

특집논문 (Special Paper)

방송공학회논문지 제23권 제6호, 2018년 11월 (JBE Vol. 23, No. 6, November 2018)

<https://doi.org/10.5909/JBE.2018.23.6.747>

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

## 사용자 시점 기반 360 영상을 위한 렌더러 구현

장 동 민<sup>a)</sup>, 손 장 우<sup>a)</sup>, 정 종 범<sup>a)</sup>, 류 은 석<sup>a)†</sup>

### Implementing Renderer for Viewport Dependent 360 Video

Dongmin Jang<sup>a)</sup>, Jang-Woo Son<sup>a)</sup>, JongBeom Jeong<sup>a)</sup>, and Eun-Seok Ryu<sup>a)†</sup>

#### 요 약

본 논문에서는 실시간 고화질 360 영상 전송을 위해 사용자 시점에 기반한 타일 분할 기법을 적용하고 화질 평가를 위해 360 영상을 가상현실 기기 화면에 표현하는 구현을 설명한다. 사용자 시점에 기반한 고화질 360 영상 전송을 위한 방안으로, 움직임 참조 문제를 해결하기 위한 MCTS (Motion Constrained Tile Sets) 기술과 미리 구성된 타일 정보들을 포함하는 EIS (Extraction Information Sets) SEI (Supplemental Enhancement Information), 타일 정보를 추출해내고 영상을 분할 및 추출해주는 추출기(extractor)를 구현한다. 또한 사용자 시점에 기반한 타일 추출 방법과 추출된 영상을 이용해 가상현실 기기 화면에 표현하는 방법에 대한 구현 내용을 설명한다. 따라서, 제안된 구현물을 기반으로 영상 전송을 수행하면, 사용자 시점 영역의 영상만 전송하여 불필요한 영상 전송을 하지 않게 되어 화질 대비 낮은 대역폭의 향상된 영상을 표현할 수 있다.

#### Abstract

In this paper, we implement viewport dependent tile partitioning for high quality 360 video transmission and rendering method to present a HMD (Head Mounted Display) screen for 360 video quality evaluation. As a method for high-quality video transmission based on a user's viewport, this paper introduces MCTS (Motion Constrained Tile Sets) technique for solving the motion reference problem and EIS (Extraction Information Sets) SEI including pre-configured tile information, and extractor that extracts tiles. In addition, it explains tile extraction method based on user's viewport and implementation contents of the method of expressing on an HMD. Therefore, if 360 video is transferred by the proposed implementation which only transfers video from the user viewport area, it is possible to express higher quality video with lower bandwidth while avoiding unnecessary image transmission.

Keyword : 360 Video, Viewport, HEVC, MCTS, Renderer

a) 가천대학교 컴퓨터공학과(Department of Computer Engineering, Gachon University)

† Corresponding Author : 류은석(Eun-Seok Ryu)

E-mail: [esryu@gachon.ac.kr](mailto:esryu@gachon.ac.kr)

Tel: +82-31-750-8905

ORCID: <https://orcid.org/0000-0003-4894-6105>

※ 이 논문의 연구 결과 중 일부는 “한국방송·미디어공학회 2018년 하계학술대회”에서 발표한 바 있음.

※ 본 연구는 한국전력공사의 2016년 선정 기초연구개발과제 연구비에 의해 지원되었음(과제번호: R17XA05-68).

※ This research was supported by Korea Electric Power Corporation. (Grant number : R17XA05-68).

· Manuscript received September 4, 2018; Revised November 5, 2018; Accepted November 5, 2018.

## 1. 서론

최근 몰입형 실감 미디어인 360 영상의 이용이 급속도로 증가하고 있다. 이러한 360 영상을 높은 몰입감으로 보기 위해 최소한 UHD급의 고화질 영상이 요구된다<sup>[1]</sup>. UHD 동영상의 전송은 높은 대역폭을 요구하게 되고, 따라서 고해상도 360 영상의 효율적인 전송 기법이 요구되고 있는 실정이다.

MCTS는 영상의 각 타일을 공간적 및 시간적 독립성을 적용해 부호화하는 기법으로 각 타일을 독립적으로 전송할 수 있다<sup>[2]</sup>. 각 타일을 독립적으로 전송함으로써 고화질 360 영상에서 사용자 시점에 해당하는 타일만 추출하여 전송하고, 사용자는 뷰포트에 해당하는 타일들만 영상 부호화 및 구체 매핑, 화면에 렌더링할 수 있다. 이러한 타일링 기법을 이용한 적응형 360 영상 전송은 기존 전송 기법에 비해 화질 대비 적은 대역폭을 이용하고 영상을 부호화하는 속도가 향상되며, 화면에 렌더링하기 위해 이용되는 처리량이 낮아진다.

일반적으로 360 동영상 플레이어는 MCTS 기법이 적용되지 않고 표준화된 H.264/AVC 와 H.265/HEVC 두 가지 동영상 압축 기술을 주로 사용하여 고화질 360 영상 압축 처리를 하고 있다<sup>[3]</sup>. 적응형 스트리밍을 위해 MCTS 기법을 비롯한 타일링 기법을 이용한 동영상 압축 방법은 일반

적인 동영상을 부호화하는 과정과 다르게 타일을 추출하는 과정과 비트스트림을 선택하는 과정을 진행한다. 따라서, 타일 기반으로 부호화된 영상을 렌더링을 하기 위해서는 비트스트림을 부호화하는 과정에서 생기는 타일 정보와 해상도, 프레임에 관련된 추가적인 메타데이터를 구현물에서 처리할 수 있는 데이터로의 구성이 필요하다. 또한, 렌더러(renderer)는 영상과 함께 전송된 메타데이터를 이용해 특정 타일만 구체에 매핑하고 사용자 화면에 렌더링 가능해진다.

본 논문에서는 타일 기반으로 추출된 영상을 렌더링하기 위하여 MCTS와 EIS SEI 정보를 통해 추가적인 메타데이터를 구성하고 사용자 시점에 해당하는 타일을 추출하는 방법과 구체에 각 타일을 매핑하는 방법을 제안한다. 그리고 나서, 렌더러의 설계 및 구현에 대해 설명한다. 그림 1은 MCTS가 적용된 영상에서 추가적인 정보를 포함한 전송과 해당 타일만 추출한 영상을 부호화하는 과정, 타일 기반 렌더링 방식에 대한 제안을 보여준다.

본 논문의 구성은 다음과 같다. 2절에서는 관련 연구로 진행중인 프로젝트에 관한 내용과 사용자 시점 기반 비디오 스트리밍 연구를 설명한다. 3절에서는 MCTS와 EIS SEI, 렌더러를 위한 타일 기반 전송 이슈에 대해 설명하고 4절에서는 추출기의 구현과 사용자 시점에 해당하는 타일 추출 구현 렌더러 구현에 대한 내용을 토의한다. 5절에서는 360 영상 테스트 절차를 따른 실험 결과에 대한 설명을 진행한다. 마지막으로 6절에서는 본 논문에 대한 결론을 맺는다.

