Spectrum AUC Difference (SAUCD): Human-aligned 3D Shape Evaluation Supplementary Materials

The Supplementary Materials include the following contents:

1. The implementation details of our Spectrum AUC Difference metric and evaluation method.
2. Proof of our revision of the Cotan formula to be positive semidefinite.
3. A counterexample of the original Cotan formula not guaranteed to be semidefinite.
4. Objects and distortions in our *Shape Grading* dataset.
5. Swiss system tournament in the human scoring process.
6. Examples and evaluation results of different metrics in our dataset.
7. Implementation details on adapting SAUCD to training loss for 3D hand mesh reconstruction.
8. Failure cases.
9. Discussion of future works.

1. Implementation Details

1.1. Discretization of Spectrum AUC Difference

Our Spectrum AUC Difference (SAUCD) is defined in main paper Equation (7) as

$$d = D(\hat{M}, M_{gt}) = \int_{\lambda} |\hat{F}(\lambda) - F_{gt}(\lambda)| d\lambda, \qquad (1)$$

where $\hat{F}(\lambda)$ and $F_{gt}(\lambda)$ are the test and groundtruth mesh spectrum, respectively. To discretize Eq. (1) in the experiments, we let $\{\hat{\lambda}_i\}$ to be the discretized frequencies of $\hat{F}(\lambda)$ and $\{\lambda_{gt,i}\}$ to be the discretized frequencies of $F_{gt}(\lambda)$. We sort the two sets $\{\hat{\lambda}_i\}$ and $\{\lambda_{gt,i}\}$ into one array from low to high, resulting in a sorted array $\{\lambda_i\}$ with $N_{gt} + \hat{N}$ frequencies, where N_{gt} is the vertex number of the ground truth mesh and $\hat{N}$ is the vertex number of the test mesh. The $N_{gt} + \hat{N}$ frequencies discretize Eq. (1) into the sum of the area of $N_{gt} + \hat{N} - 1$ segments as:

$$d = \sum_{i=1}^{N_{gt}+\hat{N}-1} s_i, \qquad (2)$$

where the area of each segment

$$s_i = \begin{cases} \frac{1}{2}|H_i + H_{i-1}|(\lambda_i - \lambda_{i-1}), & H_i H_{i-1} \geq 0 \\ \frac{H_i^2 + H_{i-1}^2}{2|H_i + H_{i-1}|}(\lambda_i - \lambda_{i-1}), & H_i H_{i-1} < 0, \end{cases} \qquad (3)$$

is either a trapezoid when $H_i H_{i-1} \geq 0$ or two triangles when $H_i H_{i-1} < 0$. Here,

$$H_i = \hat{F}(\lambda_i) - F_{gt}(\lambda_i) \qquad (4)$$

is the amplitude difference between $\hat{F}(\lambda)$ and $F_{gt}(\lambda)$ at λ_i. If λ_i is originally from the test mesh spectrum, then

$$\hat{F}(\lambda_i) = \hat{F}(\hat{\lambda}_i), \qquad (5)$$

and $F_{gt}(\lambda_i)$ is calculated using interpolation as

$$F_{gt}(\lambda_i) = \frac{(\lambda_{gt,i+} - \lambda_i)F_{gt}(\lambda_{gt,i+}) + (\lambda_i - \lambda_{gt,i-})F_{gt}(\lambda_{gt,i-})}{\lambda_{gt,i+} - \lambda_{gt,i-}}, \qquad (6)$$

where $\lambda_{gt,i-}$ and $\lambda_{gt,i+}$ are the left and right nearest frequencies of λ_i in the groundtruch frequency set $\{\lambda_{gt,i}\}$. Similarly, if λ_i is originally from the ground truth mesh spectrum, then

$$F_{gt}(\lambda_i) = F_{gt}(\lambda_{gt,i}), \qquad (7)$$

and $\hat{F}(\lambda_i)$ is calculated using interpolation as

$$\hat{F}(\lambda_i) = \frac{(\hat{\lambda}_{i+} - \lambda_i)\hat{F}(\hat{\lambda}_{i+}) + (\lambda_i - \hat{\lambda}_{i-})\hat{F}(\hat{\lambda}_{i-})}{\hat{\lambda}_{i+} - \hat{\lambda}_{i-}}, \qquad (8)$$

where $\hat{\lambda}_{i-}$ and $\hat{\lambda}_{i+}$ are the left and right nearest frequencies of λ_i in the test frequency set $\{\hat{\lambda}_i\}$.

