m57390
EE4.1 mesh metric software v0.1.11 updates and review

Jean-Eudes Marvie (Interdigital)

Animated meshes with animated textures

Introduction

- The Mesh Metric and Tools software v0.1.11 is avaliable at:
  - http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-mmetric/-/tree/0_1_11
- Original purpose is to provide mesh metrics:
  - lossless, near-lossless, lossy
  - Also provides some tools we used to validate/test the metric
  - Also provides some tools for anchor content preparation

- Please refer to m56760 for an introduction and overview

- The software comes as a command line program
  - Each tool is implemented as a “sub-command”,
  - Those are named “command” in the rest of the presentation.

In this presentations we will present the changes since meeting 134

Changes since meeting 134

Version 0.1.10

Version 0.1.11

Version 0.1.10, delivered 2021-06-09

- Add: some ibsm test output validation
- Add: image-based sampling metric (compare --mode ibsm) and tests
- Add: software and hardware renderer
- Add: outputCsv option to compare pcc and pcqm
- Add: pcc hausdorff option test
- Add: meshlab projects in test folder
- Add: extention of grid sampling
  - add options to use sequence bounding box,
  - add useNormal option to filter sampling directions
  - Contributors: Danillo Graziosi (Sony), J.E. Marvie (InterDigital)
- Add: pseudo random sampling (sample --mode prand)
  - Contributors: Jungsun Kim (Apple), Khaled Mammou (Apple).
    - Support for color per vertex not provided
    - No experimental results provided for validation
  - unit tests added by jean-eudes marvie (InterDigital)
- Fix: frSize overflow in _loadImageFromVideo
- Fix: nearest and bilinear filtering
- Fix: renaming renderer and compare ibsm parameters
  - rename "compare --mode raster" as "compare --mode ibsm"
    - ibsm: Image Based Sampling Metric
  - rename "render --mode" into "render --renderer"
  - realigned parameter names from on command to the other
  - please read carefully the documentation to update your scripts

Version 0.1.11, delivered 2021-07-06

- Add: ibsm automatic reindex and tests update.
- Add: Color Space Conversion for Qunatize/Dequantize functions
  - Contributor: Danillo Graziosi (Sony)
- Add: sw renderer/ibsm metric optimization, gains 0.4sec on 4K renders.
- Add: degrade --nbFaces option
- Fix: use max depth sw render value (fix frederick metric infinite issue).
- Fix: ibsm MSE and PSNR processing use mask Size instead of image size
- Fix: attribute names masking in compare. renamed class attributes with '_' prefix.

Commands

Used to validate/test the different metrics and generate the anchor contents

- quantize/dequantize
- degrade

Quantize/dequantize commands

Used for anchor content generation

```
Quantize model (mesh or point cloud)
Usage:
  mm.exe quantize [OPTION...]

  -i, --inputModel arg        path to input model (obj or ply file)
  -o, --outputModel arg       path to output model (obj or ply file)
  -h, --help                  Print usage
      --dequantize            set to process dequantification at the ouput
      --qp arg                Geometry quantization bitdepth. A value < 7
                              means no quantization. (default: 12)
      --qt arg                UV coordinates quantization bitdepth.  A value
                              < 7 means no quantization. (default: 12)
      --qn arg                Normals quantization bitdepth. A value < 7 no
                              quantization. (default: 12)
      --qc arg                Colors quantization bitdepth. A value < 7 no
                              quantization. (default: 8)
      --minPos arg            min corner of vertex position bbox, a string of
                              three floats. Computed of not set.
      --maxPos arg            max corner of vertex position bbox, a string of
                              three floats. Computed of not set.
      --minUv arg             min corner of vertex texture coordinates bbox,
                              a string of three floats. Computed of not set.
      --maxUv arg             max corner of vertex texture coordinates bbox,
                              a string of three floats. Computed of not set.
      --minNrm arg            min corner of vertex normal bbox, a string of
                              three floats. Computed of not set.
      --maxNrm arg            max corner of vertex normal bbox, a string of
                              three floats. Computed of not set.
      --minCol arg            min corner of vertex color bbox, a string of
                              three floats. Computed of not set.
      --maxCol arg            max corner of vertex color bbox, a string of
                              three floats. Computed of not set.
      --outputVar arg         path to the output variables file.
      --useFixedPoint         internally convert the minPos and maxPos to
                              fixed point 16.
      --colorSpaceConversion  Convert color space from RGB to YUV.
```

