m56759
EE4.1 on mesh metric status

Jean-Eudes Marvie (Interdigital)

Danillo Bracco Graziosi (Sony)

EE4.1 Summary

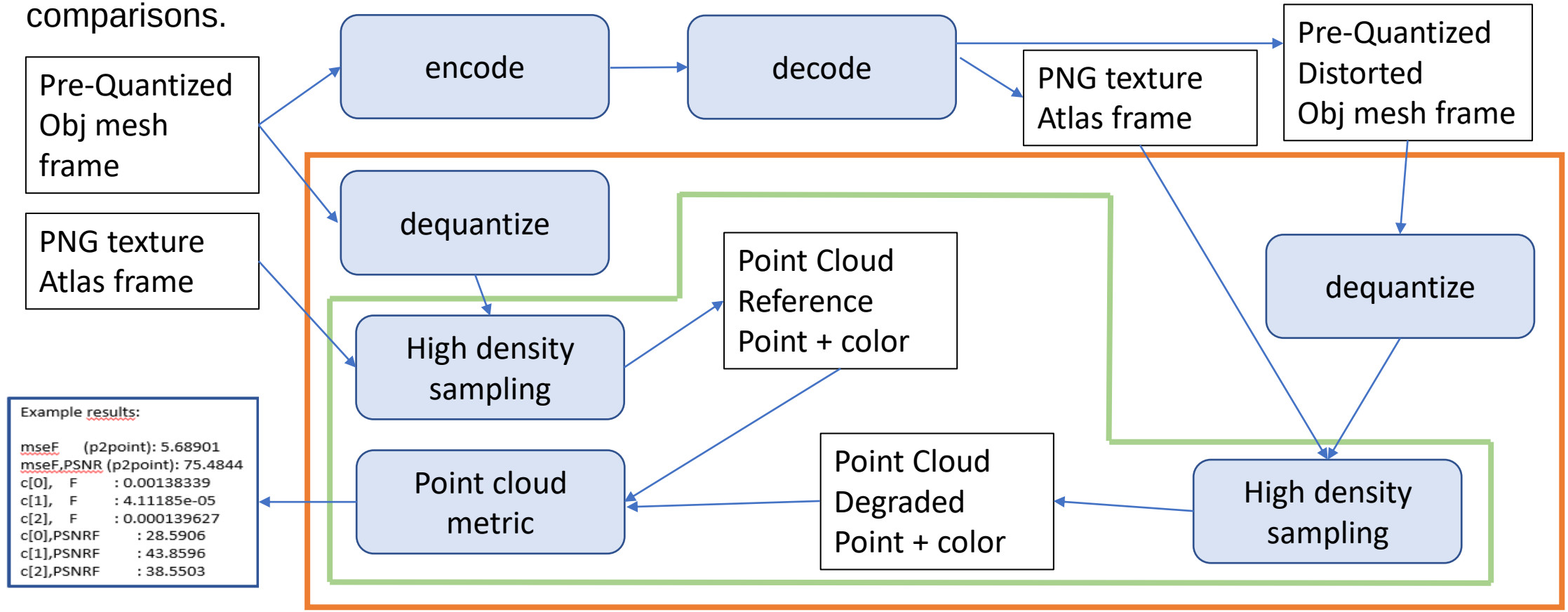
- Status report for EE4.1
  - Concluded:
    - mmetric software will be used
      → http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-mmetric.git
    - metric for lossy comparison is provided using sampling and PCC/PCQM
      - map/face/grid/sdiv/**ediv** sampling were implemented and tested
    - metric for lossless compression is provided (not using sampling)
  - On-going:
    - metric for near-lossless has been implemented, still to be tested
    - metric for lossless compression to evaluate texture map has been implemented
    - choice of sampling mode being evaluated
      - With high number of samples (e.g. 4 million), choice of sampling node is not critical
      - ediv might be our first choice, sampling parameters were provided for each model
  - Issues
    - For the near-lossless metric, encoder should dump the mapping for the triangles between the original and distorted model
    - PCC and PCQM metrics still under discussion
      - Should PCQM be provided as response to the CfP? Optional or mandatory?
      - Correlation between PCQM and subjective test results from the CfP responses should be evaluated to validate the metric.
    - No metric for normal in lossy and near-lossless mode (supported for lossless mode)
      - Decided to discuss during standardization and not consider normal coding for CfP

Metrics - context

- Lossless coding/decoding
  - Decoded topology must be identical to source at vertex shift difference
    - For instance, A B C and B C A are equivalent, but A B C and A C B are not
  - Decoded vertex attributes and texture maps must be identical to source
- Near lossless coding/decoding
  - Decoded topology must be identical to source at vertex shift difference
  - Decoded vertex attributes and texture maps can be different
- Lossy coding/decoding
  - Decoded topology can be different (can have less triangles),
  - Decoded vertex attributes and texture maps can be different

Lossy and near lossless metric - overview

Highly sample the textured meshes and use colored point cloud metrics [pcc_error, PCQM] for the comparisons.