In summary, to calculate the area of the region between the two curves (*i.e.* AUC difference), we first sort the frequencies from the test and ground truth spectrum in one array, and interpolate the test and ground truth spectrum using the frequencies from the other spectrum. Then, we calculate each AUC difference in the range between two adjacent frequencies and add them together. When $H_i H_{i-1} \geq 0$, the region between the two curves is a trapezoid; when $H_i H_{i-1} < 0$ the region is two triangles and we calculate the sum area of the two triangles. Finally, the sum of the areas between adjacent frequencies is our Spectrum AUC Difference metric.

1.2. Discretization of Human-adjusted SAUCD

Our Human-adjusted SAUCD is defined in main paper Equation (8) as

$$d = D(\hat{M}, M_{gt}) = \int_{\lambda} w(\lambda)|\hat{F}(\lambda) - F_{gt}(\lambda)| d\lambda. \qquad (9)$$

Similar to SAUCD discretization, Human-adjusted SAUCD can be discretized as

$$d = \sum_{i=1}^{N_{gt}+\hat{N}-1} w_i s_i, \tag{10}$$

where s_i is defined the same as in Eq. (2), and w_i is the human-adjusted weight at λ_i in Eq. (3). Since the weight vector $\mathbf{w}$ we use is only 20-dimensional to avoid overfitting, we get each w_i by interpolating $\mathbf{w}$ at each λ_i. Specifically, the 20 elements of $\mathbf{w}$ represent the weights at frequencies uniformly distributed in the range from 0 to 0.05. We denote those 20 frequencies as $\{\lambda_{\mathbf{w},k}\}$ on which the weights $\mathbf{w}$ are explicitly defined, which means $0 \leq k < 20$, $\lambda_{\mathbf{w},0} = 0$, and $\lambda_{\mathbf{w},19} = 0.05$. The last frequency location 0.05 is picked empirically. Note that we use a revised version of Discrete Laplace-Beltrami Operator (DLBO) as in main paper Equation (4) to make sure $\lambda_i \geq 0$, then to calculate weight w_i whose corresponding $\lambda_i \notin \{\lambda_{\mathbf{w},k}\}$, we only consider when $\lambda_i > 0$. We use interpolation to calculate λ_i as

$$w_i = \begin{cases} \frac{(\lambda_{\mathbf{w},i+} - \lambda_i)\mathbf{w}(\lambda_{\mathbf{w},i+}) + (\lambda_i - \lambda_{\mathbf{w},i-})\mathbf{w}(\lambda_{\mathbf{w},i-})}{\lambda_{\mathbf{w},i+} - \lambda_{\mathbf{w},i-}}, & 0 < \lambda_i < \lambda_{\mathbf{w},19} \\ \lambda_{\mathbf{w},19}, & \lambda_i > \lambda_{\mathbf{w},19}, \end{cases} \tag{11}$$

where $\lambda_{\mathbf{w},i-}$ and $\lambda_{\mathbf{w},i+}$ are the left and right nearest element to λ_i in $\{\lambda_{\mathbf{w},k}\}$.

Having w_i, we can calculate Human-adjusted SAUCD following Eq. (10).

1.3. Evaluation methods

We use 3 different evaluation methods to evaluate the correlation between our metrics and human scoring (ground truth) on our provided *Shape Grading* dataset.

Pearson's linear correlation coefficient (PLCC). Pearson's correlation [21] evaluates the linear alignment between our metrics and human evaluation. It is defined as

$$p = \frac{\sum_{i=1}^{N}(h_i - \bar{h}_i)(m_i - \bar{m}_i)}{\sqrt{\sum_{i=1}^{N}(m_i - \bar{m}_i)^2}\sqrt{\sum_{i=1}^{N}(h_i - \bar{h}_i)^2}}, \tag{12}$$

where m_i is the score of mesh i given by the tested metric and h_i is the groundtruth score (human scoring) of mesh i. $\bar{h}_i$ and $\bar{m}_i$ are the average score of h_i and m_i, respectively.