NEW

```
Dequantize model (mesh or point cloud)
Usage:
  mm.exe dequantize [OPTION...]

  -i, --inputModel arg        path to input model (obj or ply file)
  -o, --outputModel arg       path to output model (obj or ply file)
  -h, --help                  Print usage
      --qp arg                Geometry quantization bitdepth. No
                              dequantization of geometry if not set or < 7.
      --qt arg                UV coordinates quantization bitdepth. No
                              dequantization of uv coordinates if not set or < 7.
      --qn arg                Normals quantization bitdepth. No
                              dequantization of normals if not set or < 7.
      --qc arg                Colors quantization bitdepth. No dequantization
                              of colors if not set or < 7.
      --minPos arg            min corner of vertex position bbox, a string of
                              three floats. Mandatory if qp set and >= 7
      --maxPos arg            max corner of vertex position bbox, a string of
                              three floats. Mandatory if qp set and >= 7
      --minUv arg             min corner of vertex texture coordinates bbox,
                              a string of three floats. Mandatory if qt set
                              and >= 7
      --maxUv arg             max corner of vertex texture coordinates bbox,
                              a string of three floats. Mandatory if qt set
                              and >= 7
      --minNrm arg            min corner of vertex normal bbox, a string of
                              three floats. Mandatory if qn set and >= 7.
      --maxNrm arg            max corner of vertex normal bbox, a string of
                              three floats. Mandatory if qn set and >= 7
      --minCol arg            min corner of vertex colors bbox, a string of
                              three floats. Mandatory if qc set and >= 7
      --maxCol arg            max corner of vertex colors bbox, a string of
                              three floats. Mandatory if qc set and >= 7
      --useFixedPoint         interprets minPos and maxPos inputs as fixed
                              point 16.
      --colorSpaceConversion  Convert color space from YUV to RGB.
```

Color per vertex are first transformed from RGB to YUV, quantized YUV is written in output file, and the invert for deq.

Degrade command

Used for metric validation

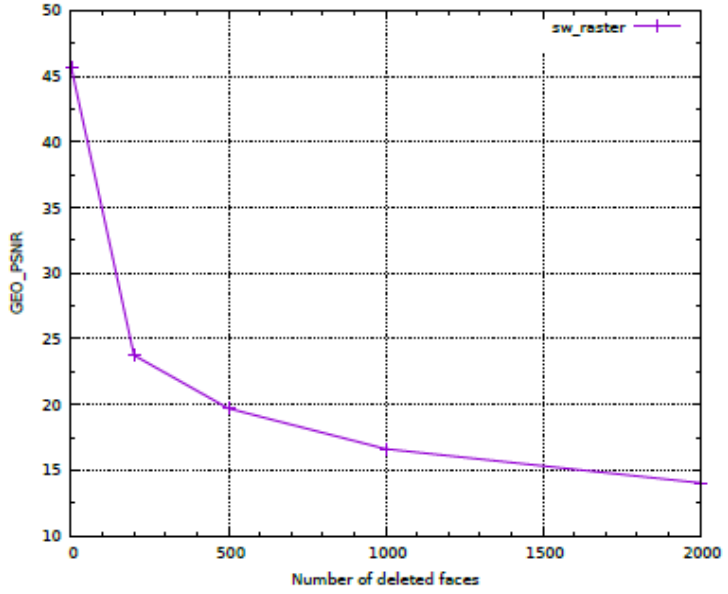
Set target number of evenly deleted faces, easier and more precise than –nthFaces.

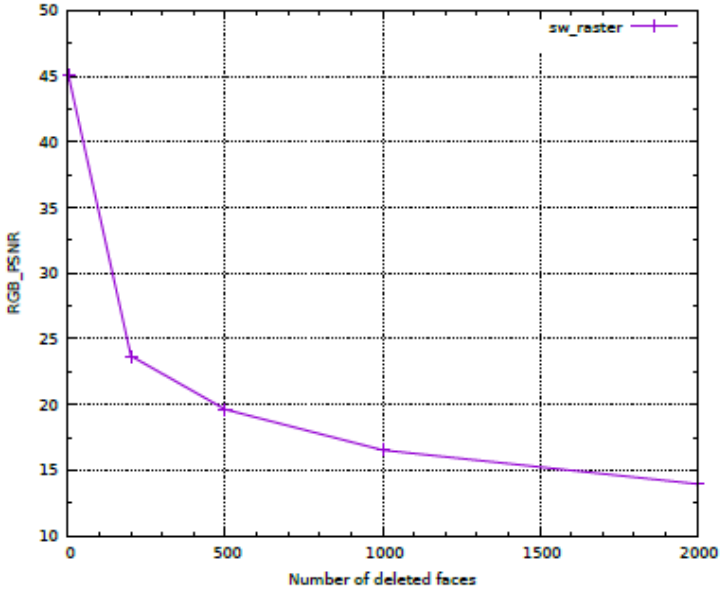
```
Degrade a mesh (todo points)
Usage:
  mm.exe degrade [OPTION...]

  -i, --inputModel arg   path to input model (obj or ply file)
  -o, --outputModel arg  path to output model (obj or ply file)
      --mode arg         the sampling mode in [delface]
      --nthFace arg      in delface mode, remove one face every nthFace.
                         (default: 50)
      --nbFaces arg      in delface mode, if nthFace==0, remove nbFaces.
                         (default: 0)
  -h, --help             Print usage
```