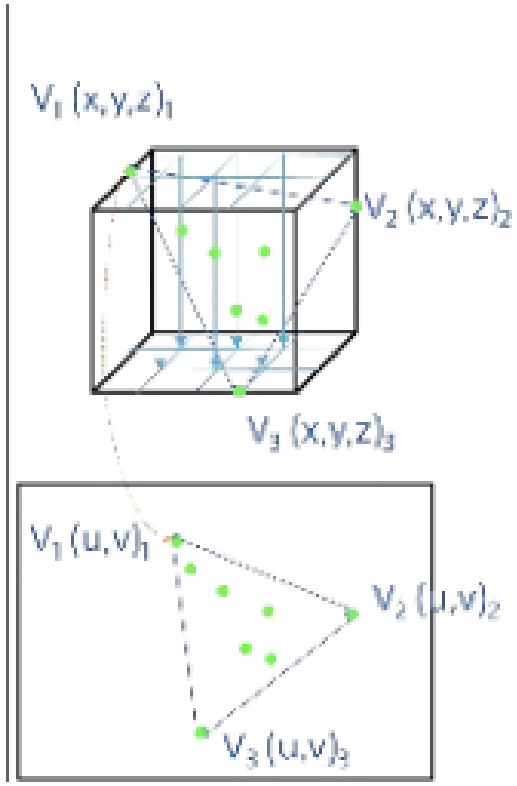


The green part was implemented inside derror

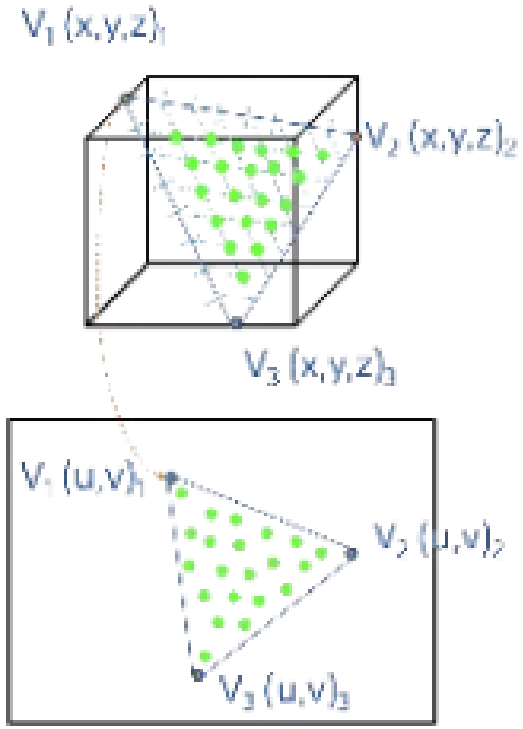
The orange part was implemented merror softwares for faster executions

For near lossless mode an additional topology validation step is added in merror (see latter in the presentation)

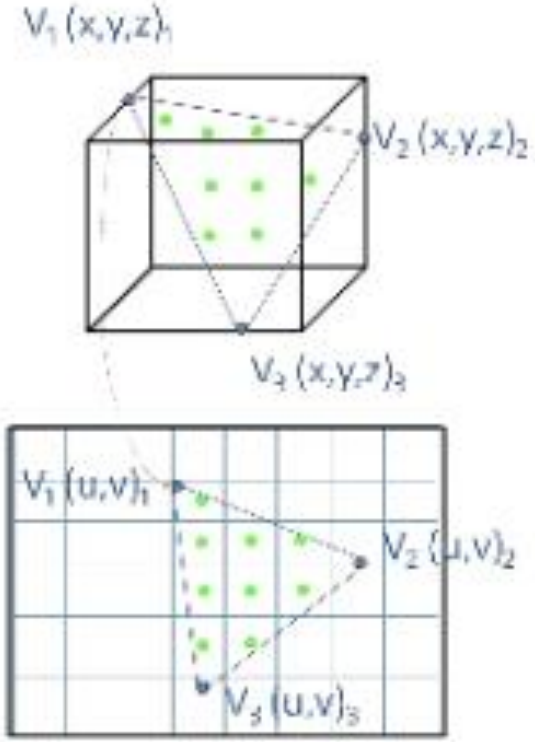
Lossy and near lossless metric - sampling