Spearman's rank order correlation coefficient (SROCC). SROCC [23] is one of the most commonly used metrics to measure rank correlations. It is defined as

$$r_s = 1 - \frac{6\sum(R(m_i) - R(h_i))^2}{n(n^2-1)}, \tag{13}$$

where m_i and h_i is are defined the same as in Eq. (12). $R(m_i)$ and $R(h_i)$ are the rankings of m_i and h_i, and n is the amount of data. In our paper, n is the number of meshes scored by one subject.

Kendall's rank order correlation coefficient (KROCC). KROCC [10] is also a rank order correlation. It is defined as

$$\tau = 1 - \frac{2}{n(n^2-1)} \sum_{i<j} sgn(m_i - m_j) sgn(h_i - h_j), \tag{14}$$

where m_i, h_i, and n is the same with Eq. (13), and $sgn(\dot{})$ is the sign function, which means $sgn(x) = 1$ when $x > 0$, $sgn(x) = -1$ when $x < 0$, and $sgn(x) = 0$ when $x = 0$. The difference between SROCC and KROCC is that SROCC considers the actual amount of rank order difference of input data, while KROCC only counts the number of inverse pairs.

The possible ranges of all 3 metrics are $[-1, 1]$. Higher numbers mean stronger correlations.

1.4. Human-adjusted SAUCD training

During training, Pearson's correlation loss $\mathcal{L}_{plcc}$ and Spearsman's rank order loss $\mathcal{L}_{srocc}$ in main paper Equation (9) are defined the same as Eq. (12) and Eq. (13), respectively. Note that, since the rank part of SROCC is not naturally differentiable, we used a differentiable ranking approach provided in [1] to make Eq. (13) differentiable. We set $\lambda_p = 0.1$, $\lambda_{sr} = 10$, and $\lambda_{regu} = 1$ for main paper Equation (9). The training process took about 1 minute on a 14-core Intel Xeon CPU. The training code is implemented using PyTorch [20].

2. Proof of Positive-semidefiniteness of Revised Cotan Formula

In this section, we prove that our revised version of the Cotan formula in main paper Equation (4) is positive semidefinite. Here, the DLBO defined in main paper Equation (4) is

$$L_{ij} = \begin{cases} \frac{1}{2}\sum_{j \in N(i)} A_i^{-\frac{1}{2}} A_j^{-\frac{1}{2}} |\cot\alpha_{ij} + \cot\beta_{ij}|, & i = j \\ -\frac{1}{2} A_i^{-\frac{1}{2}} A_j^{-\frac{1}{2}} |\cot\alpha_{ij} + \cot\beta_{ij}|, & i \neq j \wedge j \in N(i) \\ 0, & i \neq j \wedge j \notin N(i). \end{cases} \tag{15}$$

According to the Gershgorin circle theorem [8], for every eigenvalue λ_k of L,

$$\lambda_k \in \bigcup_i S_i, \tag{16}$$

where S_i is the ith Gershgorin disc. The Gershgorin disc is defined as

$$S_i = \{z \in \mathbb{C} : |z - L_{ii}| \leq R_i = \sum_{i \neq j} |L_{ij}|\}, \tag{17}$$

where $\mathbb{C}$ means the complex space. Since L is a real symmetric matrix, according to Eq. (15), the Gershgorin disc degenerates into a line segment in the real space as

$$S_i = \{s \in \mathbb{R} : |s - L_{ii}| \leq R_i = \sum_{i \neq j} |L_{ij}|\}. \tag{18}$$

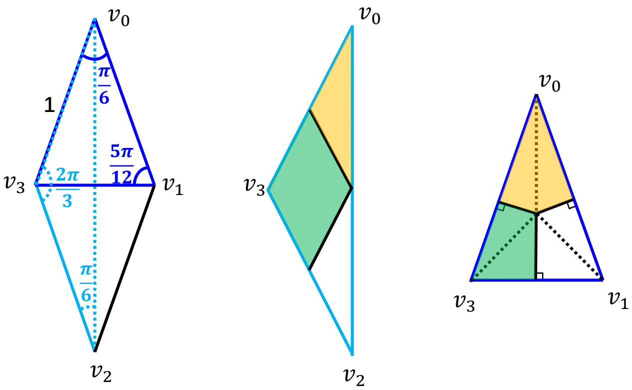


Figure 1. A simple mesh example to show that the original Cotan formula does not guarantee to be positive semidefinite.

