KRNet2019

3DoF+ 360-Degree Video Streaming System

Jun. 25. 2019

Eun-Seok Ryu (류은석: esryu@gachon.ac.kr)
Multimedia Communications and Systems Lab (MCSL)
http://mcsl.gachon.ac.kr
Department of Computer Engineering
Gachon University
Immersive Media - 360 Virtual Reality

• 360° as part of “10 Breakthrough Technologies (MWC)”

• Inexpensive cameras that make spherical images are changing the way people to share stories (Mobile World Congress).

• Next-generation real-world media to overcome space and time constraints and provide viewers with a natural, 360-degree, fully immersive experience.

* Source from Prof. Jewon Kang
Immersive Media Standard

• **Step-by-step objective of ISO/IEC MPEG Immersive Video**
  • MPEG-I is responsible for standardizing immersive media in MPEG and specifies the goals of Step 3.
  • Goal of Revitalizing VR Commercial Service by 2020
  • Goal of 6DoF media support by 2022 after completing 3DoF standard by 2017

---

**Step 1**
- Complete 3DoF standard by 2017
- Rotate head in a fixed state
- 360 video full streaming by default
- Tiled streaming if possible

**Step 2**
- Enable VR commercial servicers by 2020
- Allow head rotation and movement within a restricted area
- User-to-user conversations and projection optimization

**Step 3**
- Support 6DoF by 2022
- 6DoF video will reflect user’s walking motion
- Support interaction with virtual environments
Immersive Media Standard Roadmap

- **Beyond Media**
  - Immersive Media Standards
    - Systems and Tools
      - VR 360
        - PCC Systems Support
        - Immersive Media Scene Description Interface
        - OMAF v.2
        - CMAF v.2
        - Partial File Format
        - Web Resource Tracks
        - Internet of Media Things
      - Media Orchestration
      - Network-Based Media Processing
      - Multi-Image Application Format
      - CMAF v.2
      - Partial File Format
      - Web Resource Tracks
      - Internet of Media Things
  - Beyond Media
    - Media Coding
      - Versatile Video Coding
      - 3DoF+ Video
      - Video Point Cloud Compression
      - Geometry Point Cloud Compression
      - PCC Systems Support
      - Immersive Media Scene Description Interface
      - OMAF v.2
      - CMAF v.2
      - Partial File Format
      - Web Resource Tracks
      - Internet of Media Things
    - Descriptors for Video Analysis (CDVA)
    - Color Support in Open Font Format
    - Essential Video Coding
    - Low Complexity Enhancement Video Coding
    - Neural Network Compression for Multimedia
  - Genome Compression Extensions
  - Genome Compression

Immersive Media Standard Projects

New MPEG project: ISO/IEC 23090

*Coded Representation of Immersive Media*

8 parts:
1. Architectures for Immersive Media (Technical Report)
2. Omnidirectional Media AF
3. Versatile Video Coding
4. 6 Degrees of Freedom Audio
5. Video Point Cloud Coding (V-PCC)
6. Metrics for Immersive Services and Applications
7. Metadata for Immersive Services and Applications
8. Network-Based Media Processing
9. Geometry Point Cloud Coding (G-PCC)
Immersive Media Standard

• MPEG & 5G
  • Viewport-Adaptive Streaming
    • Benefits greatly from fast edge response times
    • OMAF v.2

• Immersive Media Access and Delivery
  • Finer granularity access for huge media data volumes
  • Partial retrieval of partially visible and audible scenes
  • Fast network responses for immersive and interactive media

• Partial (“split”) or Complete Edge Rendering
  • Do heavy lifting in network, save on processing in mobile devices
  • Allowing very sophisticated media to be consumed with reasonable device power use
  • Mixing immersive feeds for real-time communication

• Network-Based Media Processing
  • Using the network and the edge to support media processing
  • Network-based, last-second media personalization
3DoF (Degree of Freedom) Viewport-Adaptive Streaming System
Requirements of 360 Video Streaming

- **High Bandwidth Requirement of VR**
  - Recently released various HMDs as 360 Degree Video Viewer
  - Require 12K resolution for High quality VR
  - The current VR don’t fully support High quality VR resolution

<table>
<thead>
<tr>
<th>Requirement</th>
<th>details</th>
</tr>
</thead>
<tbody>
<tr>
<td>pixels/degree</td>
<td>40 pix/deg</td>
</tr>
<tr>
<td>video resolution</td>
<td>11520x6480 (12K)</td>
</tr>
<tr>
<td>framerate</td>
<td>90 fps</td>
</tr>
</tbody>
</table>