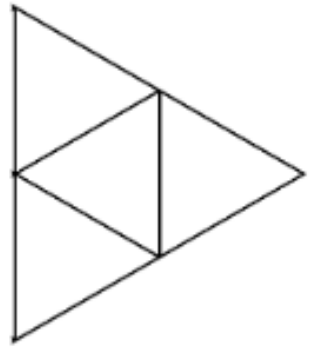
Grid sampling



Face sampling



Map sampling



Triangle subdivision:

Sdiv sampling
- Stop on area criterion

Ediv sampling
- Stop on edge length

Lossy and near lossless metric - assesment

- Some result charts were computed using
  - derror: face, grid and map sampling
    - Outputs mpeg pcc error metric: d1, d2 Y, U, and V PSNRs
  - merror: face, grid, map, sdiv, ediv sampling
    - Outputs mpeg pcc error metric: d1, d2 Y, U, and V PSNRs
    - Outputs PCQM error metric

Lossy and near lossless metric - assesment

- Both metrics were applied to the following distorted models
  - Quantization of positions and uv coordinates (Qp x Qt matrix) using draco –CL 10
    - on all the textured models of the anchor (original version, not pre-quantized)
  - Introduction of artefacts (local artefacts), remove a triangle every n triangle
    - On all the textured models of the anchor (original version, not pre-quantized)
  - Quantization of texture map (texture QP) using HM
    - on basketball_player_00000001 model
  - Quantization of positions (Qp) performed on non dense color per vertex models:

