**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 7**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC 1/SC 29/WG 7 m** **57827**

**Online – October 2020**

**Title:** [V-Mesh] VE 4.1 on IBSM metric rig rotation

**Author: Jean-Eudes Marvie**

# Introduction

During meeting 136, the group requested to perform the computation of the IBSM using YUV BT color space, instead of in the RGB space. Following the implementation, the same validation tests as the ones performed for RGB had to be performed.

During this meeting, we also raised an issue that could show up regarding the rotation of virtual camera rig used in IBSM metric. The team wanted to verify if some rotation of the rig would introduce a strong bias in the results. We decided that we would do additional testing using similar methodology but rotating the rig for some random orientations. This work was performed by InterDigital, Tencent and Sony. The rotation of the rig was implemented by Tencent and presented in m57582 “[V-Mesh] Image based mesh metrics with user-defined camera positions”. The software feature is available in the master with tag 0\_1\_12:

[http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-mmetric/-/tree/ 0\_1\_12](http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-mmetric/-/tree/%200_1_12)

An additional feature was added to the software to clamp the PSNR results to a maximum value of 999.99 to prevent numerical issues when processing Mean values over sequences. This feature was delivered on the following branch:

<http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-mmetric/-/tree/version_0_1_13>

# VE 4.1 Participants

The following table lists the contacts and their companies/institutes participating to VE 4.1

|  |  |  |
| --- | --- | --- |
| Contact | Affiliation | Email |
| Jean-Eudes Marvie | InterDigital | Jean-eudes.marvie@interdigital.com |
| Danillo Graziosi | Sony | Danillo.Graziosi@sony.com |
| Chao Huang | Tencent | mrchaohuang@tencent.com |
| Xiang Zhang | Tencent | xxiangzhang@tencent.com |
| Shan Liu | Tencent | shanl@tencent.com |
|  |  |  |

The 3DG (WG7) reflector was used for exchanges related to VE 4.1.

# VE 4.1 Mandates

* Perform the implementation of the YUV and run the validation tests. Once the implementation performed in the git repository it will be reported on the reflector together with a description document.
  + Due date 2021/08/15. Done. CfP draft updated accordingly and shared.
* Perform experimentation on rotating the rig. Once performed, the results will be shared on the reflector together with a description document.
  + Due date 2021/09/15. Done. Results shared on the Nextcloud server.
* A call will be organized by the VE4.1 coordinator to validate the YUV implementation as well as to review and validate the YUV and rotation results. In case of no additional objection is raised during this call the VE4.1 will be considered closed.
  + Due date 2021/09/15. Done
  + Results were discussed with a wider group than only the VE participants.

# YUV implementation

The YUV implementation has been implemented in version 0.1.12 and is available on the master with following tag:

<http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-mmetric/-/tree/0_1_12>

The YUV PSNR’s are computed as follows. This statement is an extract of the CfP draft that have been updated accordingly, please refer to the CfP for additional details and figures.

The calculus of the distortion is based on the general Mean Squared Error (MSE) formula. Let be a sample of an original image, a sample of a distorted image and the width of both buffers in pixels. The for the two images is calculated as follows:

In the rest of this document, we will decline this formula to our needs, especially restricting processing to parts where the mask generated from the reference and distorted models is equal to one. Let be a matrix of same size of and (Each mask buffer contains for each pixel i,j a binary value set to 1 if a projection for this pixel exists and 0 otherwise.).

Let be the width of the buffers in pixels. The combinate number of projected pixels of all the view directions for one frame is:

A conversion from RGB space to YUV space is performed by using ITU-R BT.709, since YUV space correlates better with human perception.

In the following the color image/buffer is considered YUV.

Let be a sample of the color image/buffer of the reference model for view direction , a sample of the color image/buffer of the distorted model for view direction . The color MSE for the YUV images, noted , is calculated as follows on each component in :

Let be a sample of the depth image/buffer of the reference model for view direction , a sample of the depth image/buffer of the distorted model for view direction . Let be the dynamic of the depth signal initialized with the diagonal of the bounding box of both models. The depth MSE for the depth images/buffers, noted , is calculated as follows:

This depth renormalization to 255 is used to get geometric MSEs and PSNRs comparable to the color ones.