The emergence of various HMD
(Gear VR, Oculus Rift, Daydream, PlayStation VR)

Requirements for high quality VR
Source: Technicolor, Oct. 2016 (m39532, MPEG 116th Meeting)
Background

- **Field of View (FOV)**
  - The field of view (FOV) in the HMDs: 96° to 110°
  - The user sees only a portion of the 360° picture
    - The user’s current viewport: high resolution
    - Remaining parts: low-resolution

- **Tiles**
  - New Parallelization Tools in HEVC and SHVC Standard
  - Divided into rectangular regions
  - Flexible horizontal and vertical boundaries
    - Spatially refers to only its own tile, but temporally refers to other tiles
    - Temporal motion prediction problems arises when transmitting only some tiles

Field of View (FOV)

A frame divided into 8ix Tiles
Background - cont’d

• **Viewport Independent**
  • Transmit whole picture with pre-processing
  • Projection and packing
  • Downsampling / adjusting QP

• **Viewport-dependent (Adaptive)**
  • Transmit viewport only
  • Bitrate saving
  • But, delay

  • Consider several cases to extract areas, that shows poor encoding efficiency
  • Experience greater visual quality with lower bandwidth consumption*

---

Background - cont’d

• Saliency Map Prediction
  • Eye tracking in Head-mounted display
  • Data-driven approach
  • CNN model from eye fixation history

Bitrate Savings
Rate adaptation algorithm:
17% (Low motion) to 20% (High motion)

CASM algorithm:
18% (Low motion) and 23% (High motion)
Tile-based Viewport Adaptive Video Streaming

- Viewport Adaptive Streaming
  - Motion Constrained Tile Sets (MCTS) refer to the encoder for time and space movement reference for independent tile transfer within the current location tile
  - Extract and composite specific tiles from the bitstream with MCTS to form an adaptive environment at the time of the user
  - Reduce bandwidth when sending only tiles that correspond to a user’s area of interest

Ensure independence between tiles
Considerations of SHVC and HEVC for MCTS

HEVC Encoder

- EL: Enhancement Layer
- BL: Base Layer
- TIP: Temporal Inter Prediction
- ILP: Inter Layer Prediction

Reference blocks in the tile at same position:
Temporal Inter Prediction

Reference blocks in other tiles:
Intra Prediction

SHVC Encoder

Reference blocks in the tile at same position:
Temporal Inter Prediction

Reference blocks in other tiles:
ILP with upsampled BL

Upsampled PicBL t

Considerations: Interpolation, Temporal Candidate of AMVP and MERGE
**MCTS Considerations – Interpolation**

- **Modify Reference Range of Motion Vectors for MCTS**
  - Use interpolation to obtain more accurate motion vectors as the TIP
  - HEVC and SHVC use an eight-tap filter to interpolate luma prediction
  - Interpolation filters Use 3 pixels of left and top and 4 pixels of right and bottom
  - Modify reference range to not use pixels of other tiles as interpolation

![Diagram of current pixel and pixel used for interpolation](image)

The current pixel and the pixel used for interpolation

Interpolation problem of referring to a tile at the same position in TIP

- Subtract 3 pixels from top
- Subtract 3 pixels from left
- Subtract 4 pixels from right
- Subtract 4 pixels from bottom

Tile

Interpolation problem of TIP

Modified TIP reference range
MCTS Considerations – Temporal Candidate of AMVP and MERGE

- Temporal Candidate of AMVP and MERGE at The Column Boundary Between Tiles
  - Use AMVP and Merge to reduce the amount of motion information in the inter prediction
  - The block to the bottom right and at the center of the current PU are used as temporal candidates
  - There is a problem when the candidate block goes out of the column boundary
  - Exclude H block from candidate for above condition

![Diagram showing temporal candidate problem at column boundary between Tiles]
Tile-based Viewport Adaptive Video Streaming - cont’d

Encoder with MCTS

Raw video

MCTS bitstreams

Encoder

Extracted

EIS SEI Message (NAL)

File encapsulation

Delivery

Decoder

Merging

Renderer

Playback

Tile Selector

Selection & Chunk Request

DASH Client

Tile Priority Generator

Prediction Model

Adaptive Network Bandwidth Model

Raw video

Sub-picture bitstreams @quality 1

Sub-picture bitstreams @quality 2

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 1 + 8 8 + 1 1 + 8 8 + 1 1 + 8 8 + 1 1 + 8 8 +