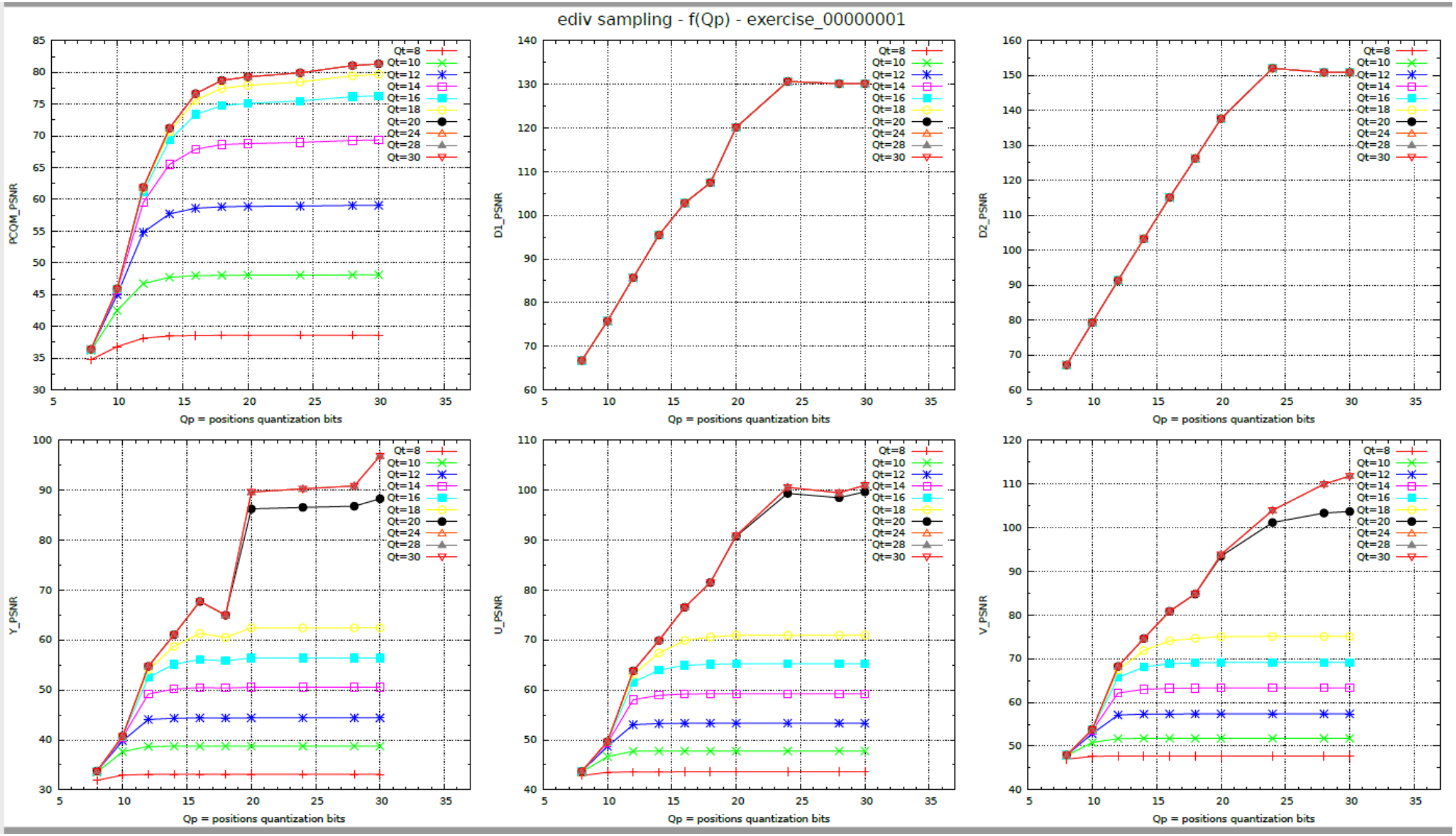
    cpv_basketball_player_00000001

    cpv_dancer_00000001

    cpv_exercise_00000001

    cpv_model_00000001

Lossy and near lossless metric - excerpt



Lossy and near lossless metric - assesment

- These preliminary results are available as accompanying pdf:
  - charts-derror.pdf
  - charts-merror.pdf

- We decided to keep the merror software over the derror one
  - Merror provides all the sampling modes
  - Merror proves to be faster to execute
  - Merror provides additional metrics for lossless and near lossless

- At this point we decided to discard the map sampling
  - Motivation from the fact that map/uvs distortions introduced impact on D1/D2 : which was not desired.

Lossy and near lossless metric - command

```
mm.exe  \
        reindex --sort oriented –I reference.obj -o ID:ref_reordered END \
        sample --mode ediv --bilinear --lengthThreshold $THRES -i ID:ref_reordered --hideProgress -o ID:ref END \
        reindex --sort oriented -i degraded.obj -o ID:deg_reordered END \
        sample --mode ediv  --bilinear --lengthThreshold $THRES -i ID:deg_reordered --hideProgress -o ID:deg END \
        compare --mode pcqm --inputModelA ID:ref --inputModelB ID:deg \
              --inputMapA reference.png  --inputMapB distorted.png END \
        compare --mode pcc  --resolution 1024 --inputModelA ID:ref --inputModelB ID:deg \
               --inputMapA reference.png  --inputMapB distorted.png
```

- Each model (reference and distorted) are reordered and than sampled
  - “ediv” mode is given as an example (next slides provides available modes)
  - “--mode ediv --lengthThreshold $THRES” value are specific for each model (see next slide)
  - Whatever the sampling mode one must set the **–bilinear** option
- Then reference and distorted are compared using PCC and PCQM
  - We use a fixed resolution of 1024 for pcc in order to get homogenous PSNRs
  - InputModelA must be the reference model

Lossy and near lossless metric – PCQM & PCC

- The output values of PCC mode are
  - D1 (p2point), D2 (p2plane), Y, U, V PSNRs
- The output of PCQM is a single PSNR
  - The single value is perceptually correlated
  - It is obtained by linear combination of 3 main features extracted from a set 8 features (can be dumped by merror)
    - F1 is Curvature comparison
    - F2 is Curvature contrast
    - **F3 is Curvature structure**
    - **F4 is Lightness comparison**
    - F5 is Lightness contrast
    - **F6 is Lightness structure**
    - F7 is Chroma comparison
    - F8 is Hue comparison
  - The coefficient were learned from the features and a large experimental perception test.
  - If needed these coefficient could be re-learned for our context.

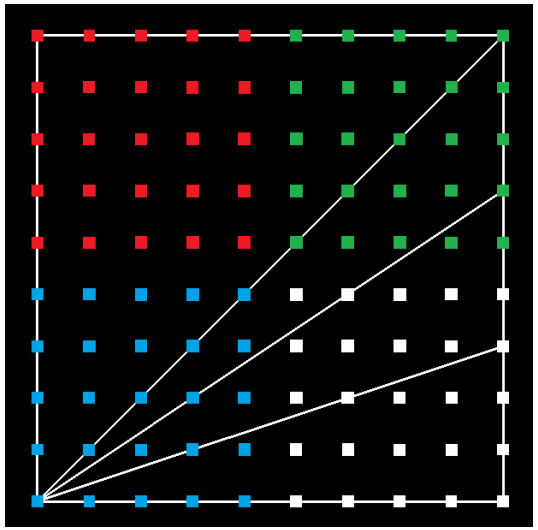
Lossy and near lossless metric - parameters

- 4K map => 16MPixels and 2K map =>4MPixels
- We would like to collect most sample from the map in affordable execution time, we thus chose 4M samples
- Parameters automatically found in order to obtain nbSamples close to 4000000, computed on first frame