## II. 관련 연구

### 1. 360 비디오 스트리밍 시스템을 위한 지난 연구

본 논문은 대역폭 문제와 프로세스 성능에 제한이 있는 모바일 VR을 위한 고해상도 영상 전송에 관한 MVP (Merciless Video Processing) 프로젝트의 일부분이며 MVP 프로젝트는 여러 연구를 포함한다. 그림 2는 MVP 프로젝트의 개념적인 구조를 보여준다. MVP 프로젝트의 첫 번째로, HEVC(High Efficiency Video Coding)와 HEVC의 확

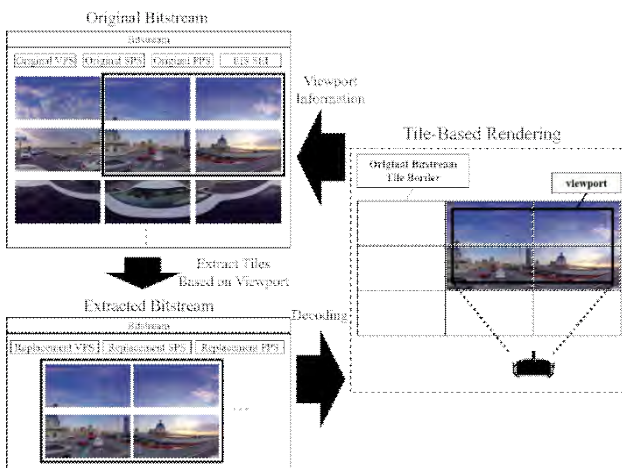


그림 1. 사용자 시점에 기반한 개념적인 타일 추출 구조  
Fig. 1. The conceptual architecture of tile extracting system based on user viewport

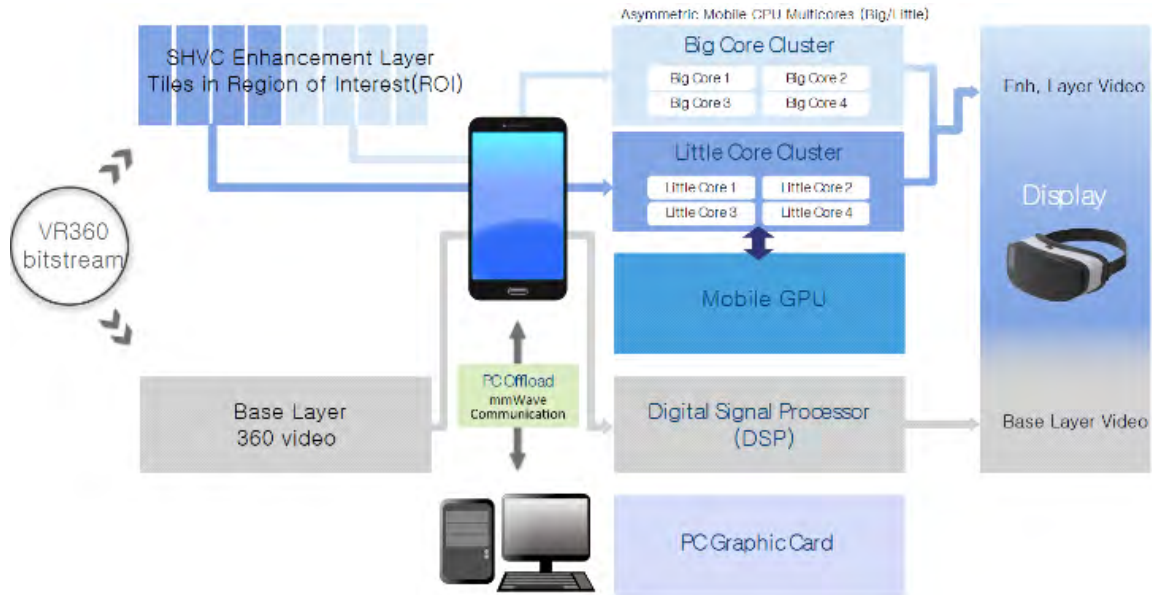


그림 2. MVP(Merciless Video Processing) 프로젝트 구조  
 Fig. 2. The conceptual architecture of merciless video processing project

장인 SHVC(Scalable HEVC)를 이용하여 압축한 영상 중 사용자의 ROI(Region Of Interest)에 해당하는 영역만 고휘상도로 전송하기 위해 해당하지 않는 영역은 기본 계층(base layer)로, 해당하는 영역은 부가적으로 확장 계층(enhancement layer)도 전송하는 연구가 있다. 두번째로, 사용자의 ROI에 해당하는 영역만 모바일의 비대칭(asymmetric) big/little 코어를 이용해 병렬 처리한다. 현재는 타일 별 해상도나 복잡도를 선형적으로 계산하여 가중치를 선정하고 big/little 코어에 각 프로세스를 스케줄링한다<sup>[4]</sup>. 세번째로, 프로세서 성능에 제한이 있는 모바일을 위해 복호화가 가능한 장비 또는 DSP(Digital Signal Processor)를 통해 복원된 전체 영상을 밀리미터파(mmWave)를 이용해 전송한다<sup>[5]</sup>. 본 논문에서 설명하는 구현물은 MVP 프로젝트 일부분으로 사용자 시점에 기반한 타일 분할 기법을 구현하는 방법과 360 영상의 일부 영역을 렌더링하는 결과, 화질 측정 결과를 보여준다.

## 2. 사용자 시점 기반 비디오 스트리밍 연구

뷰포트(viewport)는 사용자가 바라보는 영역으로 360 영상에서 화면에서 사용자가 바라보고 있는 영역은 전송되는

전체 영상 중 일부이다. 영상을 사용자의 장치에 렌더링할 시에 불필요한 영역이 존재한다. 이러한 부분을 제거하고 전송하면 대역폭은 줄어든다. 이런 문제를 해결하기 위해 전체화면을 여러 영역으로 나누어 압축하는 방법과 압축된 영상을 분할할 수 있도록 부가적인 정보를 주는 방법이 제안되었다. 하지만 전체화면을 여러 영역으로 나누어 압축하는 방법은 간단하게 구현할 수 있지만 전체 전송 시스템에서 번거로운 절차를 생성하고 분할된 영상에 대한 각각의 오버헤드와 움직임 보상이 적어져 많은 저장 공간이 요구된다. 또한, 영상을 분할하여 부호화하기 위해서는 다중 인코더를 필요하게 되며 마찬가지로 분할하여 부호화된 비트스트림(bitstream)을 부호화하기 위해서는 다중 디코더를 요구한다. 반면에 본 논문에서 제안하는 방법인 부호화된 비트스트림으로부터 분할된 영역을 추출하는 방법은 단일 인코더로부터 비트스트림을 생성할 수 있으며, 분할된 비트스트림을 생성하는 속도가 빠르고 타일 영역에 의존하여 원본 영상에서 다양한 영역으로 구성된 비트스트림을 생성할 수 있다. 원본 영상에서 다양한 영역으로 구성할 수 있으므로 사용자 시점 기반 스트리밍에서 기존에 비해 적은 수의 디코더로 영상을 복원할 수 있게 된다.

MPEG 표준화 단체는 하나의 영상으로부터 추가적인 영

상이나 작업없이 분할하기 위해 현재 화면의 움직임 벡터를 제한하는 MCTS 기법을 논의했다. MCTS 기법은 기존 HEVC에 병렬처리를 위해 구현 되어있는 타일을 이용해 공간적으로 움직임 참조 문제에 더해 복원되지 않은 타일 영역으로의 시간적 참조 문제를 해결하기 위한 방법이다. MCTS 기법은 인코더에서 시간적 움직임 정보를 제한하기 때문에 약간의 인코딩 효율 하락이 발생한다. 하지만, MCTS 기법이 적용된 비트스트림은 타일의 독립성을 보장하여 추출을 가능하게 한다. A. Zare 등은 HEVC 인코더를 MCTS 형태로 수정했다. 해당 연구는 3개의 타일 분할 기법을 사용하였으며, 3%~6%의 인코딩 효율 손실 결과를 갖는다<sup>[6]</sup>. 하지만 FOV(Field Of View)에 해당하는 타일을 전송할 경우 약 30%~40%의 비트레이트(bitrate)를 절약한다. 또한 R. Skupin 등은 MCTS를 이용해 HEVC 인코딩하여 원본 해상도(full resolution) 비트스트림과 가로 및 세로로 절반 축소된 해상도(half resolution)로 나누어 인코딩하여 사용자 시점 기반 렌더링을 진행하였으며, 최대 30% BD-rate의 코딩 효율 향상을 이루었다<sup>[7]</sup>. 본 논문에서는 MCTS를 적용한 HM을 이용해 뷰포트에 기반한 추출기를 구현하고 PSNR 수치를 측정한다. 또한, 뷰포트 기반 렌더링과 시나리오를 적용한 효율 측정에 대한 내용을 기술한다.