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8

1 2 3 4
5 6 7 8
Experimental Results

• Experimental Setup
  • Uses the 8K test sequences selected by JVET
  • Be encoded with general coding options for Random Access (RA) coding structure
  • Use uniformly 3x3 9 Tiles
  • SHM12.3 encoder / HM 16.16 encoder

<table>
<thead>
<tr>
<th>Name</th>
<th>Resolution</th>
<th>Frame Length</th>
<th>Frame Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>KiteFlite</td>
<td>8192×4096</td>
<td>300</td>
<td>30 fps</td>
</tr>
<tr>
<td>Harbor</td>
<td>8192×4096</td>
<td>300</td>
<td>30 fps</td>
</tr>
<tr>
<td>Trolley</td>
<td>8192×4096</td>
<td>300</td>
<td>30 fps</td>
</tr>
<tr>
<td>GasLamp</td>
<td>8192×4096</td>
<td>300</td>
<td>30 fps</td>
</tr>
</tbody>
</table>

8K Test sequences

<table>
<thead>
<tr>
<th>Coding Option</th>
<th>SHM Parameter</th>
<th>HM parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>12.3</td>
<td>16.16</td>
</tr>
<tr>
<td>CTU size</td>
<td>64×64</td>
<td></td>
</tr>
<tr>
<td>Coding structure</td>
<td>RA</td>
<td></td>
</tr>
<tr>
<td>QP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Base Layer QP</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>Enhancement Layer QP</td>
<td></td>
<td>22</td>
</tr>
<tr>
<td>Tile</td>
<td>Uniformly 3x3 = 9 tiles</td>
<td></td>
</tr>
<tr>
<td>Slice mode</td>
<td>Disable all slice options</td>
<td></td>
</tr>
<tr>
<td>WPP mode</td>
<td>Disable all Wavefront parallel processing options</td>
<td></td>
</tr>
</tbody>
</table>
Experimental Results – cont’d

- **Bitrate Saving When Transmitting Only Some Tiles using MCTS**
  - Average bit rate savings of **51% and 87%** are achieved for 4 tiles and 1 tile using Modified SHVC encoder.
  - Average bit rate savings of **49% and 86%** are achieved for 4 tiles and 1 tile using Modified HEVC encoder.

- **Demo Video**
  - [https://youtu.be/--zDLYEau54](https://youtu.be/--zDLYEau54)

![Diagram showing 1 tile and 4 tiles]

<table>
<thead>
<tr>
<th>Name</th>
<th>4 tiles bitrate saving</th>
<th>1 tile bitrate saving</th>
<th>4 tiles bitrate saving</th>
<th>1 tile bitrate saving</th>
</tr>
</thead>
<tbody>
<tr>
<td>KiteFlite</td>
<td>52%</td>
<td>88%</td>
<td>51%</td>
<td>87%</td>
</tr>
<tr>
<td>Harbor</td>
<td>53%</td>
<td>88%</td>
<td>51%</td>
<td>87%</td>
</tr>
<tr>
<td>Trolley</td>
<td>50%</td>
<td>87%</td>
<td>49%</td>
<td>87%</td>
</tr>
<tr>
<td>GasLamp</td>
<td>49%</td>
<td>87%</td>
<td>47%</td>
<td>86%</td>
</tr>
</tbody>
</table>

| **Average bitrate saving** | 51% | 87% | 49% | 86% |

**Comparison ratio of the bitrate to select and transmit tiles using modified SHM and HM encoding**
Experimental Results

• Region-Wise Packing

Region-wise packed bitstream rendering
(AerialCity_3840×1920)
Experimental Results

- Extract Encoded Bitstream
Experimental Results

- Viewport Adaptive One Tile Rendering

Extracted bitstream rendering
(DrivingInCity / 1280×640 (3840×1920) / 1Tile (Texture))
MPEG-Immersive 3DoF+
Call for Proposal
3DoF(Degree of Freedom)+ and 6DoF Video

- Definitions of 3DoF+ and 6DoF omnidirectional video

  - Video that provides viewing freedom of motion in each direction, including head movement in the direction of yaw, pitch, and roll of 3DoF to provide full immersion (light field, multi-viewpoint 360 video)

  - Provides motion parallax in addition to the two sides of the existing 3DoF video and the point of view

* Source from Prof. Jewon Kang
Pseudo 3DoF+ and 6DoF 360 Video Streaming

Interaction ----> Viewpoint Selection ----> Viewpoint movement (e.g. Google street view) ----> Rendering