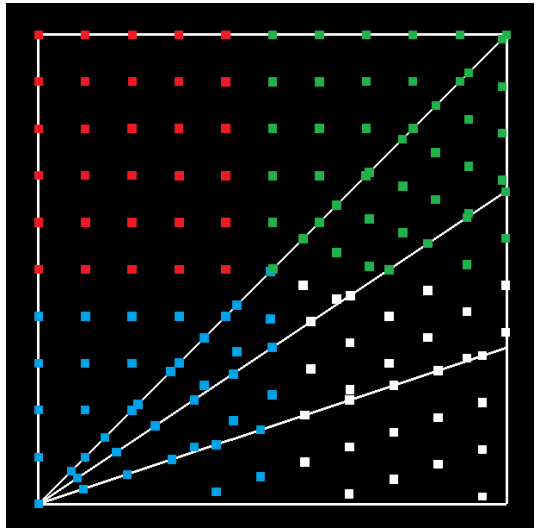
Model	--mode ediv --lengthThreshold	ediv nbSamples	--mode sdiv --areaThreshold	Sdiv nbSamples	--mode face --resolution	Face nbSamples	--mode grid --resolution	Grid nbSamples
Basket	1.04980469	4000905	1.25830078	4000560	1941	3997943	2055	4000826
Dancer	0.985595703	4000091	1.11791992	4000072	2010	3997339	2133	3997828
Exercise	0.945678711	4000817	1.04125977	4000428	2013	4000824	2120	3999311
Model	0.990356445	4000065	1.08349609	4000899	1923	3999930	2095	3998281
Axeguy	0.000340637984	4010236	1.48976113e-07	3999984	1614	3992779	3555	3998859
Thomas	1.03942871	4000996	1.15917969	4000226	1903	3997411	2364	4000646
Mitch	1.09912109	4000946	1.25854492	4000458	1787	3996524	2437	3997997
Longdress	0.515014648	4001293	0.336791992	4000971	2055	3999887	2083	4000007
Loot	0.524291992	4000440	0.345581055	4000127	1993	4000669	2041	4000522
Redandblack	0.508178711	4001080	0.335327148	4000721	2033	3998653	2065	3999814
Soldier	0.599487305	3999692	0.469360352	4000232	1784	3997938	1777	3997820
Football	0.0010175705	4000719	1.40117481e-06	4000914	1845	4000304	2086	3997807

