

MeshValmet 3.0

Christine Xu, xushun@cs.unc.edu

1 Description

MeshValmet is a tool that measures surface to surface distance between two triangle meshes using user-specified uniform sampling. Thus, users can always choose finer sampling level to calculate errors in order to gain more accuracy in the "error space", or sparser sampling to achieve speed requirement and get an approximate feeling of error distribution between boundaries.

Besides its pleasant visualization using VTK library, MeshValmet also provides useful histogram and statistical information, such as mean distance, root mean square distance, Hausdorff distance, 95 percentile, 68 percentile, etc..

MeshValmet is based on the work of Nicolas Aspert, etc.: <http://mesh.epfl.ch>. *MESH: Measuring Errors between Surfaces using the Hausdorff distance* in the proceedings of the *IEEE Int. Conf. on Multimedia and Expo 2002 (ICME)*, vol. I, pp. 705-708. Based on their code and paper, I made a lot of modification and added many new functionalities.

2 Features

- Use point to triangle closest point distance
- User specified uniform sampling
- Space subdivision algorithm, fast computation (avoid brute force)

- Can be run in a batch mode, pipelining of statistical analysis
- Compute the volume overlap of two meshes, as well as dice's coefficient
- Provide histogram and statistical information
- Provides GUI and 3D visualization of surface distances

3 Usage

From GUI:

- Step 1. First user needs to load in two triangle meshes. Currently RAW, BYU, and IV are the only three supported file format. Key "t" and "j" can switch between the "trackball" and "joystick" mode in two display windows.
- Step 2. Then specify sampling step and minimum sampling frequency to be used for sub-sample the meshes. Note that the sampling step is measured in the percentage of the bounding box of the second mesh.
- Step 3. Choose whether you want to measure the distance error from model A->B, or B->A, or both. This is due to the asymmetric property of Hausdorff distance.
- Step 4. Choose whether you want to compute signed distance error or absolute distance error, with positive distance indicating outward distance and negative distance for inward distance.
- Step 5. Click button 'Compute Error'. The corresponding model will be colormapped based on the errors of the sampling points. The slider between the two windows controls display of the sampling points.
- Step 6. On the lower left of the panel is Colormap Control, where users can change the colormap properties to achieve better visual effect and gain more intuition.
- Step 7. On the middle right of the panel is the Histogram Control, where users can change the display range of the histogram.

- Step 8. On the lower right of the panel is the Statistics Panel, where users can view the statistical results of the comparison.

From COMMAND line:

- Usage:

```
MeshValmet [options]
-p      <batchFile>
-s      <samplingStep>      (Default value = 0.5)
-f      <minSamplingFreq>  (Default value = 2)
-b      <numOfBins>         (Default value = 256)
-type   <num>              (use 0 to compute error of A <-> B,
                           use 1 to compute the error of A -> B,
                           use 2 to compute the error of B -> A.
                           Default value = 0)
-absolute
-in1    <inFile1>
-in2    <inFile2>
-o      <outFile>
```

Batch File Format:

```
<numOfFiles>
<inFile1>    <inFile1>  <outFile1>
<inFile2>    <inFile2>  <outFile2>
...
...
...
```

4 Menus and Settings

- File->Load Model A/B

Load 3D triangle meshes into display windows. Supported file formats are RAW, BYU, and IV.

- View->Switch BLK/WHT Background

Switch the background between color black and white in the display windows. Generally, a white background is more clear. But sometimes, especially when two objects are overlayed, it is better to view it in a black background.

- **View->Reset Cameras**

Reset the cameras in both display windows into the same original position.

- **View->Synchronize Viewport**

Synchronize/Unsynchronize the movement of the objects in the two windows. When you are in synchronization mode and you move/rotate/scale an object in one window, the other object in the other window will be moved/rotated/scaled the same way.

- **View->Overlay two views**

Overlay two objects to see the boundary differences. Object from the opposite window is represented as a yellow point distribution model.

- **View->Fill/Line/Point for Model1/2**

Change the visualization representation of the model and switch between solid fill, wireframe, and point representations.

- **Compute->Compute Error**

After loading both models, we can compute the closest point distance errors and compute the statistical information.

- **Compute->Update Histogram**

After changing the range (min, max) of either the x-axis or the y-axis in the Histogram Control panel, we can update the histogram in display.

- **Output->Save Model 1/2**

Output the content of the display window to an IV file.

- **Output->External TextWindow**

To output the compute information to an external text window rather than the console window.

- **Output->Output Histogram**

Output information of the computed histogram to an external file.

- **Output->Output Sample Errors**

Output all the sample errors to an external file.

- Output->Output Vertex Errors

Output all the vertex errors to an external file.

- Output->Output Statistical Infos

Output all the information in the Statistics panel to an external file.