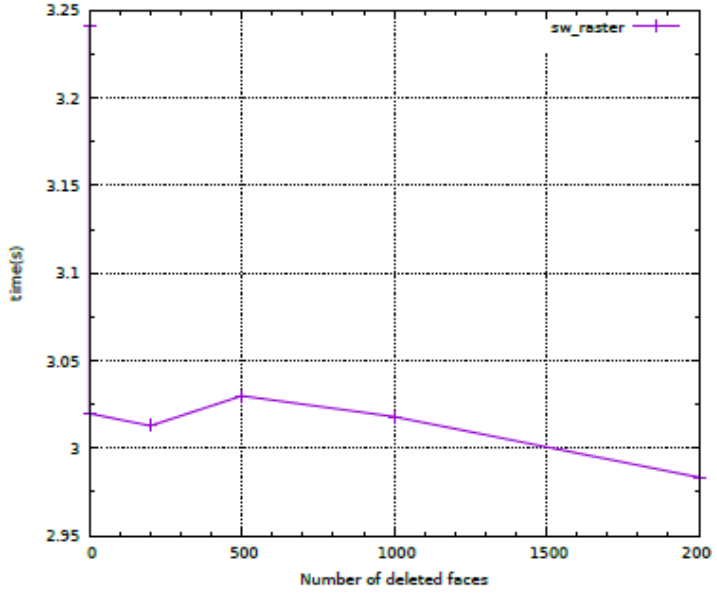
NEW

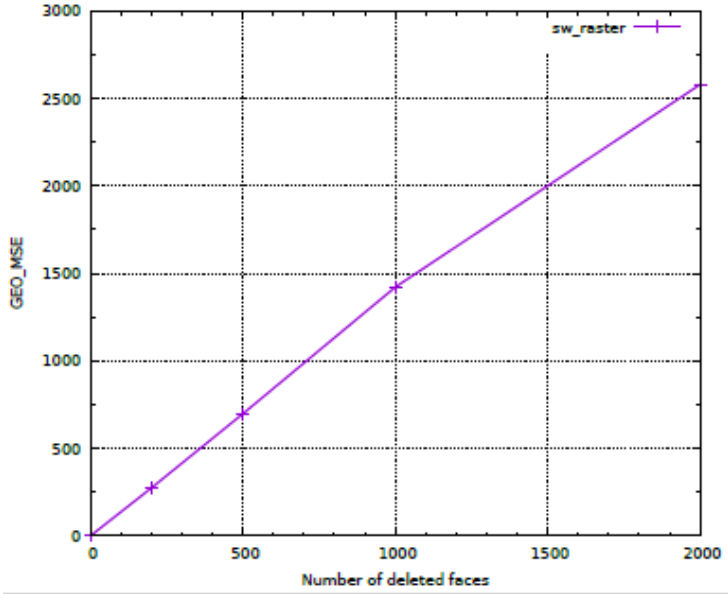
Degrade – example use in metric assessment

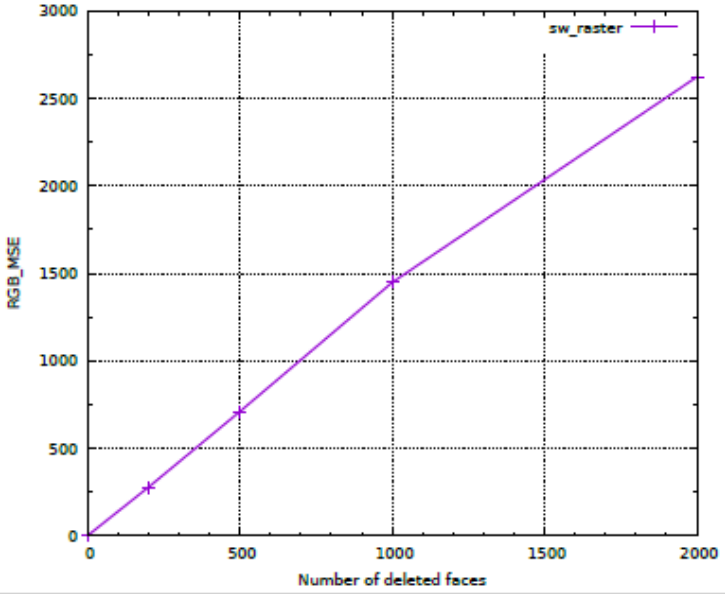
ibsm = f(local artefacts) - basketball_player_fr0001











num	GEO_PSNR	RGB_PSNR	GEO_MSE	RGB_MSE	time(s)
2000	14.0172	13.9429	2578.4751	2622.9287	2.9833
1000	16.6052	16.5236	1420.9045	1447.8259	3.0180
500	19.6940	19.6174	697.7118	710.1384	3.0299
200	23.7453	23.6914	274.5049	277.9338	3.0128
1	45.6393	45.1065	1.7748	2.0065	3.0198
0	inf	inf	0.0000	0.0000	3.2411