From Eq. (15), we can also have

$$\sum_{i\neq j}|L_{ij}| = \sum_{j\in N(i)} \frac{|\cot\alpha_{ij}+\cot\beta_{ij}|}{2\sqrt{A_iA_j}} = L_{ii}. \tag{19}$$

Note that $L_{ii}\geq 0$, so having Eq. (19), from Eq. (18) we get

$$S_i = \{s\in R : |s-L_{ii}|\leq R_i = L_{ii}\} \Leftrightarrow 0\leq S_i\leq 2L_{ii}. \tag{20}$$

Thus, according to Eq. (16), we have

$$0\leq\lambda_k\leq 2\max_i L_{ii}, \forall 0\leq k\leq N, \tag{21}$$

where N is the number of vertices. Then, L is positive semidefinite since L is a real symmetric matrix and all its eigenvalues are greater than or equal to zero.

Q.E.D.

3. A Counterexample of the Original Cotan Formula not Being Positive Semidefinite

In this section, we provide a simple mesh example to show that the original Cotan formula in main paper Equation (2) does not guarantee to be positive semidefinite. As shown in Fig. 1a, we reconstruct a 4-vertex mesh that is not Delauney triangulated and the mixed Voronoi areas of the vertices are not all equal. We make the two faces on the bottom ($v_1v_2v_0$ and $v_3v_0v_2$) be two congruent obtuse isosceles triangles (shown in Fig. 1b). The apex angles of the two isosceles triangles are $\frac{2\pi}{3}$, and the base angles are $\frac{\pi}{6}$. If we make the bottom two obtuse triangles form different angles to each other, the top two triangle faces ($v_0v_1v_3$ and $v_2v_3v_1$) are always congruent isosceles triangles (as in Fig. 1c), and their apex angles vary continuously in the range of $(0, \frac{\pi}{3})$. Here, we make the bottom two obtuse triangles form a certain angle to each other so that the apex angles of the top two triangles are equal to $\frac{\pi}{6}$, which means their base angles are $\frac{5\pi}{12}$. For simplicity, we set the equal sides of the isosceles triangles to be 1 (shown in Fig. 1a).

Now, we calculate the DLBO metric of this reconstructed mesh using the Cotan formula in main paper Equation (2). First, we calculate the mixed Voronoi area for each vertex. Because of the shape symmetry, we only need to calculate the mixed Voronoi areas for vertex v_0 and v_3. The mixed Voronoi areas for vertex v_2 and v_1 are equal to v_0 and v_3, respectively. For vertex v_0, its mixed Voronoi area A_0 can be calculated as the sum of 2 times of yellow area in Fig. 1b and 1 time of yellow area in Fig. 1c, which means

$$\begin{aligned} A_0 &= 2\times(\frac{1}{4}\times\frac{1}{2}\cos\frac{\pi}{3}) + 1\times(0.5\tan\frac{\pi}{12}\times 0.5) \\ &= \frac{4-\sqrt{3}}{8}, \end{aligned} \tag{22}$$

where $\frac{1}{2}\cos\frac{\pi}{3}$ is the area of the outer triangle in Fig. 1b and $0.5\tan\frac{\pi}{12}\times 0.5$ is the area of the yellow part in Fig. 1c. For vertex v_3, its mixed Voronoi area A_3 can be calculated as the sum of 1 time of green area in Fig. 1b and 2 times of green area in Fig. 1c, which means

$$\begin{aligned} A_3 &= 1\times(\frac{1}{2}\times\frac{1}{2}\cos\frac{\pi}{3}) \\ &+ 2\times(\frac{1}{2}\times(\sin\frac{\pi}{12}\cos\frac{\pi}{12} - 0.5\tan\frac{\pi}{12}\times 0.5)) \\ &= \frac{3\sqrt{3}-2}{8}, \end{aligned} \tag{23}$$

where $\sin\frac{\pi}{12}\cos\frac{\pi}{12}$ is the area of the outer triangle in Fig. 1c.

