**3GPP TSG-SA WG4 Meeting #132 S4-250880**

**Fukuoka, JP, 19 – 23 May 2025**

**Source: China Mobile Com. Corporation**

**Title: [FS\_Beyond2D] Scenario 1: Detailed Test Condition**

**Agenda item: 9.7**

**Document for: Agreement**

**1. Introduction**

This proposal provides updates on encoding and decoding constraints, and detailed test condition for Scenario 1: UE-to-UE Stereoscopic Video Live Streaming.

**2. Proposal**

It is proposed to agree the following changes to the 3GPP draft TR 26.956 V0.4.0

\* \* \* First Change \* \* \*

[LS-13] HTM Codec Software, version 16.3, <https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/branches/HTM-16.3-fixes/cfg/MV-HEVC/>

[LS-22] MPEG, Conformance Testing for Multiview Main and 3D Main profiles, https://mpeg.chiariglione.org/sites/default/files/files/standards/parts/docs/w14984-v2-w14984.zip

\* \* \* Second Change \* \* \*

### 7.2.4 Encoding and Decoding Constraints

Table 7.2.4-1 provides an overview of encoding and decoding constraints for UE-to-UE Stereoscopic Video Live Streaming scenario using H.265/HEVC and MV-HEVC. This information supports the definition of detailed anchor conditions.

Table 7.2.4-1 Encoding and Decoding Configurations

|  |  |  |
| --- | --- | --- |
| Encoding and Decoding Constraints | H.265/HEVC | MV-HEVC |
| Relevant Codec and Codec Profile/Levels | H.265/HEVC Main 10 Profile  Level 4.1, 5.1 | Multiview Main or Multiview Main10 profile  Level 4, 5.1 and higher |
| Random access frequency | 1 second | 1 second |
| Bit rates and quality configuration | Fixed QP: [17, 22, 27, 32, 37]  CBR  Half Width/Height: 5-8Mbps  Full Width/Height: 8-16Mbps  Capped-VBR | Similar to HEVC configurations |
| Bit rate parameters (CBR, VBR, CAE, HRD parameters) | Covering a range of relevant bitrates and qualities | Covering a range of relevant bitrates and qualities |
| Latency requirements and specific encoding settings | Low latency requirements | Low latency requirements |
| Encoding complexity context | Real-time encoding, Cloud-based encoding | Real-time encoding, Cloud-based encoding |
| Required decoding capabilities | H.265/HEVC Main 10 Profile  Level 4.1, 5.1 | Multiview Main or Multiview Main10 profile  Level 4, 5.1 and higher |

### 

\* \* \* Third Change (All New) \* \* \*

### 7.2.8 Test Condition

#### 7.2.8.1 Test model and configuration files

The encoder configuration settings for both encodings are consistent with the common test conditions (CTC) define in MPEG [22], which are outlined below:

- Inter-view coding structure

- 2 view case: left-right (in coding order)

- I-P inter-view prediction MV-HEVC

- Temporal prediction structure: GOP 8, intra every 24 frames (random access at ~1sec)

- Full resolution texture coding

- Codec software: HTM v16.3 for Simulcast HEVC and MV-HEVC

The following configuration files are provided in [LS-13]:

- /HTM-16.3-fixed/cfg/MV-HEVC/baseCfg\_2view.cfg: Used to configure input/output filenames and encoder parameters (I-frame interval, number of B-frames, etc.)

- /HTM-16.3-fixed/cfg/MV-HEVC/qpCfg\_QP25.cfg: : Used to configure the encoding QP

- /HTM-16.3-fixed/cfg/MV-HEVC/seqCfg\_Shark.cfg: Contains the source sequence parameters (resolution, frame count, frame rate, etc.)

For each selected test sequence, configuration files containing information needed for HTM-16.3 configuration will be provided.

#### 7.3.9.2 Rate points and test conditions

The performance of both MV-HEVC and Simulcast HEVC is compared and evaluated in terms of PSNR (dB) and bitrate (kbps) over the set of QP values [17, 22, 27, 32, 37].



#### 7.3.9.3 Profiles

MV-HEVC Main Profile is used.

#### 7.3.9.4 Bitstream Generation, output

The HTM v16.3 is used to encode and decode test sequences as described in clause 7.2.8.1.