360 영상을 전송하기 위한 기법으로 MCTS를 이용한 타일 기법 외에 하나의 영상으로부터 추가적인 영상을 생성하여 부호화하는 타일링 기법에서 적응적 영상 스트리밍은 높은 비트레이트 감소율을 보여주었다. 하지만, 사용자의 머리 움직임으로 인한 고화질 전송이 이루어지는 타일이 전환되면서 생기는 지연이 존재하여 QoE(Quality of Experience)의 요구 사항에 만족하지 못하였다<sup>[8]</sup>. 따라서, 영상 전체의 평균적인 영상 화질의 향상과 적응적 영상 전송 환경에서 새로운 영상 패킷(packet) 요구(request)로 인한 완충(buffering) 기법의 향상, 전송부와 수신부에서의 비트레이트 조절(rate control), 머리 움직임 시 저화질 영역으로의 전환 횟수 감소와 저화질에서 고화질로의 지연 시간을 감소해야 한다. 위와 같은 문제들 중 저화질에서 고화질로의 지연 시간을 감소시키기 위한 해결 방안으로 머리 움직임을 예측하는 기술에 기반한 적응적인 뷰포트 영역 전송 기법이 존재한다. F. Qian 등은 이러한 기법의 첫번째 시도로 이전 사용자들의 피드백으로부터 도출된 관심 영역(존

재 확률)을 이용하여 뷰포트 영역에 해당하는 예측 전송을 시도하였으며, WLR(Weighted Linear Regression)을 이용하여 0.5초 후의 예측은  $96.6\% \pm 2.0\%$ , 1초 후의 예측은  $92.4\% \pm 3.7\%$ 의 예측 정확도를 보여주었다<sup>[9]</sup>. 계속해서 저화질에서 고화질로의 지연 시간을 감소시키기 위한 연구결과들이 발표되고 있으며, 본 논문의 추후 연구로 MCTS 기반의 관심 영역을 설정하고 머리 움직임 예측을 통한 타일 전환 시의 지연 시간 감소를 주제로 진행하고자 한다.

### III. 사용자 시점 기반 360 영상 추출

#### 1. MCTS를 이용한 타일 기반 부호화

기존의 HEVC와 SHVC에 정의된 타일은 병렬 처리를 지원하기 위해 공간적 예측 부호화를 각 타일별로 독립적 예측을 한다. 하지만, 각 타일들은 시간적 예측 부호화에 대한 독립성을 가지고 있지 않아, 이전 화면의 모든 타일들이 복호화 되어 있어야 현재 화면의 타일들을 복호화 할 수 있는 문제점이 있다. MCTS는 일시적으로 인코더에서 참조 프레임으로의 시간적 움직임 추정을 HEVC에서 병렬처리를 위해 지원하는 기능인 타일 내 공간으로 제한하여 시간적 및 공간적으로 독립적인 타일로 구성된 비트스트림 파일을 생성한 후, 타일과 타일을 구별하기 위해 NAL 단위의 슬라이스를 타일과 1:1로 병합한다<sup>[2]</sup>. 또한, 본 연구에서 제안된 소스는 HM 16.18에서 #if MCTS\_ENC\_CHECK 지시문을 통해 확인할 수 있다<sup>[10]</sup>.

##### 1.1. 움직임 추정 화소 간의 예측오차 거리 수정

SHM 및 HM에서 현재 타일이 동일한 위치의 타일을 시간적으로 참조할 때, 8탭 필터를 사용하여 휘도 성분에 대한 보간을 진행한다. 8탭 필터를 수평으로 적용될 때, 현재 픽셀의 좌측으로 3픽셀 및 우측으로 4 픽셀이 사용된다. 수직으로 적용될 때, 현재 픽셀의 위 측의 3픽셀과 아래 측의 4픽셀이 사용된다. 따라서, 시간적으로 동일한 위치의 타일에 위치한 PU를 참조할 때에도, 픽셀 간 보간이 되는 영역이 다른 타일을 참조하지 않도록 고려하여야 한다. 그림 3은 TIP(Temporal Inter Prediction)에서 동일한 위치의 타일

을 참조할 때, 문제가 나타나는 보간 영역을 보여준다. PU가 보간 문제가 일어나는 영역을 참조하면 PU가 다른 타일의 픽셀을 사용하여 보간하기 때문에, 타일을 독립적으로 전송할 수 없다. 따라서, 문제가 나타나는 보간 영역은 TIP가 참조할 수 있는 범위에서 제외되어야 한다.

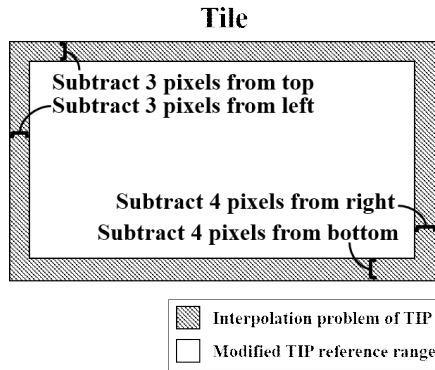


그림 3. TIP에서 동일한 위치로의 타일 참조로 인한 보간 영역 문제  
 Fig. 3. Interpolation area problem due to the tile reference to the same location in TIP

### 1.2. 타일 경계에서 AMVP와 Merge 후보 결정

SHM 및 HM은 개선된 움직임 예측 방식인 AMVP(Advanced Motion Vector Prediction) 및 Merge 모드를 사용하여 화면 간 예측에서 움직임 오차를 줄인다. 두 가지 방법 모두 공간적 및 시간적 후보를 찾아서 사용한다. 앞서 언급한대로 참조하는 후보 블록은 동일한 타일 내의 후보를 사용하도록 고려해야 한다. 기본적으로, 현재 PU의 우측 하단에 있는 블록이 현재 CTU(Coding Tree Unit)의 수평 경

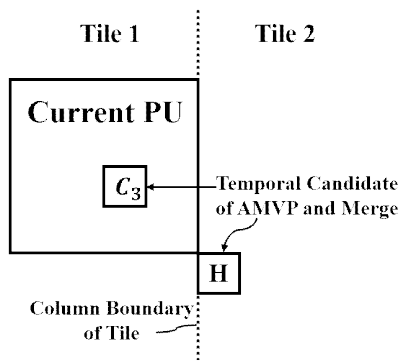


그림 4. AMVP와 Merge 모드의 타일 간 수직 경계에서 후보 결정 문제  
 Fig. 4. Problems determining candidate on vertical boundaries between tiles in AMVP and Merge mode

계를 넘어서면 후보로 사용되지 않는다. 그러나 참조하는 후보 블록이 수직 경계 밖으로 나갈 경우에는 문제가 생긴다. 그림 3과 같이 H 후보 블록이 타일의 수직 경계 밖으로 나갈 때, AMVP 및 Merge 방식에 의해 H 후보 블록이 선택되면 문제가 생길 수 있다. 따라서 타일 간 수직(column) 경계에서 H의 시간적 후보를 제외시킨다.