Second, we calculate the DLBO matrix according to main paper Equation (2). The DLBO matrix of the constructed mesh can be represented as

$$L = \begin{pmatrix} \frac{w_1}{2A_0} & \frac{w_0}{2A_0} & \frac{w_3}{2A_0} & \frac{w_0}{2A_0} \\ \frac{w_0}{2A_3} & \frac{w_2}{2A_3} & \frac{w_0}{2A_3} & \frac{w_4}{2A_3} \\ \frac{w_3}{2A_0} & \frac{w_0}{2A_0} & \frac{w_1}{2A_0} & \frac{w_0}{2A_0} \\ \frac{w_0}{2A_3} & \frac{w_4}{2A_3} & \frac{w_0}{2A_3} & \frac{w_2}{2A_3} \end{pmatrix}, \tag{24}$$

where

$$\begin{aligned} w_0 &= -(\cot\frac{5\pi}{12}+\cot\frac{\pi}{6}) = -2, \\ w_1 &= 2(\cot\frac{5\pi}{12}+\cot\frac{\pi}{6}+\cot\frac{2\pi}{3}) = 4-\frac{2\sqrt{3}}{3}, \\ w_2 &= 2(\cot\frac{5\pi}{12}+\cot\frac{\pi}{6}+\cot\frac{\pi}{6}) = 4+2\sqrt{3}, \\ w_3 &= -2\cot\frac{2\pi}{3} = \frac{2\sqrt{3}}{3}, \\ w_4 &= -2\cot\frac{\pi}{6} = -2\sqrt{3}. \end{aligned} \tag{25}$$

Then, we can calculate the symmetric part of L as

$$L_{sym} = \frac{L+L^\top}{2}. \tag{26}$$

Distortion types	Description	Generating details
Impulse	Adding impulsive noise on mesh surface	We add Gaussian noise on r percent of the ground truth mesh vertices. The mean of the Gaussian noise is set to 0 and standard derivation is set to σ percent of the mesh scale. For 4 levels of this distortion, (r, σ) are set to (1, 0.5), (5, 2), (8, 3), and (1, 5), respectively.
Poisson reconstruction noise	Synthesizing the noise occurs in Poisson reconstruction [9]	We first use Poisson disk sampling [2] to sample sN points from the groundtruth mesh surface, where N is the number of vertices in groundtruth mesh. Then, we use Poisson reconstruction provided in MeshLab [5] to reconstruct the mesh surface from the sampled points. The reconstruction depth is set to 6. For 4 levels of this distortion, s is set to 0.9, 0.5, 0.2, and 0.05, respectively.
Smoothing	Smoothing mesh surface	We apply i times of $\lambda - \mu$ Taubin smoothing [25] to smooth the groundtruth mesh surface, where $\lambda = 0.5$ and $\mu = -0.53$. For 4 levels of this distortion, i is set to 5, 20, 50, and 200, respectively.
Unproportional scaling	Stretching (or shrinking) the mesh along x, y, and z axis with different rates	We stretch the mesh to s_x percent to its original length along x axis, and shrink the mesh to s_z percent to its original length along z axis. For 4 levels of this distortion, (s_x, s_z) are set to (98, 102), (95, 105), (90, 110), (80, 120), respectively.
Low-resolution mesh	Simplifying mesh surface to lower resolution	We simply the ground truth mesh surface using edge collapse algorithm [7]. For 4 levels of this distortion, the target face number is set to 5000, 2000, 1000, and 500, respectively.
White noise	Adding Gaussian white noise on mesh surface	We add Gaussian noise on *all* the groundtruth mesh vertices. The mean of the Gaussian noise is set to 0 and standard derivation is set to σ percent of the mesh scale. For 4 levels of this distortion, σ is set to 0.1, 0.2, 0.3, and 0.5, respectively.
Outlying noise	Adding outlying small floating spheres around the mesh	We add floating spheres around the ground truth mesh to synthesize outlying noise that occurs in 3D reconstruction. The number of the spheres is set to n and the radius rA, where A is the maximum length of the mesh along x, y, and z dimensions. The locations of the spheres are sampled randomly from a cube that surrounds the ground truth mesh. The edge size of the cube is set to $(1 + 6r)A$. For 4 levels of this distortion, (n, r) are set to (20, 0.002), (30, 0.004), (40, 0.006), (80, 0.008), respectively.

Table 1. Distortions in our provided *Shape Grading* dataset.