Sampling comparison on non regular mesh

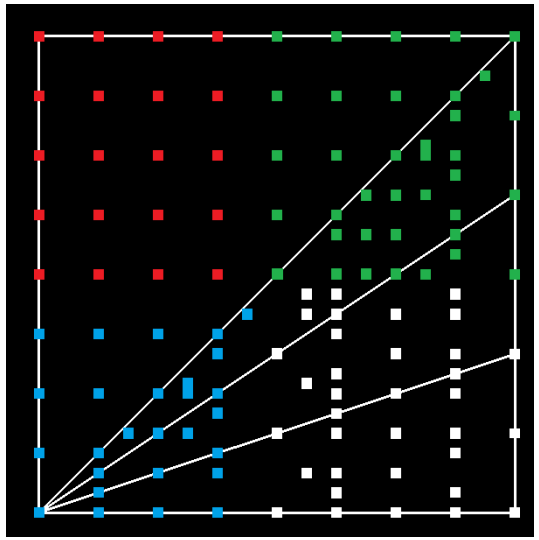
Grid 10



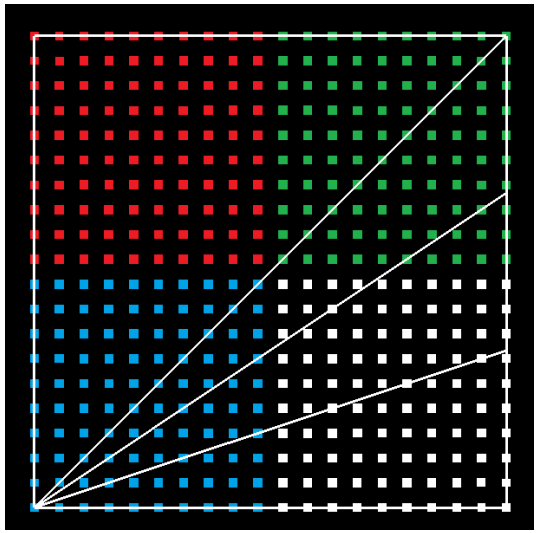
Face 10



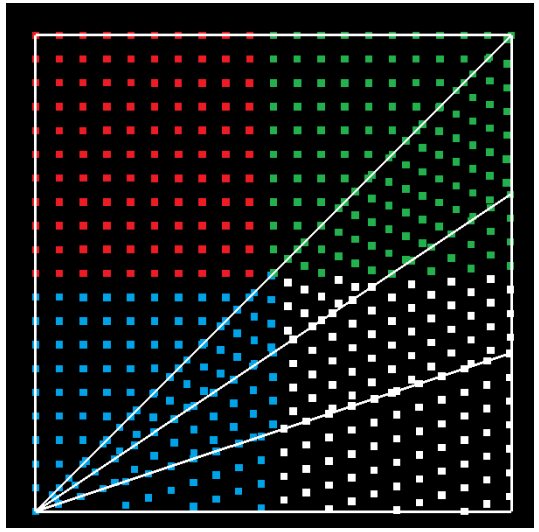
Ediv 10



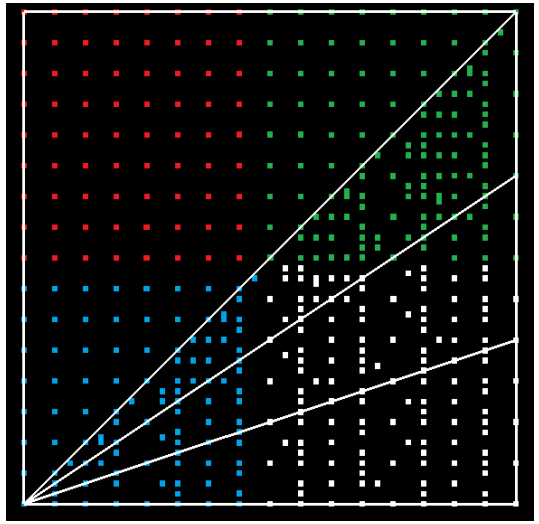
Grid 20



Face 20



Ediv 20

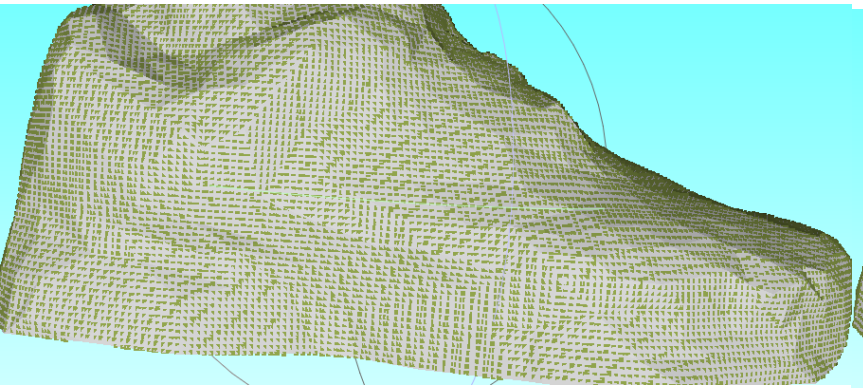


Content available here: https://mpeg.expert/live/nextcloud/index.php/s/e7e8tLN4ZEHBrAW

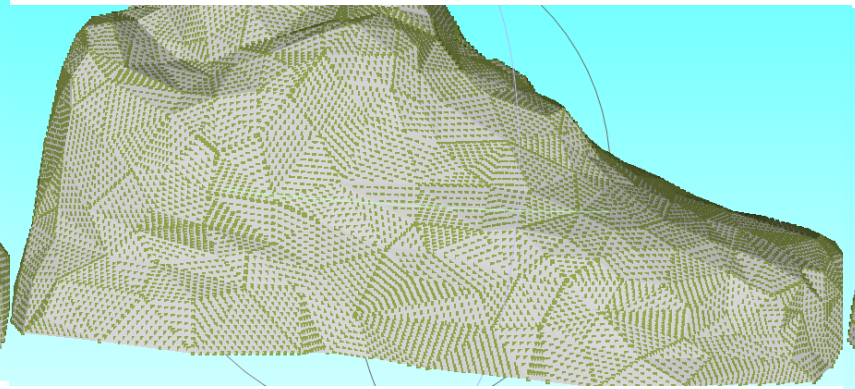
Content available here: https://mpeg.expert/live/nextcloud/index.php/s/e7e8tLN4ZEHBrAW