Free viewpoint (e.g. 3DoF+, 6DoF)

3DoF with viewpoint movement

- Get the nearest source view
- Render to projector
- Check user’s viewpoint & viewport

3DoF+ & 6DoF (supports free viewpoint)

- Render to projector
- Generate virtual view
- Check user’s viewpoint & viewport

Left Right

Forward Backward

Up Down

Left Right

Pitch Roll

Yaw

© 2019 Gachon University. All rights reserved.
Common Test Conditions (CTC)

CTC – w18089(Common Test Conditions on 3DoF+ and Windowed 6DoF)

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Class</th>
<th>Resolution</th>
<th>No. of views</th>
<th>Frame count</th>
<th>Frame rate</th>
<th>Source FoV</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClassroomVideo</td>
<td>A</td>
<td>4096x2048</td>
<td>15</td>
<td>120</td>
<td>30</td>
<td>360° x 180°</td>
</tr>
<tr>
<td>TechnicolorMuseum</td>
<td>B</td>
<td>2048x2048</td>
<td>24</td>
<td>300</td>
<td>30</td>
<td>180° x 180°</td>
</tr>
<tr>
<td>TechnicolorHijack</td>
<td>C</td>
<td>4096x4096</td>
<td>10</td>
<td>300</td>
<td>30</td>
<td>180° x 180°</td>
</tr>
<tr>
<td>TechnicolorPainter</td>
<td>D</td>
<td>2048x1088</td>
<td>16</td>
<td>300</td>
<td>30</td>
<td>46° x 25°</td>
</tr>
<tr>
<td>IntelKermit</td>
<td>E</td>
<td>1920x1080</td>
<td>13</td>
<td>300</td>
<td>30</td>
<td>77.8° x 77.8°</td>
</tr>
</tbody>
</table>
Common Test Conditions (CTC) - cont’d

Technical proposal with pre- and post-processing
Common Test Conditions (CTC) - cont’d

<table>
<thead>
<tr>
<th>Test class</th>
<th>Sequence Name</th>
<th># of source views</th>
<th># of anchor-coded views</th>
<th>Anchor-coded views</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>ClassroomVideo</td>
<td>15</td>
<td>15</td>
<td>All</td>
</tr>
<tr>
<td>A2</td>
<td>ClassroomVideo</td>
<td>15</td>
<td>9</td>
<td>v0, v7…v14</td>
</tr>
<tr>
<td>B1</td>
<td>TechnicolorMuseum</td>
<td>24</td>
<td>24</td>
<td>All</td>
</tr>
<tr>
<td>B2</td>
<td>TechnicolorMuseum</td>
<td>24</td>
<td>8</td>
<td>v0, v1, v4, v8, v11, v12, v13, v17</td>
</tr>
<tr>
<td>C1</td>
<td>TechnicolorHijack</td>
<td>10</td>
<td>10</td>
<td>All</td>
</tr>
<tr>
<td>C2</td>
<td>TechnicolorHijack</td>
<td>10</td>
<td>5</td>
<td>v1, v4, v5, v8, v9</td>
</tr>
<tr>
<td>D1</td>
<td>TechnicolorPainter</td>
<td>16</td>
<td>16</td>
<td>All</td>
</tr>
<tr>
<td>D2</td>
<td>TechnicolorPainter</td>
<td>16</td>
<td>8</td>
<td>v0, v3, v5, v6, v9, v10, v12, v15</td>
</tr>
<tr>
<td>E1</td>
<td>IntelKermit</td>
<td>13</td>
<td>13</td>
<td>All</td>
</tr>
<tr>
<td>E2</td>
<td>IntelKermit</td>
<td>13</td>
<td>7</td>
<td>v1, v3, v5, v7, v9, v11, v13</td>
</tr>
</tbody>
</table>

Anchor-coded views per class

<table>
<thead>
<tr>
<th>Test class</th>
<th>Frames – objective evaluation</th>
<th>Frames – subjective evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1, A2</td>
<td>1-32, 89-120</td>
<td>1-120</td>
</tr>
<tr>
<td>B1, B2</td>
<td>1-32, 269-300</td>
<td>1-300</td>
</tr>
<tr>
<td>C1, C2</td>
<td>1-32, 269-300</td>
<td>1-300</td>
</tr>
<tr>
<td>D1, D2</td>
<td>1-32, 269-300</td>
<td>1-300</td>
</tr>
<tr>
<td>E1, E2</td>
<td>1-32, 269-300</td>
<td>1-300</td>
</tr>
</tbody>
</table>