Commands

Used to evaluate the different metrics

- sample
- compare

Sample command

```
Convert mesh to point cloud
Usage:
  mm.exe sample [OPTION...]

  -i, --inputModel arg   path to input model (obj or ply file)
  -m, --inputMap arg     path to input texture map (png, jpg, rgb, yuv)
  -o, --outputModel arg  path to output model (obj or ply file)
      --mode arg         the sampling mode in [face,grid,map,sdiv,ediv,prnd]
      --hideProgress     hide progress display in console for use by robot
      --outputCsv arg    filename of the file where per frame statistics will
                         append. (default: )
  -h, --help             Print usage

 ediv mode options:
      --lengthThreshold arg  edge length limit to stop subdivision, used only
                             if > 0, otherwise resolution is used. (default:
                             0.0)

 face and ediv modes options:
      --resolution arg  integer value in [1,maxuint], step/edgeLength =
                        resolution / size(largest bbox side). In ediv mode, the
                        resolution is used only if lengthThreshold=0. (default:
                        1024)

 face mode options:
      --float          if set the processings and outputs will be float32,
                       int32 otherwise (default: true)
      --thickness arg  floating point value, distance to border of the face
                       (default: 0.0)

 grid mode options:
      --gridSize arg   integer value in [1,maxint], side size of the grid
                       (default: 1024)
      --useNormal      if set will sample only in the direction with the
                       largest dot product with the triangle normal
```

NEW

Add pseudo-random (PRND) sampling method

- Easy to set target number of samples
- Sampling not uniform, aggregates
- Missing support for color per vertex
- No complete validation done with metrics

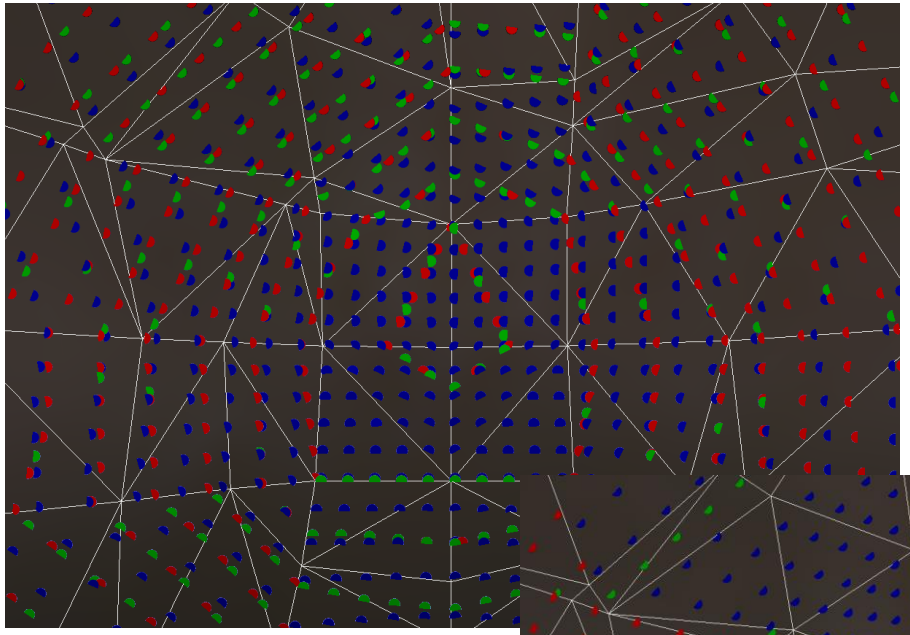
```
      --minPos arg     min corner of vertex position bbox, a string of three
                       floats.
      --maxPos arg     max corner of vertex position bbox, a string of three
                       floats.
      --useFixedPoint  interprets minPos and maxPos inputs as fixed point 16.

 grid, face, sdiv and ediv modes. options:
      --bilinear            if set, texture filtering will be bilinear,
                            nearest otherwise
      --nbSamplesMin arg    if set different from 0, the system will rerun the
                            sampling multiple times to find the best parameter
                            producing a number of samples in [nbAmplesMin,
                            nbSamplesMax]. This process is very time comsuming.
                            (default: 0)
      --nbSamplesMax arg    see --nbSamplesMin documentation. Must be > to
                            --nbSamplesMin. (default: 0)
      --maxIterations arg   Maximum number of iterations in sample count
                            constrained sampling, i.e. when --nbSampleMin > 0.
                            (default: 10)

 prnd mode options:
      --nbSamples arg  integer value specifying the traget number of points
                       in the output point cloud (default: 2000000)

 sdiv mode options:
      --areaThreshold arg  face area limit to stop subdivision (default: 1.0)
      --mapThreshold       if set will refine until face vertices texels are
                           distanced of 1 and areaThreshold reached
```