## 2. 렌더러를 위한 타일 추출 및 구조

### 2.1. EIS SEI 메시지

EIS SEI 메시지는 MCTS 기술이 적용된 압축 영상으로부터 사용자가 원하는 타일을 추출해낼 수 있도록 임의로 구성된 파라미터 셋을 제공한다<sup>[4]</sup>. 임의로 구성된 파라미터 셋은 비트스트림의 일부 정보를 대체할 수 있는 정보를 포함한다. 대체되는 정보 VPS의 레벨 및 티어, SPS의 화면 해상도, PPS의 타일 분할 정보이다. 또한, 하나의 EIS SEI 메시지는 약 2,000개의 EIS를 포함할 수 있다.

EIS SEI Message (NAL)	
Header	NAL_UNIT_PREFIX_SEI
RBPS	<ul style="list-style-type: none"> <li>the number of extraction information sets</li> <li>the number of MCTS sets</li> <li>the number of MCTSs in the MCTS set</li> <li>the MCTS index of the MCTS</li> <li>Slice reordering information</li> <li>the number of replacement VPS</li> <li>replacement VPS rbps data</li> <li>the number of replacement SPS</li> <li>replacement SPS rbps data</li> <li>the number of replacement PPS</li> <li>replacement PPS rbps data</li> </ul>

그림 5. EIS SEI Message의 구조  
 Fig. 5. Structure of EIS SEI Message

### 2.2 렌더러를 위한 추출기의 부가적인 정보 생성

EIS SEI 메시지를 이용한 시스템을 구성할 수 있는 미디어 전송 표준인 MPEG-DASH, MMT(MPEG Media Transport)와 같은 스트리밍 서비스에서는 모든 비트스트림을 패킷 단위로 구성이 되어있어 사용자 시점에 해당하는 타일들을 사용자가 요청하여 패킷 단위로 수신한다. 하지만, 그림 6과 같이 원본 비트스트림이 추출기에 의해 추출되어

대체된 VPS, SPS, PPS 정보로는 추출된 비트스트림이 원본 비트스트림의 몇 번째 타일인지와 몇 개의 타일로 구성되었는지 알 수 있는 정보가 없다. 따라서 실시간 영상 전송을 위해 시스템을 구성하는 경우에 수신자와 송신자가 동기화되지 않은 플랫폼 상에서는, 사용자 시점 정보를 보내더라도 타일 분할 기능을 포함한 송신자로부터 NAL로 이루어진 패킷 단위의 추출된 비트스트림이 현재 사용자 시점이 위치한 타일들과 동일한 위치의 타일들이 아닐 수 있다.

추출된 비트스트림을 360 영상으로 만들기 위해서는 추출기에서 사용자 시점 정보를 이용해 원본 비트스트림의 정보와 현재 추출된 비트스트림에 대한 추가적인 정보를 생성 및 전달해주어야 한다. 제안하는 연구에서는 360 영상을 가상현실기기에 렌더링하기 위해 추출기를 구현할 때, 원본 비트스트림의 해상도 정보, 원본 비트스트림의 총 타일 개수, 현재 추출된 비트스트림의 타일 개수, 추출된 비트스트림이 원본 비트스트림의 몇 번째 타일들로 구성되었는지에 대한 부가적인 정보들을 포함한다. 그리고 디코더와 렌더러의 정보를 유동적으로 사용할 수 있게 구성해 영상과 부가적인 정보들을 처리한다.

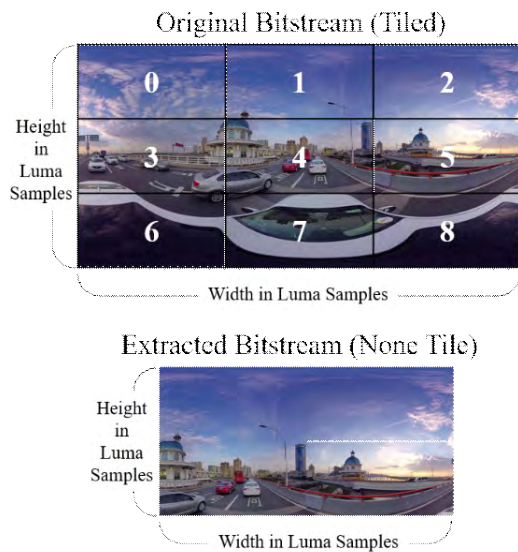


그림 6. 원본 및 추출된 비트스트림의 정보  
Fig. 6. Information of original and extracted bitstream

분할된 타일들은 각각 시간적 및 공간적 독립성을 가져

서 여러 HEVC 디코더를 사용한다면 문제가 없지만 하나의 HEVC 디코더에서 디코딩하기 위해서는 분할된 타일들은 뭉쳐진 사각형의 형태여야 한다. 따라서, 제안하는 연구에서 진행하는 방법으로 하나의 HEVC를 이용해 단일 프로세스로 디코딩과 렌더링을 진행하는 경우에는 시점에 위치하지 않은 타일을 포함할 수 있다. 또한, 사용자 시점을 고려해 가상현실기기 화면 상에 표현하기 위해 필요한 타일 구성은 최소 35개를 넘어간다. 따라서, 분할과 렌더링 시 단일 HEVC 디코더만 이용하는 것으로 고려하여 14개의 사각형 형태의 연결된 타일들로 구성하였다.

#### IV. 시스템 구현

##### 1. 타일 추출기 구현

타일 추출기는 원본 비트스트림과 EIS SEI 메시지를 파싱하고 VPS, SPS, PPS 일부 정보를 대체한다. 그림 7은 추출기의 기능 흐름도를 나타낸다. 입력 값은 target EIS id, MCTS set id, highest temporal id이다. 입력 값 설정 후 추출기는 NAL 단위로 해석한다. 기존 PPS를 해석할 때, 추출기는 화면단위로 NAL 유닛을 해석하기 위해 기존 화면을 구성하는 타일 개수를 구한다. 다음으로 추출기는 인코더에서 생성한 EIS SEI 메시지의 기존 파라미터 셋을 target EIS id 값에 맞는 대체 파라미터 셋으로 대체한다. 이후 추출기는 원하는 타일을 추출하기 위해 타일/슬라이스의 NAL 헤더를 분석하여 nuh\_temporal\_id\_plus1 - 1의 값이 target highest temporal id 값보다 작은 값을 선별한다. temporal id는 대역폭에 적응적인 대응을 할 수 있게 한다. 그 후 temporal id와 nuh\_layer\_id 조건을 만족하면 추출기는 target MCTS set id에 해당하는 타일/슬라이스를 추출하고, 추출한 타일/슬라이스 헤더의 first\_slice\_segment\_in\_pic\_flag와 slice\_segment\_address를 수정한다. 만약 타일/슬라이스가 화면의 첫번째 타일/슬라이스일 경우 first\_slice\_segment\_in\_pic\_flag 값을 1로 설정하고 그렇지 않으면 0으로 설정한다. 만약 EIS SEI 메시지의 slice\_reordering\_flag 값이 1인 경우, slice\_segment\_address 값을 EIS SEI 메시지의 output\_slice\_segment\_address로 설정한다. 그 외의 경우

에는 slice\_segment\_address 값을 래스터 주사 방식으로 설정한다. 추출기는 위의 방식으로 하면 단위로 세부 기능은 원본 비트스트림에서 타일 개수, EIS 및 MCTS sets, 대체 파라미터 셋을 파싱하고 VPS, SPS, PPS의 정보를 대체한다. SPS에서는 휘도 성분의 높이와 너비에 대한 정보를 타일의 높이와 너비로 대체하고 PPS에서는 타일 정보를 대체한다.