Below are examples of command lines for encoding and decoding test sequence:

- Before compilation, navigate to source/Lib/TLibCommon/TypeDef.h and modify the following parameter to configure the software as an MV-HEVC encoder:

/\*\* \file TypeDef.h

\brief Define macros, basic types, new types and enumerations

\*/

#ifndef \_\_TYPEDEF\_\_

#define \_\_TYPEDEF\_\_

#ifndef \_\_COMMONDEF\_\_

#error Include CommonDef.h not TypeDef.h

#endif

#include <vector>

#include <utility>

//! \ingroup TLibCommon

//! \{

/////////////////////////////////////////////////////////////////////////////////////////

///////////////////////////////// EXTENSION SELECTION ///////////////////////////////////

/////////////////////////////////////////////////////////////////////////////////////////

/\* HEVC\_EXT might be defined by compiler/makefile options.

Linux makefiles support the following settings:

make -> HEVC\_EXT not defined

make HEVC\_EXT=0 -> NH\_MV=0 H\_3D=0 --> plain HM

make HEVC\_EXT=1 -> NH\_MV=1 H\_3D=0 --> MV only

make HEVC\_EXT=2 -> NH\_MV=1 H\_3D=1 --> full 3D

\*/

#ifndef HEVC\_EXT

#define HEVC\_EXT 1

#endif

#if ( HEVC\_EXT < 0 )||( HEVC\_EXT > 2 )

#error HEVC\_EXT must be in the range of 0 to 2, inclusive.

#endif

#define NH\_MV ( HEVC\_EXT != 0)

#define NH\_3D ( HEVC\_EXT == 2)

#define NH\_MV\_ALLOW\_NON\_CONFORMING 0 // Allow non-conforming representations formats

/////////////////////////////////////////////////////////////////////////////////////////

/////////////////////////////////// FIXES AND INTEGRATIONS ////////////////////////

/////////////////////////////////////////////////////////////////////////////////////////

#if NH\_MV

// Things that needs to be fixed also in the Specification ... CHECK if this has been done

#define NH\_MV\_FIX\_NO\_REF\_PICS\_CHECK 1 // !!SPEC!!

#define NH\_MV\_FIX\_INIT\_NUM\_ACTIVE\_REF\_LAYER\_PICS 1 // Derivation of NumActiveRefLayerPIcs. !!SPEC!!

#define NH\_MV\_FIX\_NUM\_POC\_TOTAL\_CUR 1 // Derivation of NumPocTotalCur for IDR pictures. !!SPEC!!

// To be done

#define NH\_MV\_HLS\_PTL\_LIMITS 0

#define NH\_MV\_SEI\_TBD 0

#endif

- to encode a test sequence:

./TAppEncoder.exe -c baseCfg\_2view.cfg -c seqCfg\_Shark.cfg -c qpCfg\_QP25.cfg

- to decode a test sequence:

./TAppDecoder.exe -b stream.bit -o shark\_qp25.yuv -w 1

Additionally, FFmpeg supports decoding MV-HEVC in x265 v4.0 and later (Released in September 13th, 2024). The following guide, adapted from spatialgen.com (<https://spatialgen.com/blog/encode-mvhevc-with-ffmpeg/> ), explains how to encode MV-HEVC video using FFmpeg.

- Enable MV-HEVC Support in x265

make ENABLE\_MULTIVIEW=ON

- Create an mv\_config.cfg file for the x265

#Configure number of views in the multiview input video#

#--num-views <integer>#

--num-views 2

#Configuration for the input format of the video#

#--format <integer>#

# 0 : Two seperate input frames#

# 1 : One input frame with left and right view#

# 2 : One input frame with top and bottom view#

--format 1

#Configure input file path for each view#

##NOTE:Other input parameters such as input-csp/input-depth/fps must be configured through CLI##

--input "stereo\_raw.yuv"

- Run x265 for MV-HEVC Encoding, specify the correct frame rate, if using side-by-side stereoscopic video as input, the input resolution width should be half of the original video width.

./x265 --multiview-config mv\_config.cfg \

--fps 23.98 --input-res 960x1080 --output mvhevc\_output.hevc \

--profile main10 --colorprim bt709 \

--transfer bt709 --colormatrix bt709

\* \* \* End of Changes \* \* \*