Frame ranges for 3DoF+ view synthesis
Common Test Conditions (CTC) - cont’d

- anchor view / source view

Viewpoint arrangement in ClassroomVideo sequence
Call for Proposals on 3DoF+ -Cont’d

• **Background**
  - MPEG defined degree of freedom of VR as 3DoF, 3DoF+, and 6DoF
  - Limited movements for user sitting in a chair is available for 3DoF+

• **Requirements**
  - Solution will be built on HEVC with 3DoF+ metadata (included in MPEG-I part 7)
  - Both objective and subjective quality evaluation will be performed

Architecture for 3DoF+ media
Call for Proposals on 3DoF+ -Cont’d

<table>
<thead>
<tr>
<th>Software name</th>
<th>Location</th>
<th>Tag/branch</th>
</tr>
</thead>
<tbody>
<tr>
<td>RVS</td>
<td><a href="http://mpegx.int-evry.fr/software/MPEG/Explorations/3DoFplus/RVS">http://mpegx.int-evry.fr/software/MPEG/Explorations/3DoFplus/RVS</a></td>
<td>v3.1</td>
</tr>
<tr>
<td>ERP-WS-PSNR</td>
<td><a href="http://mpegx.int-evry.fr/software/MPEG/Explorations/3DoFplus/WS-PSNR">http://mpegx.int-evry.fr/software/MPEG/Explorations/3DoFplus/WS-PSNR</a></td>
<td>v2.0</td>
</tr>
<tr>
<td>HDRTools</td>
<td><a href="https://gitlab.com/standards/HDRTools">https://gitlab.com/standards/HDRTools</a></td>
<td>v0.18</td>
</tr>
<tr>
<td>360Lib</td>
<td><a href="https://jvet.hhi.fraunhofer.de/svn/svn_360Lib">https://jvet.hhi.fraunhofer.de/svn/svn_360Lib</a></td>
<td>5.1-dev</td>
</tr>
<tr>
<td>HM</td>
<td><a href="https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware">https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware</a></td>
<td>16.16</td>
</tr>
</tbody>
</table>

List of used tools

<table>
<thead>
<tr>
<th>Sequence name</th>
<th>Rate 1</th>
<th>Rate 2</th>
<th>Rate 3</th>
<th>Rate 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClassroomVideo</td>
<td>6.5</td>
<td>10</td>
<td>15</td>
<td>25</td>
</tr>
<tr>
<td>TechnicolorMuseum</td>
<td>10</td>
<td>15</td>
<td>25</td>
<td>40</td>
</tr>
<tr>
<td>TechnicolorHijack</td>
<td>6.5</td>
<td>10</td>
<td>15</td>
<td>25</td>
</tr>
<tr>
<td>TechnicolorPainter</td>
<td>6.5</td>
<td>10</td>
<td>15</td>
<td>25</td>
</tr>
<tr>
<td>IntelKermit</td>
<td>6.5</td>
<td>10</td>
<td>15</td>
<td>25</td>
</tr>
</tbody>
</table>

Bitrate points not to be exceeded
System Overview

- General framework for 3DoF+ S/W
- Reduces bitrate by removing redundancy among the source views
- Reduces the number of decoders by merging sparse views into packed view

3DoF+ System Architecture
System Architecture

- General framework for 3DoF+ S/W
- Reduces bitrate by avoiding redundancy among the source views
- Reduces the number of decoders by merging sparse views into packed view

Block diagram for 3DoF+ S/W platform
Source View Pruning

- Generates central view that contains most of information of source views
- Remove redundancy between central view and source views
- Apply dilation to reduce edge artifacts in reconstruction

*T : texture, *D : depth

**Source View Pruning**

1. **Source views**: $T_1^* D_1^*$, $T_N^* D_N^*$
2. **Generate N cfg files from central view to source view**
3. **Central view synthesis**: $T_C D_C$
4. **HEVC Encoder**
   - **Encoding**
5. **HEVC Decoder**
   - **Decoding**

**For i in [1, N], do**:

1. **Warping**: $(D_i^C)_{pruned}$
2. **Texture mapping**
3. **Dilation**
4. **Medium value hole filling**
5. **(T_i^C)_{sparse} (D_i^C)_{sparse}**

**Diagram**

- Central view synthesis
- Source view pruning
- HEVC Encoder
- HEVC Decoder
- Texture mapping
- Dilation
- Medium value hole filling
- (T_i^C)_{sparse} (D_i^C)_{sparse}
Source View Pruning: Central View Synthesis