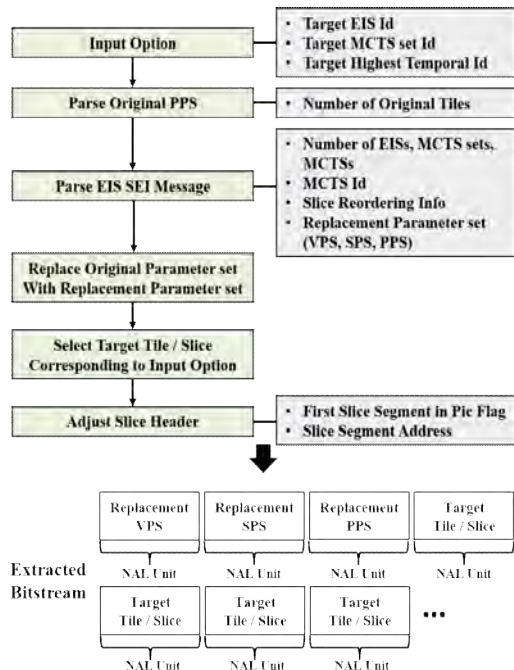


그림 7. 추출기의 기능 흐름도  
 Fig. 7. Flow chart of the extractor

## 2. 사용자 시점 기반 타일 추출

화질 측정과 렌더링을 고려하여 타일 결정 및 추출에 대한 구현을 진행한다. 그림 8은 사용자 시점에 해당하는 타일들을 추출하는 과정을 나타낸 블록도다. 가상현실 기기에 렌더링 되는 과정에서는 추적되는 정보는 사용자의 위치 정보와 사용자의 시점 정보이며, 사용자의 시점 정보는 yaw, pitch, roll로 표현할 수 있다. 반면에, 화질 측정을 위해 동적 뷰포트(dynamic viewport)에서 사용하는 정보는 yaw, pitch로 두 가지이다<sup>[11]</sup>. 따라서, 추출기에서 같은 좌표계를 이용하기 위해 가상현실 기기에서 추적되는 사용자 시점 정보 및 동적 뷰포트 정보를 구면좌표계 상의 yaw, pitch 값으로 변환한다. 그리고 나서 뷰포트에 해당하는 타일을 결정하기 위해 프레임마다 사용자 시점에 대한 정보를 생성하고 구체의 앵글(angle)을 이용한 구면 좌표계에서 3차원 직교 좌표계로 변환하여 뷰포트에 포함될 타일을 결정한다. 타일들은 서로 시간적 및 공간적으로 독립적이지만, 임의의 타일이 같은 위치에 있는 이전 프레임의 타일이 존재하지 않으면 참조 프레임이 없는 시간적 참조 문제가 발생한다. 임의 접근(RA: Random Access)이 가능한 서비스 상황을 고려하여 GOP(Group Of Picture)에서는 그림 8의 아래 그림과 같이 GOP에 포함되는 타일들이 I-프레임에서 포함이 되어있어야 한다. 하지만, 사용자 시점은 시간에 따라 변경이 되므로 현재 사용자 시점에 위치하는 타일이 DPB(Decoded Picture Buffer)에 존재하지 않을 수 있다. 따

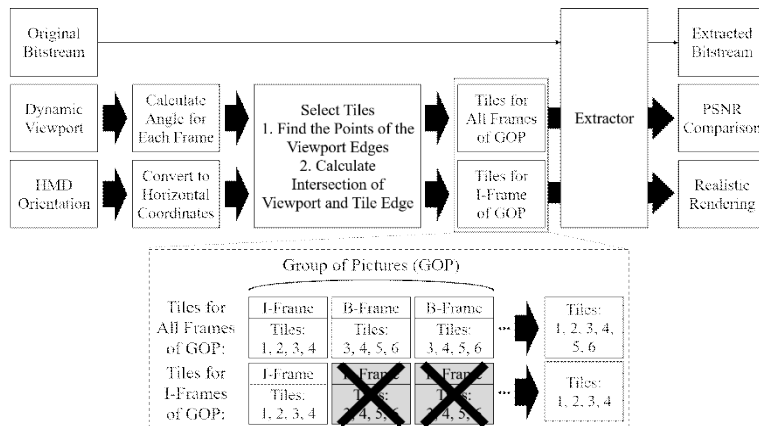


그림 8. 사용자 시점에 해당하는 타일들을 추출하는 블록도  
 Fig. 8. Block diagram of extracting tiles corresponding to the user viewport

라서, 실제 렌더링 시에 GOP 내의 모든 프레임의 타일들은 I-프레임에 위치한 타일들만 이용할 수 있다. 또한, I-프레임과 참조 프레임에 위치한 타일들만 참조 가능한 문제로 인해 머리 움직임 시에 타일 전환의 지연이 생긴다. 반면에, 화질 측정을 위해 미리 정의된 시나리오로 동적 뷰포트를 이용할 경우에는 미리 계산된 사용자 시점 정보를 이용하여 GOP 내에 포함되어야 할 타일들을 미리 알 수 있다. 본 논문에서는 화질 측정을 위해 GOP 내에 포함되어야 할 모든 타일을 포함하여 측정한다.

본 논문에서는 사용자의 시점에 해당하는 타일을 판단하기 위한 방법으로 뷰포트가 투영되는 3차원 상의 면과 타일 가장자리 점들 간 선분의 교차 여부를 이용하여 판단한다. 그림 9의 타일의 가장자리에 해당하는 두 점 간 선분을 직선  $s1$ 이라고 뷰포트의 가장자리에 위치한 평면을  $p1$ 이라

하면,  $p1$ 의 법선 벡터를 구해  $p1$ 과  $s1$  간 평면의 방정식과 직선의 방정식을 이용해 교차점을 구한다. 그 다음, 구체의 원점과 교차점으로서의 벡터를 구체면 상의 점으로 정규화한다. 평면의 방정식에 이용된 사용자 시점에 위치한 두 점을 각각  $v0$ 과  $v1$ , 구체면으로 정규화된 점을  $i0$ 이라 하고,  $v0$ 에서  $i0$ 까지의 거리와  $v1$ 에서  $i0$ 까지의 거리는  $v0$ 에서  $v1$ 까지의 거리보다 작다면 교차점은  $v0, v1$ , 원점으로 이루어진 삼각형 안에 위치한다. 따라서 해당 선분이 어떤 타일의 경계인지 구하여 추출해야 할 타일들을 선택할 수 있다. 그림 10은 실제로 사용자 시점에 기반한 타일 결정 결과를 보여준다. 사각형들은 뷰포트 영역을 표시하며 선분들은 타일 가장자리 점들 간 선분을 표시하고 각 점들은 교차점을 나타낸다.