Same test on basket player

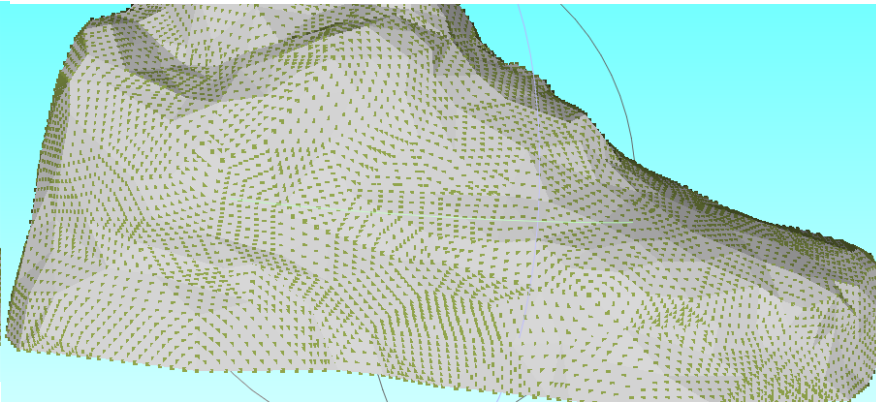
Grid 1M samples



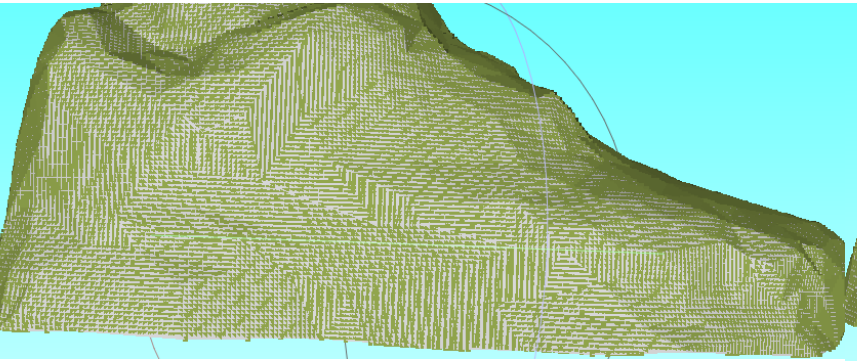
Face 1M samples



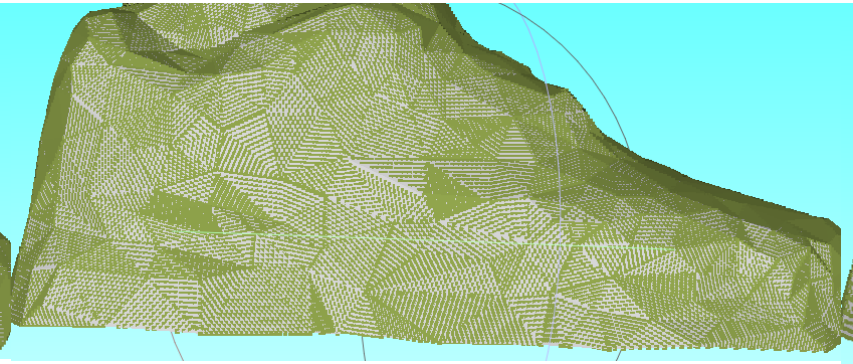
Ediv 1M samples



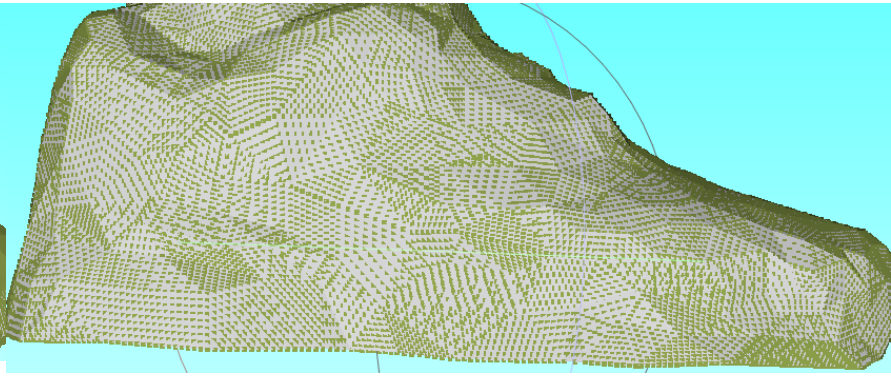
Grid 2M samples



Face 2M samples



Ediv 2M samples



Sampling choice - recommendation

- Ediv is very sensitive to edge length
  - hence minor modification of a vertex might lead to locally double the sampling density
  - Results are not regular at all
- Sdiv quite similar issues to Ediv ones, result not regular.
- Face is not regular and might miss the edges
- Grid is regular but might miss the edges

$\Rightarrow$ Grid + original vertices + edge sampling might be a good choice

$\Rightarrow$ Need to re-check Interdigital and Sony grid modes

Lossy and near lossless metric – stability

PSNR	Windows	Linux	Abs(diff)
Pcc_basket_qp8	67.6032562	67.6038208	0,0007646
Pcc_basket_qp16	115.398285	115.3321	0,066185
Pcc_basket_qp16_nomap	115.398285	115.3321	0,066185
Pcc_sphere_qp8	66.4296112	66.4646606	0,0350494
Pcqm_basket_qp8	36.899058	36.899058	0
Pcqm_basket_qp16	46.7802411	46.7802411	0
Pcqm_basket_qp16_nomap	68.2679284	68.2679284	0
Pcqm_sphere_qp8	Inf	Inf	0