NEW

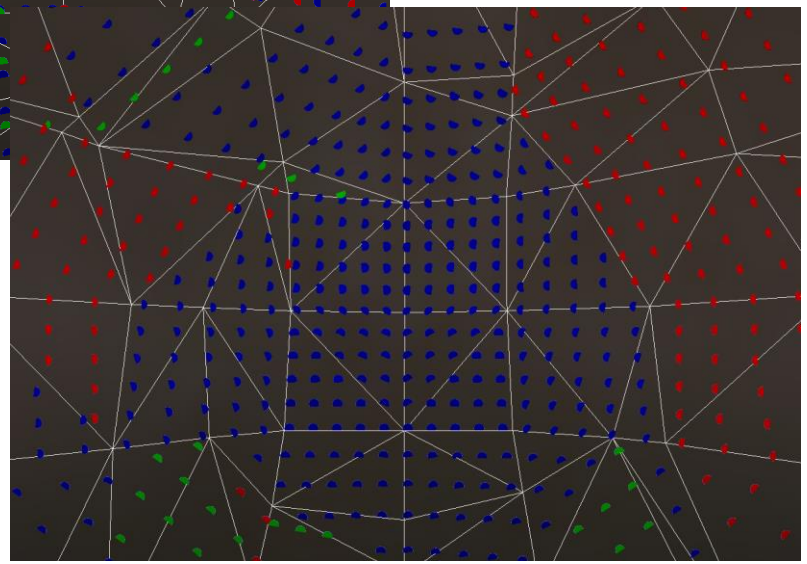
Sample command

```
Convert mesh to point cloud
Usage:
  mm.exe sample [OPTION...]

  -i, --inputModel arg   path to input model (obj or ply file)
  -m, --inputMap arg     path to input texture map (png, jpg, rgb, yuv)
  -o, --outputModel arg  path to output model (obj or ply file)
      --mode arg         the sampling mode in [face,grid,map,sdiv,ediv,prnd]
      --hideProgress     hide progress display in console for use by robot
      --outputCsv arg    filename of the file where per frame statistics will
                         append. (default: )
  -h, --help             Print usage

 ediv mode options:
      --lengthThreshold arg  edge length limit to stop subdivision, used only
                             if > 0, otherwise resolution is used. (default:
                             0.0)

 face and ediv modes options:
      --resolution arg  integer value in [1,maxuint], step/edgeLength =
                        resolution / size(largest bbox side). In ediv mode, the
                        resolution is used only if lengthThreshold=0. (default:
                        1024)

 face mode options:
      --float          if set the processings and outputs will be float32,
                       int32 otherwise (default: true)
      --thickness arg  floating point value, distance to border of the face
                       (default: 0.0)

 grid mode options:
      --gridSize arg   integer value in [1,maxint], side size of the grid
                       (default: 1024)
      --useNormal      if set will sample only in the direction with the
                       largest dot product with the triangle normal
```

NEW



off



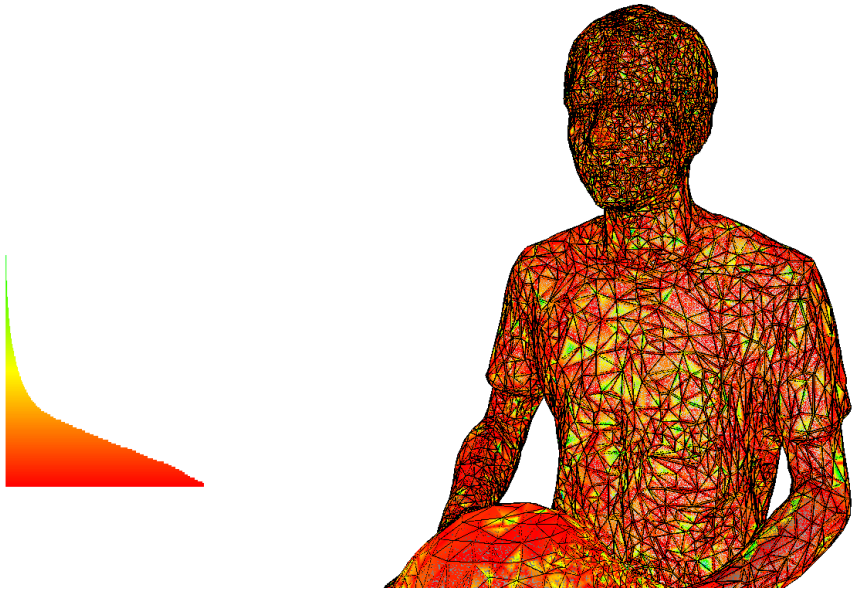
on

Add filtering of samples based on normal in GRID mode

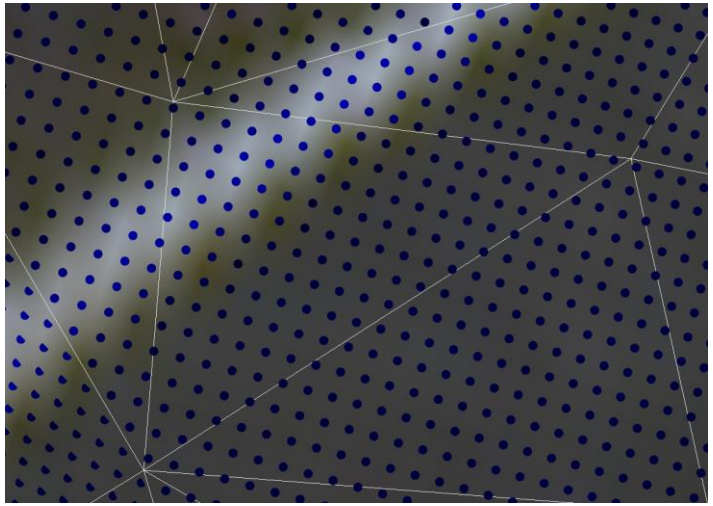
- No complete validation done with metrics
- Implies to reprocess the --gridSize for each anchor model in order to reach the 4M points