• Background
  • For pruning, central view must contain most of information among source views
  • If not, sparse views convey lots of information of source view
    > Causes bitrate increase due to rendering problem

• Solution
  • Synthesizes central view with large resolution, at the center of source views
  • If the test sequence is ERP, central view covers omnidirectional range

Sparse view mask of v0, TechnicolorHijack

Synthesized Central view

With 180° x 180° central view  With 360° x 180° central view

TechnicolorMuseum

TechnicolorHijack
Source View Pruning - cont’d

- Discards pixels from sparse views that are conveyed by central view
- Modified RVS to use 3D warping from central view to source views
- Distorted triangle means the area only represented by source view
- Generates sparse view texture, depth, and mask

(a) Input view  (b) Synthesized view

Sparse view mask

Sparse view texture
Source View Pruning - cont’d

• Background
  • For pruning, non-encoded central view is used
  • However, encoded central view is used in source view reconstruction
    -> Causes artifacts due to the mismatch between pruning and reconstruction

• Solution
  • Encoded central view is used for pruning

Reconstructed source view of v1, ClassroomVideo

Artifact

With non-encoded central view  With encoded central view
Source View Pruning: Region Dilation

- **Background**
  - In reconstructed source view, it shows edge artifacts
    > That causes user’s discomfort, which leads to bad Quality of Experience (QoE)

- **Solution**
  - After computing the mask, applies region dilation to increase sparse view pixels
  - In our experiment, MORPH_CROSS is used

Without dilation

- dilation_size=2
- dilation_size=8

With dilation

- MORPH_RECT
- MORPH_CROSS
- MORPH_ELLIPSE
Source View Pruning : Hole Filling

• Background
  • Pixel value for sparse view holes was neutral gray (512, 512, 512)
    To reduce the bitrate, representative pixel value for sparse view is needed

• Solution
  • Bitrates for medium value of sparse view, interpolation, fill nearest value were measured
  • Medium value showed the lowest bitrate

<table>
<thead>
<tr>
<th></th>
<th>With neutral gray</th>
<th>Medium value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No. of view</td>
<td>QPSlice</td>
</tr>
<tr>
<td>A - s1</td>
<td>22</td>
<td>3840.7</td>
</tr>
<tr>
<td></td>
<td>37</td>
<td>235.2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Interpolation</th>
<th>Nearest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No. of view</td>
<td>QPSlice</td>
</tr>
<tr>
<td>A - s1</td>
<td>22</td>
<td>21122.632</td>
</tr>
<tr>
<td></td>
<td>37</td>
<td>750.622</td>
</tr>
</tbody>
</table>
Source View Pruning : Poles Filtering

- **Background**
  - Projection error of ERP because of distortion in top and bottom area (1)
  - Projection error due to warping issue of RVS in left and right area (2)
  - > Unnecessary areas are included, which increases bitrate

- **Solution**
  - Set top and bottom blocks as seeds, and conduct region growing to remove useless areas
  - Mask left and right sides

<table>
<thead>
<tr>
<th>Class</th>
<th>QP</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Without poles filtering</td>
</tr>
<tr>
<td>A</td>
<td>22</td>
<td>7680x4672</td>
</tr>
<tr>
<td>B</td>
<td>22</td>
<td>7680x3648</td>
</tr>
<tr>
<td>C</td>
<td>22</td>
<td>7680x4992</td>
</tr>
</tbody>
</table>

All frames
Sparse View Packing

- Extracts informative regions from sparse views and merge into packed view
- Removes overlapped regions to reduce the packed view size
- Applies depth refinement to reduce bitrate

\[
\begin{align*}
\text{Sparse views: } & T_1^* \quad D_1^* \\
& \quad \cdots \\
& T_N^* \quad D_N^* \\
\end{align*}
\]

\[
\begin{align*}
M_i^* & \xrightarrow{\text{Merging by intraPeriod}} \text{Region growing} \quad \text{Removing overlapped region & poles filtering} \\
\text{Sort region} & \xrightarrow{\text{Region allocation}} \text{Writing metadata} \\
\end{align*}
\]

*\(M\): mask, *\(T\): texture, *\(D\): depth
### Why Packing?