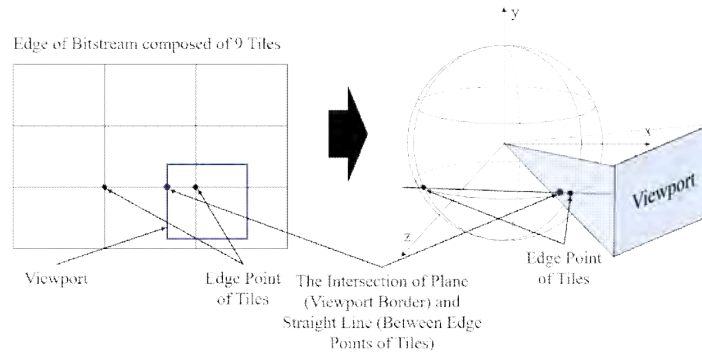


그림 9. 사용자 시점의 가장자리 면과 타일 경계 부분 간 교차하는 점  
Fig. 9. The intersection of the viewport edge plane and the tile edge line

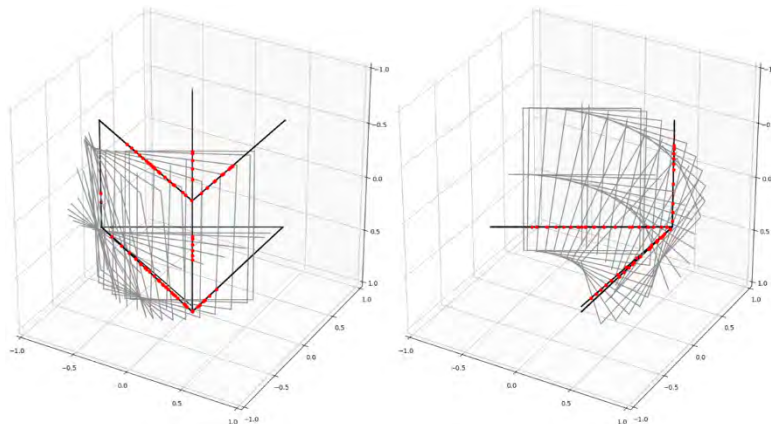


그림 10. 사용자 시점 기반 타일 결정의 예  
Fig. 10. Example of tile decision based on viewport



### 3. 렌더러 구현

본 연구에서는 HM 16.16과 Oculus Rift SDK, OpenGL을 사용하여 구현을 진행하였으며, MCTS 구현과 EIS SEI 구현은 [12,13]의 구현물을 사용하였다. 또한, 셰이더(shader)를 사용하지 않고 OpenGL 고급 라이브러리인 GLU 라이브러리를 사용해 구체를 생성하며, ERP 360 영상을 구체에 매핑하는 방식을 사용한다<sup>[14,15]</sup>. 구체는 곡면 개체를 선언하여 구체 모델인 gluSphere를 이용해 곡면을 그린다. 곡면을 이용하여 생성된 구체 객체를 곡면을 2차원으로 펼친 상의 x축으로 40, y축으로 40 분할하여 곡면이 렌더링 되었을 때 생기는 왜곡 현상을 줄인다. 정의되고 생성된 텍스처에 복호화된 ERP 360 영상 한 프레임의 YUV 420 픽셀 값을 RGB 픽셀 값으로 변환하여 텍스처를 업데이트한다.

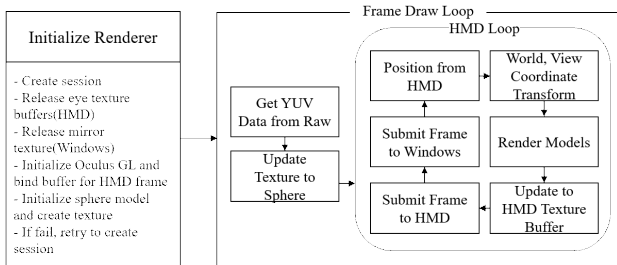


그림 11. 렌더러 개념도  
 Fig. 11. The conceptual architecture of renderer

그림 11과 같이 세션 생성, HMD 양안에 쓰이는 버퍼들, 모니터 디스플레이에 나타날 미리 버퍼들, Oculus SDK에서 재정의된 OpenGL 구체 모델의 생성과 텍스처 버퍼 생

성과 텍스처 바인딩을 한다. 그 후, 프레임을 그리는 함수 루틴을 실행한다. 프레임을 그리는 루틴은 프레임을 불러와 업데이트 하는 부분과 Oculus SDK가 지원하는 HMD 프레임을 그리는 루틴 부분으로 나눌 수 있다. Oculus Rift는 약 90Hz 정도로 디스플레이가 주사하고 동영상 프레임은 그보다 낮은 주기로 재생되기 때문에, HMD 프레임 그리는 루틴과 따로 일정 주기마다 영상을 읽어 들이는 함수를 구현한다.

타일 기반 텍스처 매핑은 원본 비트스트림의 높이와 너비 값을 이용해 텍스처를 생성한다. 그리고 각 프레임마다 9개의 타일로부터 분할 및 구성된 타일 높이와 너비 값과 타일 번호를 이용한다. 그림 12의 오른쪽 그림과 같이 현재 사용자 시점 영역의 타일만 텍스처 버퍼 영역에 그리고 생성된 텍스처를 구체 객체에 업데이트한다. 그림 13은 모든 과정을 진행하여 Oculus Rift에서 렌더링한 화면을 나타낸다.



그림 13. 렌더링 화면 (Trolley\_8192x4096)  
 Fig. 13. Rendered display (Trolley\_8192x4096)

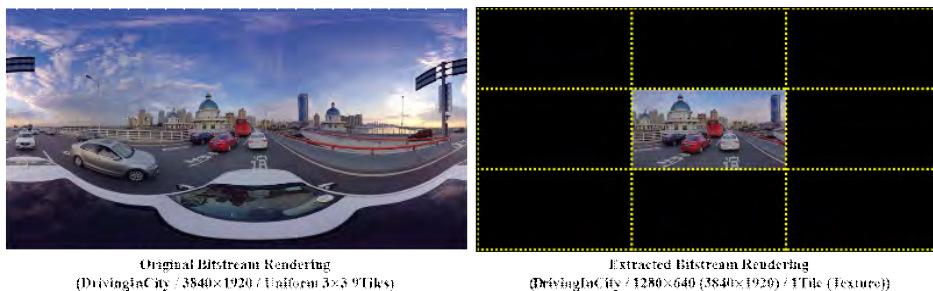


그림 12. 원본 비트스트림과 추출된 비트스트림  
 Fig. 12. Original bitstream and extracted bitstream

### V. 실험 결과

사용자 시점에 기반한 화질 측정을 위해 본 연구는 JVET(Joint Video Exploration Team)에서 선정된 테스트 시퀀스를 사용하였다<sup>[6]</sup>. 해당 시퀀스들은 표 1과 같이 8192x4096 해상도, 총 프레임은 300프레임, 초당 프레임은 30 fps, 색상 깊이는 10bit이며 RA(Random Access) 및 QP 27로 인코딩 되었다. FoV(Field of View)가 대각 시야각으로 90 ~ 110도임을 고려하여 타일을 균등하게 3x3으로 9등분 하였다. 따라서 최소 1개부터 최대 7개의 타일까지 사용자 시점 내에 포함할 수 있다.

표 1. 테스트 시퀀스 정보  
Table 1. Information of test sequences

Name	Resolution	Frame length	Frame rate	Bit depth
ChairliftRide	8192x4096	300	30 fps	10-bit
Gaslamp	8192x4096	300	30 fps	10-bit
Harbor	8192x4096	300	30 fps	10-bit
KiteFlite	8192x4096	300	30 fps	10-bit
SkateboardInLot	8192x4096	300	30 fps	10-bit
Trolley	8192x4096	300	30 fps	10-bit

본 연구에서는 JVET의 참조 소프트웨어인 360Lib를 이용하였으며, [17]에서 제안된 360 영상 테스트 절차를 수행하였고 그림 14는 본 연구에서 진행한 테스트 절차를 보여준다. 8192x4096 해상도의 ERP(Equirectangular Projection) 테스트 시퀀스를 360Lib에 포함된 포맷 변환을 통해

4096x2048 해상도의 다운 샘플링한 ERP 시퀀스를 생성한다. 다운 샘플링된 ERP 시퀀스는 MCTS가 적용된 HM 16.16 버전에서 Main 10으로 인코딩된다. 동적 뷰포트(dynamic viewport)에 따라 프레임마다 추출할 타일들을 선정하고 GOP 내에 포함된 모든 타일 위치들을 고려하여 GOP 단위의 타일들을 측정한다. 추출한 영상 및 추출되지 않은 영상을 통해 사용자 시점에 기반한 PSNR 측정과 포맷 변환을 통해 8192x4096 해상도의 복원된 ERP 영상을 End-to-End 로 PSNR 측정한다.