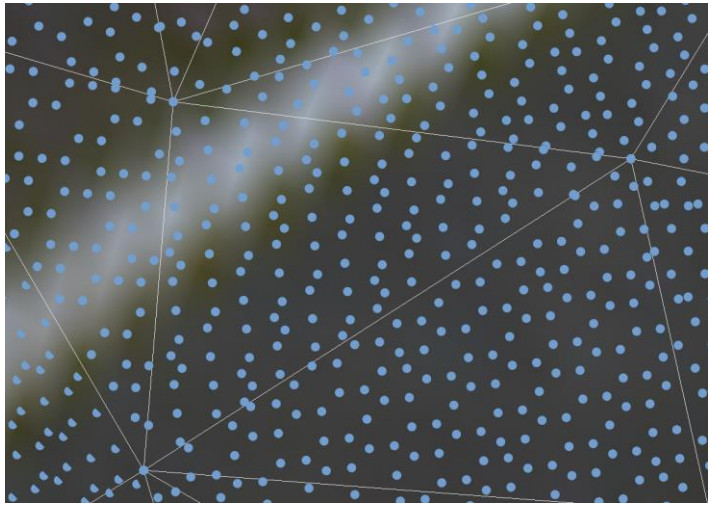
Sampling results

- USAMP has issues with the sample positions when the connectivity is modified

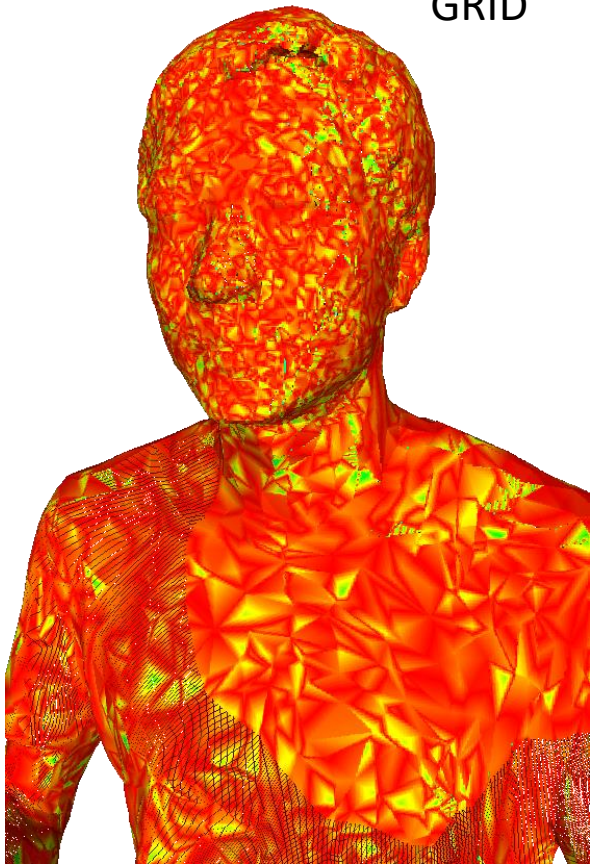


Mesh quality using Hausdorff distance (MESHLAB)

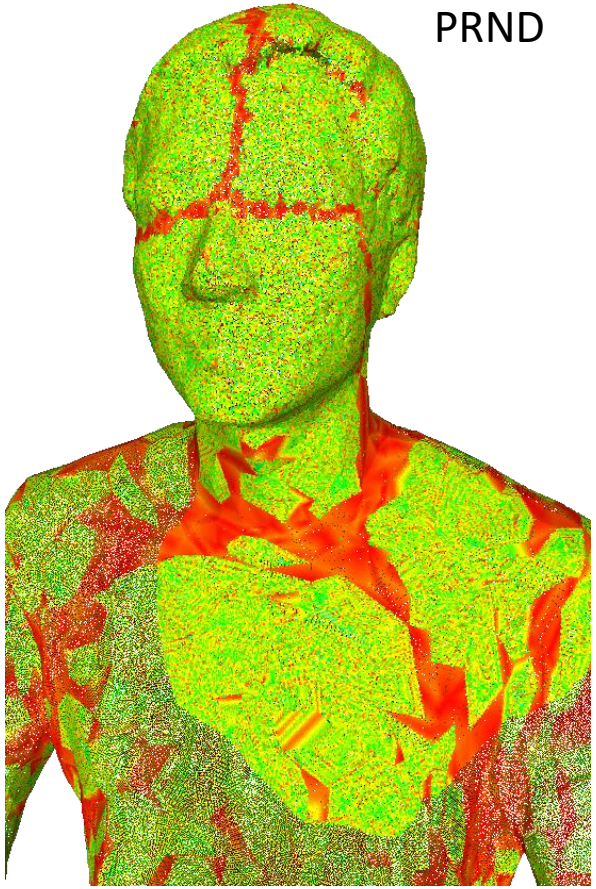




GRID



PRND



Compare command

```
Compare model A vs model B
Usage:
  mm.exe compare [OPTION...]

      --inputModelA arg   path to reference input model (obj or ply file)
      --inputModelB arg   path to distorted input model (obj or ply file)
      --inputMapA arg     path to reference input texture map (png, jpg, rgb,
                          yuv)
      --inputMapB arg     path to distorted input texture map (png, jpg, rgb,
                          yuv)
      --outputModelA arg  path to output model A (obj or ply file)
      --outputModelB arg  path to output model B (obj or ply file)
      --outputCsv arg     filename of the file where per frame statistics
                          will append. (default: )
      --mode arg          the comparison mode in [equ,pcc,pcqm,topo,ibsm]
                          (default: equ)
  -h, --help              Print usage
```