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Class</th>
<th>Resolution</th>
<th>No. of views</th>
<th>Frame count</th>
<th>Frame rate</th>
<th>Source FoV</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClassroomVideo</td>
<td>A</td>
<td>4096x2048</td>
<td>15</td>
<td>120</td>
<td>30</td>
<td>360° x 180°</td>
</tr>
<tr>
<td>TechnicolorMuseum</td>
<td>B</td>
<td>2048x2048</td>
<td>24</td>
<td>300</td>
<td>30</td>
<td>180° x 180°</td>
</tr>
<tr>
<td>TechnicolorHijack</td>
<td>C</td>
<td>4096x4096</td>
<td>10</td>
<td>300</td>
<td>30</td>
<td>180° x 180°</td>
</tr>
<tr>
<td>TechnicolorPainter</td>
<td>D</td>
<td>2048x1088</td>
<td>16</td>
<td>300</td>
<td>30</td>
<td>46° x 25°</td>
</tr>
<tr>
<td>IntelKermit</td>
<td>E</td>
<td>1920x1080</td>
<td>13</td>
<td>300</td>
<td>30</td>
<td>77.8° x 77.8°</td>
</tr>
</tbody>
</table>

- **High resolution and many views**
  - (4096x2048x15x120…) = impossible
  - → Redundancy

- **Encoding inefficiency**
  - Limit to 8K image size
  - Goal: Reduce total sequence size
Packing: Fixed Size Block

Pruned source view example (=Sparse view)

Add blocks which contain information more than threshold

Sparse view

Aggregate all sparse view in one 8K image

Packing of all sparse views

Raw Data (Contain each block position)
Packing: Fixed Size Block (Problem)

- Time validity of packing view
  - Compression performance
  - *Intra-period merging*
Packing: Intra Period Merging

- Locations of informative regions change when processing frame-by-frame.
- When merging the masks by intra-period, locations are fixed.
  -> decreases required bitrate.
Packing: Region Growing Block

- Applies 8-neighbors binary region growing

* R : Region

Region position allocation in packed view

<table>
<thead>
<tr>
<th>Class</th>
<th>QP</th>
<th>Without region growing</th>
<th>With region growing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Resolution</td>
<td>Bitrate(Mbps)</td>
</tr>
<tr>
<td>A</td>
<td>22</td>
<td>7680x2688</td>
<td>157.4</td>
</tr>
<tr>
<td>B</td>
<td>22</td>
<td>7680x2944</td>
<td>90.8</td>
</tr>
<tr>
<td>C</td>
<td>22</td>
<td>7680x4544</td>
<td>123.3</td>
</tr>
</tbody>
</table>

32 frames (0 – 31)
**Packing: Removing Overlapped Region**

- Removes or resizes overlapped regions to reduce the packed view size
- Restricts width and height size of packed view
- Sorts regions with height in descending order and applies position allocation

```
removeSmallRegion()
removeSideOverlappedRegion()
divideVertexOverlappedRegion()
```

Sparse view mask regions

Sort regions by height (descending)

```
max_region_width = 5, max_region_height = 3
```

End
Packaging: Stride Packing

Region position allocation in packed view with stride

<table>
<thead>
<tr>
<th>Class</th>
<th>QP</th>
<th>Without Striding</th>
<th>With Striding</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Resolution</td>
<td>Bitrate(Mbps)</td>
</tr>
<tr>
<td>A</td>
<td>22</td>
<td>7680x4672</td>
<td>130.2</td>
</tr>
<tr>
<td>B</td>
<td>22</td>
<td>7680x3904</td>
<td>70.1</td>
</tr>
<tr>
<td>C</td>
<td>22</td>
<td>7680x5440</td>
<td>96.4</td>
</tr>
</tbody>
</table>

32 frames (0 – 31)

Unpacking
Results of Stride Packing

Classroom video sequence (7680x2560)
Depth Refinement

• Background
  • Depth contains sharp edges and large blocks with similar pixel values
    -> Requires a large amount of bitrate

• Solution
  • Applies a 3x3 spatial median filter to smoothen the edge
  • Shows bitrate saving approximately 43.2% for class A and 28.8% for class B

<table>
<thead>
<tr>
<th>Class</th>
<th>QP</th>
<th>Bitrate (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No refinement</td>
<td>With refinement</td>
</tr>
<tr>
<td>A</td>
<td>20</td>
<td>11.5</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>8.5</td>
</tr>
<tr>
<td></td>
<td>28</td>
<td>6.3</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td>4.2</td>
</tr>
<tr>
<td>B</td>
<td>14</td>
<td>13.3</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>10.7</td>
</tr>
<tr>
<td></td>
<td>26</td>
<td>8.0</td>
</tr>
<tr>
<td></td>
<td>29</td>
<td>6.0</td>
</tr>
</tbody>
</table>
Experimental Results