표 2는 End-to-End로 MCTS가 적용된 비트스트림 및 적용되지 않은 비트스트림의 WS-PSNR과 S-PSNR-NN 측정값을 보여준다<sup>[17,18]</sup>. ChairliftRide 시퀀스와 같이 PSNR 대비 비트레이트가 크게 차이가 나는 경우가 있지만, Trolley 시퀀스와 같이 PSNR 대비 비트레이트가 크게 차이가 나지 않는 결과를 보여준다. 표 3은 사용자의 관심영역을 고려하여 실험을 위해 임의로 설정한 시나리오 1과 시나리오 2에 해당하는 동적 뷰포트의 yaw pitch에 대한 앵글의 시작과 끝을 보여준다. FoV의 수직 시야각과 수평 시야각은 각각 75°로 설정하여 실험을 진행하였으며, 표 4는 임의로 설정한 시나리오에 따른 평균적인 비트레이트 절약율과 평균적인 프레임 당 타일 개수를 보여준다. 평균적으로 4-5개의 타일들이 사용자 시점에 포함되었으며, 51.8%의 비트레이트를 줄였다. 표 5는 MCTS가 적용된 비트스트림 및 적용되지 않은 비트스트림의 시나리오에 따른 동적 뷰포트에 렌더링된 영상의 PSNR 측정 값을 보여주며 큰 차이가 없는 것을 보여준다.

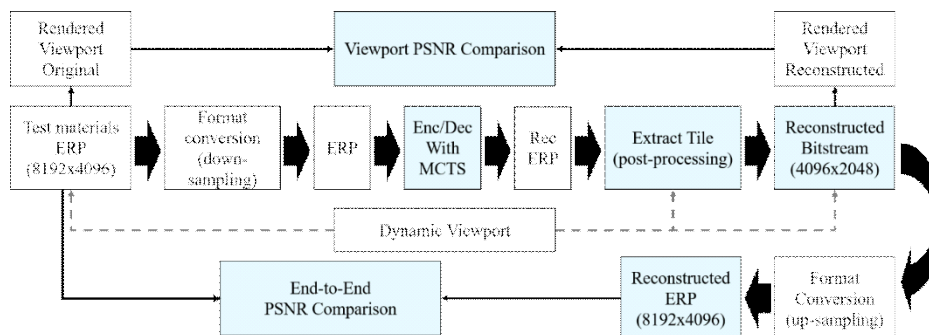


그림 14. 360Lib를 이용한 360 영상 테스트 절차  
Fig. 14. 360 video testing procedure using 360Lib

표 2. MCTS가 적용되거나 적용되지않은 영상의 End-to-End PSNR

Table 2. End-to-End PSNR with MCTS applied bitstream and non-applied bitstream

Name	MCTS Bit-rate Increasing	End-to-End Y-WS-PSNR (Non-MCTS)	End-to-End Y-SPSNR-NN (Non-MCTS)	End-to-End Y-WS-PSNR (MCTS)	End-to-End Y-SPSNR-NN (MCTS)
<i>ChairliftRide</i>	8.2%	39.7 dB	39.7 dB	39.6 dB	39.6 dB
<i>Gaslamp</i>	1.6%	41.9 dB	42.0 dB	41.9 dB	42.0 dB
<i>Harbor</i>	1.5%	41.1 dB	41.1 dB	41.1 dB	41.1 dB
<i>KiteFlite</i>	1.3%	38.4 dB	38.4 dB	38.4 dB	38.4 dB
<i>SkateboardInLot</i>	2.9%	38.8 dB	38.8 dB	38.8 dB	38.8 dB
<i>Trolley</i>	0.7%	38.7 dB	38.7 dB	38.7 dB	38.8 dB

표 3. 사용자의 관심영역을 고려한 dynamic viewport의 yaw and pitch

Table 3. Dynamic viewport's yaw, pitch considering user's region of interest

Name	Field of View	Scenario 1 Angle of Start	Scenario 1. Angle of End	Scenario 2. Angle of Start	Scenario 2. Angle of End
<i>ChairliftRide</i>	75° x 75°	-45, -15	45, 15	-201, -72	-111, -43
<i>Gaslamp</i>	75° x 75°	-51, -11	39, 19	61, -15	151, 15
<i>Harbor</i>	75° x 75°	-83, -56	7, -26	-140, 12	-50, 42
<i>KiteFlite</i>	75° x 75°	-114, -53	-24, -23	13, -41	103, -11
<i>SkateboardInLot</i>	75° x 75°	-180, 35	-270, 5	-65, -5	25, 25
<i>Trolley</i>	75° x 75°	210, -18	300, 12	30, -44	120, -14

표 4. 시나리오에 따른 평균적인 bit-rate 절약율과 프레임 당 타일 개수

Table 4. Average bit-rate savings and number of tiles per frame in scenarios

Name	Scenario 1 Number of Tiles (Average)	Scenario 1. Bit-rate Saving (Average)	Scenario 2. Number of Tiles (Average)	Scenario 2. Bit-rate Saving (Average)
<i>ChairliftRide</i>	4.6	-60.5%	5.1	-33.6%
<i>Gaslamp</i>	4.6	-30.9%	4.9	-44.6%
<i>Harbor</i>	4.6	-34.7%	3.8	-73.6%
<i>KiteFlite</i>	4.4	-63.4%	4.2	-60.8%
<i>SkateboardInLot</i>	4.5	-78.9%	4.7	-68.9%
<i>Trolley</i>	4.7	-40.7%	3.9	-31.0%

표 5. MCTS가 적용된 비트스트림 및 적용되지 않은 비트스트림의 Dynamic Viewport PSNR

Table 5. Dynamic Viewport PSNR with MCTS applied bitstream and non-applied bitstream

Name	Scenario 1. Y-PSNR on Viewport (None MCTS)	Scenario 1. Y-PSNR on Viewport (MCTS)	Scenario 2. Y-PSNR on-Viewport (None MCTS)	Scenario 2. Y-PSNR on Viewport (MCTS)
<i>ChairliftRide</i>	40.7 dB	40.5 dB	39.2 dB	39.1 dB
<i>Gaslamp</i>	40.8 dB	40.8 dB	42.0 dB	42.0 dB
<i>Harbor</i>	39.5 dB	39.5 dB	41.3 dB	41.3 dB
<i>KiteFlite</i>	40.0 dB	40.0 dB	38.7 dB	38.7 dB
<i>SkateboardInLot</i>	39.1 dB	39.1 dB	39.1 dB	39.0 dB
<i>Trolley</i>	37.9 dB	37.9 dB	37.0 dB	37.0 dB

## VI. 결 론

본 논문에서는 고화질 360 영상 전송 시 대역폭을 낮추기 위해 사용자의 시점 정보를 이용해 타일 기반 영상 분할 및 추출하는 방법을 제안하고 렌더링한 결과물을 구현하였다. 제안하는 방법은 추출된 비트스트림에 대한 타일 정보를 추가적으로 포함하고 머리 움직임의 방향 정보를 이용해 사용자 시점 내의 타일들을 결정한다. 또한, 실제로 분할된 영상에 대한 화질 측정과 Oculus rift 및 OpenGL을 이용하여 가상현실기기 화면에 표현하는 렌더러를 구현하였다. 결과적으로, 평균 9개의 타일 중 4.51개의 타일이 포함되는 것을 확인하였으며 평균 51.8%의 비트레이트 감소를 확인하였다. 따라서, 사용자 시점에 기반한 타일 분할 전송 기법을 이용하면 고화질 360 영상 전송 서비스에서 화질 대비 낮은 대역폭의 전송 효과를 얻을 수 있다. 또한, 본 논문에서 설명하고 인용한 MCTS 기술은 현재 표준화되고 있는 MPEG-I 3DoF와 6DoF 기술에서 영상 일부를 디코딩하는 방법이나 뷰합성 시 일부를 이용하는 방법 등으로 속도 개선에도 적용이 가능하다. 향후 연구에서는 활발히 진행되는 실감 미디어 표준에 적용하는 연구 또는 비대칭적 또는 다양한 조합의 타일 분할 기법을 적용할 계획이다.