The final metric results per frame are the followings:

And the respective associated with are computed with the following formula:

The experiment previously done with RGB metric on quantization/dequantization with different resolution and count, local distortions and decimation were re-run with the YUV implementation to validate the implementation. The complete results are available in accompanying file anchor-merror-raster-charts-yuv.pdf. One can see on these reaults that the behavior is homogeneous to the RGB one.

# Rig rotation

The experimentation was based on the version 0.1.12 of the mmetric software:

<http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-mmetric/-/tree/0_1_12>

The experimentation is using the rig rotation feature introduced in the mpeg-mmetric by Tencent team in: m57582 [V-Mesh] Image based mesh metrics with user-defined camera positions.

The experimental setup is to run the metric on some datasets with different rig orientation and check the PSNR variations among these different rotations. We selected three orientations:

1. ibsmCameraRotation ”0.0 0.0 0.0” (default)
2. ibsmCameraRotation ”0.0 0.0 90.0”
3. ibsmCameraRotation ”45.0 45.0 45.0”

* Tencent team did run the metric on the set of sequences presented in m57581 [V-Mesh] Image based mesh metrics on texture compression. Those sequences were used to evaluate the metric on sequences having no distortion on geometry and varying HM compression levels on texture maps.
* InterDigital team did run the metric for the first frame of all the anchor models with varying QP (geometry)/QT (UVs) quantization parameters and no distortion on texture maps.
* Finally, Sony team did run the metric on a subset of the Anchor models.

## Tests on anchors

Complete raw results are available in this archive for basket, dancer, football, levi, mitch, thomas and longdress. The anchor\_longdress folder present additional analysis files for AI/RA all rates. The script that was used to generate the results is attached in the accompanying documents.

In our observations we noticed that the rate point that shows higher discrepancies was R5, but we found that the largest difference between the different camera positions is only **0.4dB in a single frame, and in average is usually around 0.04dB**. See for instance a comparison of GEO\_PSNR, frame-by-frame for longdress, R5 all-intra condition in Figure 1.

Graphical user interface

Description automatically generated

Figure a comparison pf GEO\_PSNR, frame-by-frame for longdress, R5 all-intra condition per frame

## HM texture compression

Results for basket, Mitch and Levi are given in the accompanying file: “ibsm-vs-textureQP-ibsmCameraRotation.xlsx”.

We observe that the averaged PSNR is quite close. From the RD curve you don’t see much difference (Figure 2). The per-frame differences (Figure 3) are smaller than Anchor’s results. For basketball\_player at QP=42, the **maximum PSNR difference is about 0.08dB**. We guess the reason is that the geometry plays an important role in it, in this experiment the geometry is lossless, so the PSNR difference from different camera angles is smaller.

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generatedChart

Description automatically generated

Figure . 32 frame average

Graphical user interface, application, PowerPoint

Description automatically generated

Figure .Per frame difference

## Quantization of positions and UV coordinates

Result raw csv files are provided in the accompanying folders prefixed with qdeq for twelve models, with qp/qt varying from 8 to 12, for resolution in 512,1024,2048,4092 and for nbCamera in 16,18,24, 32,64.

The Figure 4 shows some difference results for the default setup proposed (16 views of 2K resolution). We observe **a maximum difference of 0.22dB for Dancer and 0.05dB for Levi.**

Graphical user interface, application, table, Excel

Description automatically generated Graphical user interface, application, table, Excel

Description automatically generated

Figure 4 results for varying QP/QT for dancer model (left) and the Levi model (right). The three rightmost columns show differences between PSNRs.

**Recommendations**

YUV support has been fully implemented and tested in the Anchor chain. We recommend using this version for the CfP and subsequent uses.

Experimentation with the rig rotation did not emphasis strong discrepancies when using different angles. We thus recommend using the default parameters of the metric (no rotation) for the CfP and subsequent uses.