NEW

```
ibsm mode options:
     --ibsmResolution arg     Resolution of the image buffer. (default:
                              2048)
     --ibsmCameraCount arg    Number of virtual cameras to be used per
                              frame. (default: 16)
     --ibsmRenderer arg       Use software or openGL 1.2 renderer. Value in
                              [sw_raster, gl12_raster]. (default: sw_raster)
     --ibsmDisableCulling     Set option to disable the backface culling.
     --ibsmDisableReordering  Set option to disable automatic oriented
                              reordering of input meshes, can be usefull if
                              already done previously to save very small execution
                              time.
     --ibsmOutputPrefix arg   Set option with a proper prefix/path system to
                              dump the color shots as png images (Warning,
                              it is extremly time consuming to write the
                              buffers, use only for debug).
```

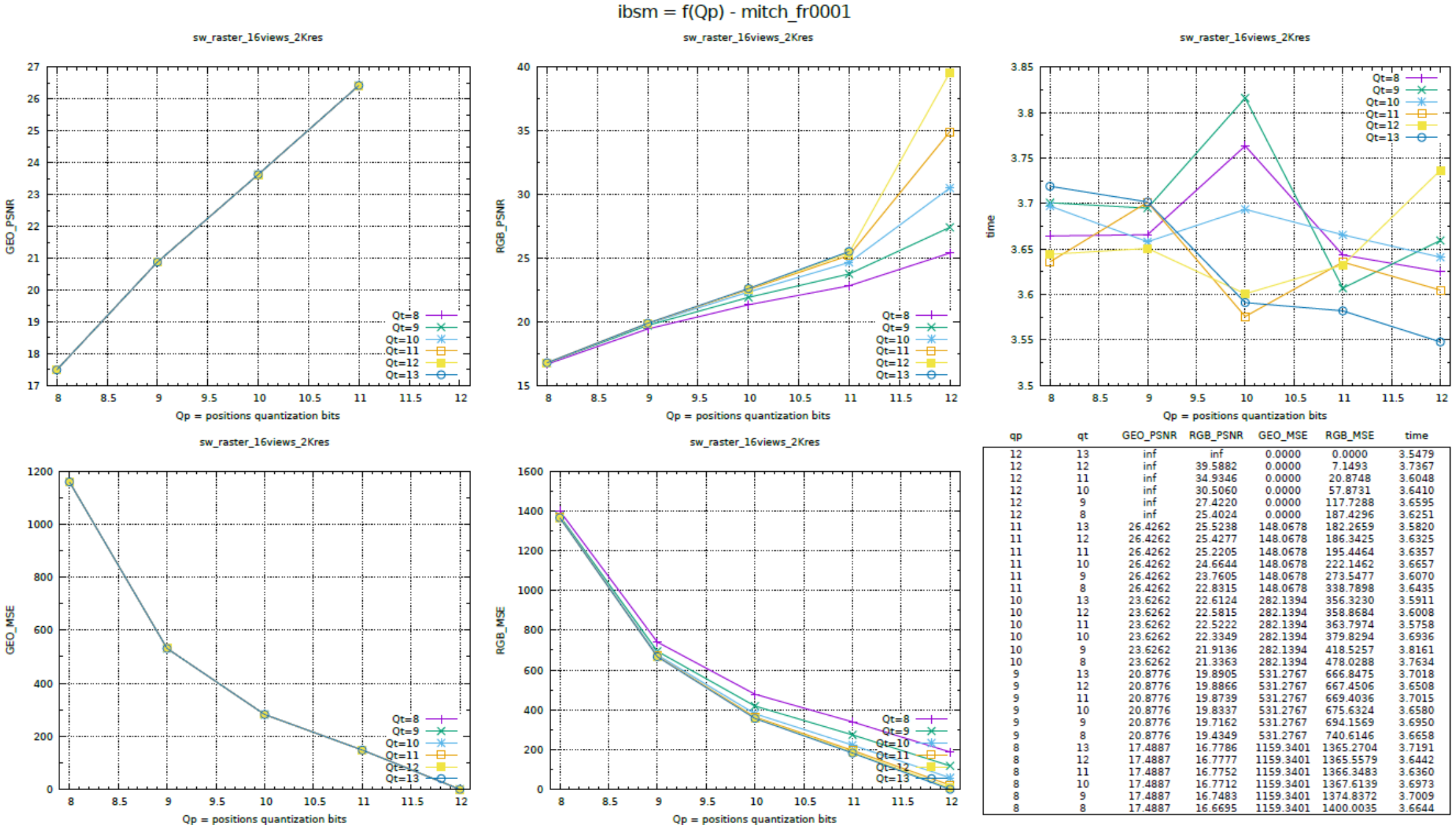
Add the new image-based sampling metric (IBSM)

