Han, "Conditional Probability Based Early Termination of Recursive Coding Unit Structures in HEVC", Journal of Broadcast Engineering, vol. 17, no. 2, pp.354-362, Mar. 2012.
- [24] Chansub Park, Byeonggyu Kim, Dongsan Jeon, Sunheung Jung, Yeonhui Kim, Jinuk Suk, Jinsun Choi, "Early Termination Algorithm of Merge Mode Search for Fast High Efficiency Video Coding (HEVC) Encoder", Journal of Broadcast Engineering, vol. 18, no. 5, pp.691-701, Sep. 2012.
- [25] Hansu Lee, Kyeongyang Kim, Taeryong Kim, Gwanghoon Park, Hwiyoung Kim, Seongchang Lim, Jinho Lee, "Improving Encoder Complexity and Coding Method of the Split Information in HEVC", Journal of Broadcast Engineering, vol. 17, no. 2, pp.325-343, Mar. 2012.
- [26] JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "Early Termination of CU Encoding to Reduce HEVC Complexity", JCTVC-F045, Torino, IT, Jul. 2011.
- [27] JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "Coding tree pruning based CU early termination", JCTVC-F092, Torino, IT, Jul. 2011.
- [28] JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "Common HM test conditions and software reference configurations", JCTVC-I1100, Geneva, Switzerland, May. 2012.
- [29] Gary J. Sullivan and Thowas Wiegand, "Rate-Distortion Optimization for Video Compression", IEEE Signal Processing Magazine, vol. 15, no. 6, pp.74-90, Nov. 1998.
- [30] JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "Samsung Response to the Call for Proposals on Video Compression Technology", JCTVC-A124, Dresden, German, Apr. 2011.
- [31] JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "Description of video coding technology proposal by Fraunhofer HHI", JCTVC-A116, Dresden, German, Apr. 2011.
- [32] Zoran Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction", International Conference Pattern Recognition, vol. 2, pp.28-31, 2004.
- [33] M. Piccardi, "Background subtraction techniques: a review", IEEE International Conference on SMC, vol. 4, pp. 3099-3100, 2004.

- [34] Yongjo Ahn, Taejin Hwang, Sungeun Yoo, Woo-jin Han, Donggyu Sim, "Statistical Characteristics and Complexity Analysis of HEVC Encoder Software", Journal of Broadcast Engineering, vol. 17, no. 6, pp.1091-1105, Nov. 2012.
- [35] Jens-Rainer Ohm, G. J. Sullivan, Heiko Schwarz, T. K. Tan, T. Wiegand, "Comparison of the Coding Efficiency of Video Coding Standards-Including High Efficiency Video Coding (HEVC)", IEEE Transactions on CSVT, vol. 22, no. 12, pp.1669-1684, Dec. 2012.
- [36] Frank Bossen, Benjamin Bross, Karsten Suhring, David Flynn, "HEVC Complexity and Implementation Analysis", IEEE Transactions on CSVT, vol. 22, no. 12, pp.1685-1696, Dec. 2012.
- [37] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," ITU-T SG16 Q.6, VCEG-M33, Texas, USA, Apr. 2001.

---

### 저자 소개

---



#### 이 윤 진

- 1999년 2월 : 연세대학교(원주) 전산학과 학사
- 2001년 2월 : 연세대학교(원주) 전산학과 석사
- 2002년 6월 - 2007년 8월 : 넷앤티비 (주) 선임연구원
- 2001년 9월 - 현재 : 경희대학교 컴퓨터공학과 박사과정
- 주관심분야 : 비디오신호처리, 멀티미디어, 멀티미디어 시스템



#### 배 동 인

- 2011년 2월 : 경희대학교 컴퓨터공학과 학사
- 2013년 2월 : 경희대학교 컴퓨터공학과 석사
- 2013년 6월 ~ 현재 : LIG 넥스원 (주) 연구원 근무 중
- 주관심분야 : 비디오신호처리, 멀티미디어, 영상처리



#### 박 광 훈

- 1985년 2월 : 연세대학교 전자공학과 학사
- 1987년 7월 : 연세대학교 전자공학과 석사
- 1991년 1월 : Case Western Reserve University, Dept. of EEAP 석사
- 1995년 1월 : Case Western Reserve University, Dept. of EEAP 박사
- 1995년 3월 ~ 1997년 2월 : 현대전자 멀티미디어연구소 책임연구원
- 1997년 3월 ~ 2001년 2월 : 연세대학교(원주) 전산학과 부교수
- 2001년 3월 ~ 현재 : 경희대학교 컴퓨터공학과 교수
- 주관심분야 : 멀티미디어, 비디오 신호처리, 패턴인식, 영상처리, 계산지능