## 참 고 문 헌 (References)

- [1] M. Champel, T. Stockhammer, T. Fautier, E. Thomas, and R. Koenen, "Quality Requirements for VR," 116th MPEG meeting of ISO/IEC JTC1/SC29/WG11, MPEG 116/m39532, 2016.
- [2] R. Skupin, Y. Sanchez, K. Suehring, T. Schierl, E. Ryu, and J. Son, "Temporal MCTS Coding Constraints Implementation," 120th MPEG meeting of ISO/IEC JTC1/SC29/WG11, MPEG 120/m41626, 2017.
- [3] J. Boyce, A. Ramasubramanian, R. Skupin, G. J. Sullivan, A. Tourapis, and Y. Wang, "HEVC Additional Supplemental Enhancement Information (Draft 4)," 29th JCT-VC meeting of ISO/IEC JTC1/SC29/WG11, JCTVC-AC1005, 2017.
- [4] Y. Ryu, and E. S. Ryu, "Video on Mobile CPU: UHD Video Parallel Decoding for Asymmetric Multicores," Proceedings of the 8th ACM International Conference on Multimedia System (MMSys 2017), Taipei, Taiwan, pp. 229-232, 2017.
- [5] T. T. Le, D. N. Van, and E. S. Ryu, "Real-time 360-Degree Video Streaming over Millimeter Wave Communication," International Conference on Information Networking 2018 (ICOIN2018), Chiang Mai, Thailand, pp. 5-17, 2018.
- [6] A. Zare, A. Aminlou, M. Hannuksela, and M. Gabbouj, "HEVC-compliant Tile-based Streaming of Panoramic Video for Virtual Reality Applications," In Proceedings of the 24th ACM Multimedia Conference, Amsterdam, Netherlands, pp. 601-605, 2016.
- [7] R. Skupin, Y. Sanchez, D. Podborski, C. Hellge, and T. Schierl, "Viewport-dependent 360 degree video streaming based on the emerging Omnidirectional Media Format (OMAF) standard," Image Processing (ICIP), 2017 IEEE International Conference on. IEEE, Beijing, China, pp. 4592-4592, 2017.
- [8] K. Brunström, S. A. Beker, K. De Moor, A. Doods, S. Egger, M.-N. Garcia, T. Hossfeld, S. Jumisko-Pyykkö, C. Keimel, M.-C. Larabi et al. "Qualinet white paper on definitions of quality of experience," <https://hal.archives-ouvertes.fr/hal-00977812> (accessed Sep. 03, 2018).
- [9] F. Qian, L. Ji, B. Han, and V. Gopalakrishnan, "Optimizing 360 video delivery over cellular networks," Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges. ACM, New York, USA, pp. 1-6, 2016.
- [10] High Efficiency Video Coding (HEVC), <https://hevc.hhi.fraunhofer.de/> (accessed Sep. 03, 2018).
- [11] T. Ikai, Y. Yasugi, and T. Aono, "AHG8: Dynamic viewport generation for 360° video evaluation," 5th JVET Meeting of ITU-T SG 16 WP 3 and ISO/IEC JTC1/SC29/WG11, JVET-E0024, 2016.
- [12] J. Son, D. Jang, and E. S. Ryu, "Implementing Motion-Constrained Tile and Viewport Extraction for VR Streaming," ACM Network and Operating System Support for Digital Audio and Video 2018 (NOSSDAV2018), Amsterdam, Netherlands, pp. 61-66, 2018.
- [13] J. Son, D. Jang, J. Chung, E. S. Ryu, "Implementing 360 Video Tiled Streaming System," ACM Multimedia Systems 2018 (ACM MMSys 2018), Amsterdam, Netherlands, pp. 521-524, 2018.
- [14] L. Bassbous, S. Steglich, and M. Lasak, "High quality 360° video rendering and streaming," NEM Summit proceedings, Porto, Portugal, 2016.
- [15] Oculus rift, <https://www.oculus.com/rift> (accessed Sep. 03, 2018).
- [16] P. Hanhart, J. Boyce, and K. Choi, "JVET common test conditions and evaluation procedures for 360° video," 11th JVET Meeting of ITU-T SG 16 WP 3 and ISO/IEC JTC1/SC29/WG11, JVET-K1012-v1, 2018.
- [17] Y. Ye, and J. Boyce, "Algorithm descriptions of projection format conversion and video quality metrics in 360Lib Version 7," 11th JVET Meeting of ITU-T SG 16 WP 3 and ISO/IEC JTC1/SC29/WG11, JVET-K1004, 2018.
- [18] K. K. Sreedhar, A. Aminlou, and M. M. Hannuksela, "Viewport-adaptive encoding and streaming of 360-degree video for virtual reality applications," IEEE International Symposium Multimedia (ISM) 2016 on. IEEE, San Jose, USA, pp. 583-586, 2016.

---

저 자 소 개

---

**장 동 민**



- 2013년 3월 ~ 현재 : 가천대학교 컴퓨터공학과 학사과정
- ORCID : <https://orcid.org/0000-0001-7920-8637>
- 주관심분야: 멀티미디어 통신 및 시스템, HMD/VR 응용분야, 적응형 스트리밍

**손 장 우**



- 2017년 2월 : 가천대학교 컴퓨터공학과 학사
- 2018년 8월 : 가천대학교 컴퓨터공학과 석사
- 2018년 9월 ~ 현재 : Fraunhofer HHI 방문연구원
- ORCID : <https://orcid.org/0000-0002-8781-1564>
- 주관심분야: 멀티미디어 통신 및 시스템, 적응형 스트리밍, 영상 압축

**정 종 범**



- 2018년 8월 : 가천대학교 컴퓨터공학과 학사
- 2018년 9월 ~ 현재 : 가천대학교 컴퓨터공학과 석사
- ORCID : <https://orcid.org/0000-0002-7356-5753>
- 주관심분야: 멀티미디어 통신 및 시스템, 다중 영상 합성, 영상 압축

**류 은 석**



- 1999년 2월 : 고려대학교 컴퓨터학과 학사
- 2001년 2월 : 고려대학교 컴퓨터학과 석사
- 2008년 2월 : 고려대학교 컴퓨터학과 박사
- 2008년 3월 ~ 2008년 8월 : 고려대학교 연구교수
- 2008년 9월 ~ 2010년 12월 : 조지아공대 박사후과정
- 2011년 1월 ~ 2014년 2월 : 인터디지털 연구소 Staff Engineer
- 2014년 3월 ~ 2015년 2월 : 삼성전자 수석연구원/파트장
- 2015년 3월 ~ 현재 : 가천대학교 컴퓨터공학과 조교수
- ORCID : <https://orcid.org/0000-0003-4894-6105>
- 주관심분야: 멀티미디어 통신 및 시스템, 비디오 코딩 및 국제 표준, HMD/VR 응용분야