- For meshes only,
- Does not need pre-sampling
- For lossy and near-lossless
- Fully supports textured meshes
- Fully supports color per vertex meshes
- It internally reindex the meshes for numerical stability
- Metric is symmetric, A or B can be reference.
- One line command, easy to use
- Default parameters suitable for MPEG V-Mesh

See m57389 [EE_4_1][image_based_mesh_metric] for details and complete validation on anchor contents

Example invocation and results

```
/> mm compare --mode ibsm -inputModelA ref.obj -inputModelB dist.obj -inputMapA ref.png -inputModelB dist.png
```



qp	qt	GEO_PSNR	RGB_PSNR	GEO_MSE	RGB_MSE	time
12	13	inf	inf	0.0000	0.0000	3.5479
12	12	inf	39.5882	0.0000	7.1493	3.7367
12	11	inf	34.9346	0.0000	20.8748	3.6048
12	10	inf	30.5060	0.0000	57.8731	3.6410
12	9	inf	27.4220	0.0000	117.7288	3.6595
12	8	inf	25.4024	0.0000	187.4296	3.6251
11	13	26.4262	25.5238	148.0678	182.2659	3.5820
11	12	26.4262	25.4277	148.0678	186.3425	3.6325
11	11	26.4262	25.2205	148.0678	195.4464	3.6357
11	10	26.4262	24.6644	148.0678	222.1462	3.6657
11	9	26.4262	23.7605	148.0678	273.5477	3.6070
11	8	26.4262	22.8315	148.0678	338.7898	3.6435
10	13	23.6262	22.6124	282.1394	356.3230	3.5911
10	12	23.6262	22.5815	282.1394	358.8684	3.6008
10	11	23.6262	22.5222	282.1394	363.7974	3.5758
10	10	23.6262	22.3349	282.1394	379.8294	3.6936
10	9	23.6262	21.9136	282.1394	418.5257	3.8161
10	8	23.6262	21.3363	282.1394	478.0288	3.7634
9	13	20.8776	19.8905	531.2767	666.8475	3.7018
9	12	20.8776	19.8866	531.2767	667.4506	3.6508
9	11	20.8776	19.8739	531.2767	669.4036	3.7015
9	10	20.8776	19.8337	531.2767	675.6324	3.6580
9	9	20.8776	19.7162	531.2767	694.1569	3.6950
9	8	20.8776	19.4349	531.2767	740.6146	3.6658
8	13	17.4887	16.7786	1159.3401	1365.2704	3.7191
8	12	17.4887	16.7777	1159.3401	1365.5579	3.6442
8	11	17.4887	16.7752	1159.3401	1366.3483	3.6360
8	10	17.4887	16.7712	1159.3401	1367.6139	3.6973
8	9	17.4887	16.7483	1159.3401	1374.8372	3.7009
8	8	17.4887	16.6695	1159.3401	1400.0035	3.6644

```
Render a mesh (todo points)
Usage:
  mm.exe render [OPTION...]

  -i, --inputModel arg     path to input model (obj or ply file)
  -m, --inputMap arg       path to input texture map (png, jpeg)
  -o, --outputImage arg    path to output image (png file) (default:
                           output.png)
      --outputDepth arg    path to output depth RGBA png file with 32bit
                           float span on the four components. If empty string will
                           not save depth (default behavior).
      --renderer arg       Use software or openGL 1.2 renderer. Value in
                           [sw_raster, gl12_raster]. (default: sw_raster)
      --hideProgress       hide progress display in console for use by robot
  -h, --help               Print usage
      --width arg          Output image width (default: 1980)
      --height arg         Output image height (default: 1024)
      --bilinear           Set --bilinear=false to disable bilinear filtering
                           (default: true)
      --viewDir arg        the view direction, a string of three floats
                           (default: 0.0 0.0 1.0)
      --viewUp arg         the view up vector, a string of three floats
                           (default: 0.0 1.0 0.0)
      --bboxMin arg        bbox min corner, a string of three floats. If not
                           both bboxMin and bboxMax set, will compute
                           automatically.
      --bboxMax arg        bbox max corner, a string of three floats. If not
                           both bboxMin and bboxMax set, will compute
                           automatically.
      --clearColor arg     background color, a string of four int8, each
                           component in [0,255]. (default: 0 0 0 0)
      --enableLighting     enable the lighting if model has normals.
      --autoLightPosition  enable light position computation from bounding
                           sphere radius and lightAutoDir parameter.
      --lightAutoDir arg   if autoLightPosition is set, will compute light
                           position as boundingSphereCenter + lightAutoDir *
                           boundingSphereRadius. Default value stands for top
                           right sector. (default: 1 1 1)
      --enableCulling      enable back face culling.
      --cwCulling          true sets Clock Wise (cw) orientation for face
                           culling, Counter Clock Wise (ccw) otherwise. (default:
                           true)
```

Render command

Add the new software and hardware renderer

- Used by the IBSM metric
- For meshes only,
- Generates canonical orthographic renderings
- With or without lighting
- With or without bilinear filtering
- With or without backface culling
    - With cw or ccw defined meshes

Examples of rendering

Light off

Light on
Top right

Questions ?