- For response of 3DoF+ CfP, 5 input documents are proposed
- All of proposals introduce pruning and packing architecture
- Proposals of Nokia and PUT & ETRI are compared with our proposed method

<table>
<thead>
<tr>
<th>Number</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>m47179</td>
<td>Philips response to 3DoF+ Visual CfP</td>
</tr>
<tr>
<td>m42372</td>
<td>Description of Nokia’s response to CFP for 3DOF+ visual</td>
</tr>
<tr>
<td>m47407</td>
<td>Technical description of proposal for Call for Proposals on 3DoF+ Visual prepared by Poznan University of Technology (PUT) and Electronics and Telecommunications Research Institute (ETRI)</td>
</tr>
<tr>
<td>m47445</td>
<td>Technicolor-Intel Response to 3DoF+ CfP</td>
</tr>
<tr>
<td>m47684</td>
<td>Description of Zhejiang University’s response to 3DoF+ Visual CfP</td>
</tr>
</tbody>
</table>
Experimental Results - cont’d

• Server
  • 2 Linux servers (ubuntu 16.04) were used in experiments
  • 2 Intel Xeon E5-2687w (48 threads) CPUs and 128GB memory
  • 2 Intel Xeon E5-2620 (24 threads) CPUs and 128GB memory

• Software
  • OpenCV 3.4.2 was used in source view pruning, reconstruction, and RVS
Experimental Results - cont’d

- Proposed method shows the best results in Classroom video test sequence.
- Anchor encodes all of the source views with HEVC encoder individually.
- 36.0% of BD-rate saving is shown in proposed method with dilation size 0
Experimental Results - cont’d

- Proposed method is more efficient than the anchor
- For the bandwidth over 15Mbps, our method shows better results than multi-layer based method

![Graph showing experimental results](image-url)
Conclusion

• Pruning and Packing Methods and Insights
  • Inter-view redundancy removal by pruning using view synthesis
  • Multi-view residual compression by packing
  • Implemented multi-view transmission system showed BD-rate saving up to 36.0%

• Future Work
  • Color correction for central view
  • Depth coding
  • Real-time 3DoF+/6DoF video streaming system with parallel processing and GPU acceleration
Thank You!

http://mcsl.gachon.ac.kr
Questions > esryu@gachon.ac.kr
Appendix

The Summary of Call for Responses
Experimental Results (Philips)

- Concept of Philips is to:
  - Keep the projection of the source views to prevent resampling and view warping errors
  - Pruning mask for intra period

Data processing flow of the proposal
Experimental Results (Philips)

- Concept of Philips is to:
  - Block tree representation
  - Prune each source view with hierarchical architecture

Each arrow is a synthesis operation.

Hierarchial pruning architecture of proposal

Subdivision of block tree
Experimental Results (TCH-intel)

• Concept of Technicolor & intel is to:
  • Refine the depth map for better view synthesis
  • Pack a central view with residual view into one view
Experimental Results (TCH-intel)

- Concept of Technicolor & intel is to:
  - Extra band size for hosting patches
  - Time validity of patches (=packing view)

Extra band with central view

intra period duration
Experimental Results (Nokia)

- Concept of Nokia is to:
  - PCC based approach
  - Segment into a set of shards (=3D bounding box)
Experimental Results (Nokia)

• Concept of Nokia is to:
  • Mosaic (=packing view) contains ordering shards
Experimental Results (PUT & ETRI)

- Concept of PUT & ETRI is to:
  - Split the source views in the spatial frequency domain
  - Synthesize a base view and supplementary views

Coding scheme of the proposal
Experimental Results (PUT & ETRI)

• Concept of PUT & ETRI is to:
  • Base layer, which contains content that is spatially low-pass filtered, and that can be efficiently coded with classic predictive coding like HEVC
  • Residual layer, which contains spatial high frequency residual content that can be represented jointly for a number of views

Base view and supplementary view of PUT & ETRI
Left: 3 input views, right: base view (top) and supplementary view (bottom)
Experimental Results (Zhejiang)

- Concept of Zhejiang is to:
  - Basic views / Non-basic views
  - Sub-picture cropping

The flow diagram of encoder side

Illustration of sub-picture cropping
Experimental Results (Zhejiang)

• Concept of Zhejiang is to:
  • Valid view selection

valid rate = \frac{\text{valid pixels of cropped view}}{\text{valid pixels of whole view}}