- Properties->Sampling step(%)

Specify the sampling step used to sample the two input surfaces. Here sampling step is measured in the percentage of the bounding box of the second mesh model.

For each triangle, the number of samples is:

$$n_samples = \text{area} * \text{dens}, \text{dens} = 1 / \text{sqr}(\text{sampling_step} * m2bboxdiagonal).$$

- Properties->Minimum sampling frequency

Once we get the initial n_samples as calculated above for each triangle, we can determine the initial sampling frequency n by the relation that $n_samples = n * (n+1)/2$. Then we pick the final sampling frequency depends on the computed initial sampling frequency n and the user-determined min sampling frequency, whichever is smaller.

When the final sampling frequency is 2, it means we just take the 3 vertices of the triangle as sample points, because:

$$n_samples = n * (n+1)/2 = 3.$$

- Compute Settings-># of bins

Specify the number of bins of the output histogram.

- Histogram Control->X/Y Min/Max

Specify the display range of the x-axis and the y-axis of the histogram.

- Colormap Control->Min/Max

Specify the color display range (min, max) of the surface distances. Any values that are lower/higher than min/max are mapped to the min/max corresponding color. Any values in-between the min and max value are mapped from color blue to red.

- ColormapControl->Interval

Specify the colormap interval within the display range. Lower interval makes the visualization more soft, while higher interval generates an error distribution colormap with stronger contrast.

5 Output

SampleErr Mean \bar{s}	$\bar{s} = \frac{1}{N} \sum_i s_i$	mean error of all the sample points
SampleErr STD σ	$\sigma = \sqrt{\frac{1}{N-1} \sum_i (s_i - \bar{s})^2}$	standard deviation of all the sample errors
SampleErr Min	$\min(s_i)$	minimum error of samples
SampleErr Max	$\max(s_i)$	maximum error of samples
FaceErr Mean \bar{f} or $d_m(S, S')$	$\bar{f} = (\sum_i f_i * a_i) / \sum_i a_i$ $d_m(S, S') = \frac{1}{ S } \int \int_{p \in S} d(p, S') dS$	mean error of all the subdivided triangle faces ^{1,2}
FaceErr RMS: f_{rms} or $d_{rms}(S, S')$	$f_{rms} = \sqrt{\sum_i f_i^2 * a_i / \sum_i a_i}$ $d_{rms}(S, S') = \sqrt{\frac{1}{ S } \int \int_{p \in S} d(p, S')^2 dS}$	root mean square of all the subdivided triangle faces ^{1,2}
Hausdorff $d(S, S')$	$d(S, S') = \max_{p \in S} d(p, S')$ $d(p, S') = \min_{p' \in S'} \ p - p'\ _2$	Hausdorff distance between two surfaces
MSD d_{msd}	$d_{msd} = \frac{1}{N} \sum_i s_i^2$	mean square distance of all the sample errors
MAD d_{mad}	$d_{mad} = \frac{1}{N} \sum_i s_i $	mean absolute distance of all the sample errors
Median		median error of samples (50 percentile)
68 Percentile		68 percentile of all the sample errors
95 Percentile		95 percentile of all the sample errors
Dice's coefficient	$D = 2 X \cap Y /(X + Y)$	twice the intersection over the sum of two volumes
Int Union Ratio	$R = X \cap Y / X \cup Y $	Intersection over the union

1. For each triangle formed by neighboring samples, the error at the vertices is averaged to obtain a single error for the sample triangle. The overall mean error is obtained by calculating the mean of the errors of the

sample triangles.

2. In a triangle i with area a_i and with values at the vertex (sample points) e_{i1} , e_{i2} and e_{i3} and using linear interpolation to obtain the values within the triangle, the mean value (i.e. integral of the value divided by the surface) is $f_i = \frac{e_{i1}+e_{i2}+e_{i3}}{3}$; the absolute mean value is $|f_i| = \frac{|e_{i1}|+|e_{i2}|+|e_{i3}|}{3}$; and the square mean value (i.e. integral of the squared value divided by the surface) is $f_i^{sqr} = \frac{e_{i1}^2+e_{i2}^2+e_{i3}^2+e_{i1}*e_{i2}+e_{i2}*e_{i3}+e_{i1}*e_{i3}}{6}$ as referred to Aspert's paper.

3. Note that, because of the asymmetric properties of the point-to-surface closest distance, the above statistics will be different for A->B, B->A, or A<->B.

For example, if the user choose A->B in the Compute Settings (step 2) panel, then MAD is referred to the minimum absolute distance from one point in object A to the surface of object B; or if the user choose A<->B in the same panel, the MAD is referred to the minimum absolute distance in both directions (from A to B and from B to A). All the other statistics are obtained analogously.

6 Screenshot