Similar sampling parameters were used for all models and all modes (See test command in git merror repository)

Recommendation: exhaustive test with all pre-quantized models shall be envisioned

Near lossless topology specifics - proposal

- Constraint: model attributes are all indexed with a single index table
  - Any model can be transformed to this representation (if needed merror does the job)

- The encoder (or decoder) must export 2 topology tables in text files where:
  - We use one line for each triangle pair as follows (space separated, fast to parse)
    - 8 458
    - which tells that the decoded triangle of index 8 matches the original triangle of index 458
  - We use one line for each vertex pair as follows (space separated, fast to parse)
    - 0 23
    - which tells that the decoded vertex of index 0 matches the original vertex of index 23
  1. Then, the command “compare –mode topo” will load the table, the original and the decoded models in order to:
     - Test if number of triangles of output matches input number of triangles
     - Test is the proposed association table is bijective
     - Test if each output triangle respects the orientation of its associated input triangle
  2. Finally, we run the lossy metric
     - and obtain a metric result that considers the same features as for the lossy chain.

Near lossless metric - command

```
mm.exe  \
        compare --mode topo --inputModelA reference.obj --inputModelB degraded.obj \
          --faceMapFile topo_face.txt --vertexMapFile topo_vert.txt END \
        reindex --sort oriented –I reference.obj -o ID:ref_reordered END \
        sample --mode ediv --bilinear --lengthThreshold $THRES -i ID:ref_reordered --hideProgress -o ID:ref END \
        reindex --sort oriented -i degraded.obj -o ID:deg_reordered END \
        sample --mode ediv -- bilinear --lengthThreshold $THRES -i ID:deg_reordered --hideProgress -o ID:deg END \
        compare --mode pcqm --inputModelA ID:ref --inputModelB ID:deg \
            --inputMapA reference.png  --inputMapB distorted.png END \
        compare --mode pcc  --resolution 1024 --inputModelA ID:ref --inputModelB ID:deg \
           --inputMapA reference.png  --inputMapB distorted.png
```

- First compare subcommand will test the topology
- Other sub commands and parameters are the same as for the lossy metric
  - “ediv” mode is given as an example

Lossless metric - command

```
mm.exe  \
   compare --mode equ --earlyReturn \
      --inputModelA reference.obj –inputModelB degraded.obj \
      --inputMapA reference.png  --inputMapB distorted.png
```

- Mesh and texture are tested separately
- The command tests if the meshes are equal or not
  - It validates that every triangle of A as an equivalent in B in terms of vertex positions and attributes
  - The equality of triangles is computed at a vertex shift difference (oriented option)
    - For instance, A B C and B C A are equivalent, but A B C and A C B are not
- The command tests if the texture maps are equal or not,
  - The metric returns the number of different pixels if any.
  - It is a raw test, the eventual padding and inter patch are also compared
- --earlyReturn is for fast answer, remove option for detailed results (slower).

Remaining work

- Could compute sampling parameters on full sequence basis instead of first frame
  - If so, shall do it on reference quantized sequences (whenever available)
- Need to re-run all the tests (the charts function of distortions) for lossy metric:
  - Using latest software available on merror master,
  - with anchor pre-quantized models (whenever available) and provided parameters.
  - To check the metric stability on these reference quantized models
  - To validate that ‘ediv’ is the best sampling approach (even if not perfect)
- Need to extend numerical stability tests (platform dependence)
  - On reference quantized models
- Finalize testing/validation of near lossless topology check
  - Available on mmetric stabilization branch - to be merged into the master
  - Need to check how to introduce distance error threshold in the metric (Khaled, Danillo, J.E.)
- Finalize testing/validation of lossless metric (available on master)
  - Test in anchor chain
- Finalize MacOS version of the cmake file

EE 4.1 Mandates

The mandate of this exploratory experiment is to:

- Select a mean of summarizing the metric for a set of mesh frames.
  - Mean, deviation, variance and Minkowsky(ms=3) mean are automatically computed
- Propose an error measure software for lossless, near lossless coding (see definition in [6]).
- Propose a similarity checking software for lossless coding (see definition in [6])
- Propose a software implementation based on both contributions (Interdigital or Sony).
- Investigate the best sampling method (might be linked to the metric choice)
- Investigate if we keep both for pcqm[5] and pcc_error [1] or select only one for the CfP.