We use Wolfram Mathematica [26] to calculate the eigenvalues of L_{sym}. The 4 eigenvalues are

$$\begin{aligned}
\lambda_0 &= \frac{2 - \frac{2\sqrt{3}}{3}}{A_0}, \\
\lambda_1 &= \frac{2 + 2\sqrt{3}}{A_3}, \\
\lambda_2 &= \frac{A_0 + A_3 - \sqrt{2(A_0^2 + A_3^2)}}{A_0 A_3}, \\
\lambda_3 &= \frac{A_0 + A_3 + \sqrt{2(A_0^2 + A_3^2)}}{A_0 A_3}.
\end{aligned} \tag{27}$$

It is obvious that when A_0 and A_3 are both greater than 0, λ_0, λ_1, and λ_3 will be greater than 0. However, for λ_2, we have

$$\begin{aligned}
\lambda_2 &= \frac{A_0 + A_3 - \sqrt{2(A_0^2 + A_3^2)}}{A_0 A_3} \\
&= \frac{\sqrt{A_0^2 + A_3^2 + 2A_0A_3} - \sqrt{2(A_0^2 + A_3^2)}}{A_0 A_3} \\
&\leq \frac{\sqrt{A_0^2 + A_3^2 + (A_0^2 + A_3^2)} - \sqrt{2(A_0^2 + A_3^2)}}{A_0 A_3} \\
&= 0.
\end{aligned} \tag{28}$$

The equation holds if and only if $A_0 = A_3$. We know from Eq. (22) and Eq. (23) that $A_0 \neq A_3$. Thus, we have

$$\lambda_2 < 0, \tag{29}$$

which means in the given mesh example, the original Cotan formula is not positive semidefinite.

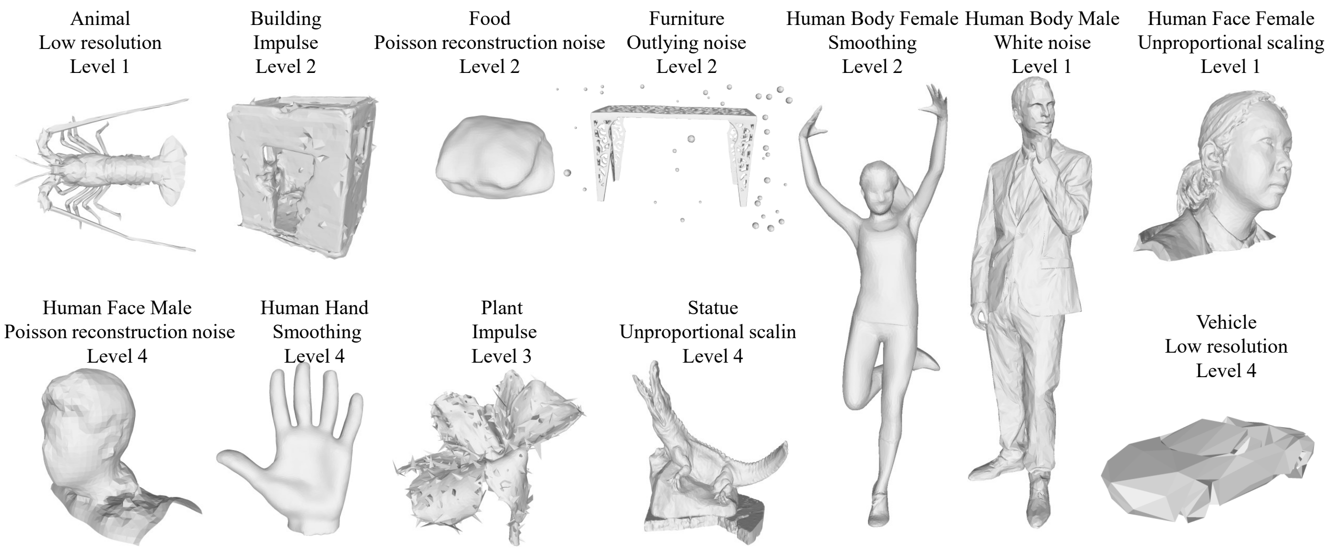


Figure 2. Examples of distorted meshes of different distortion levels in our provided *Shape Grading* dataset.

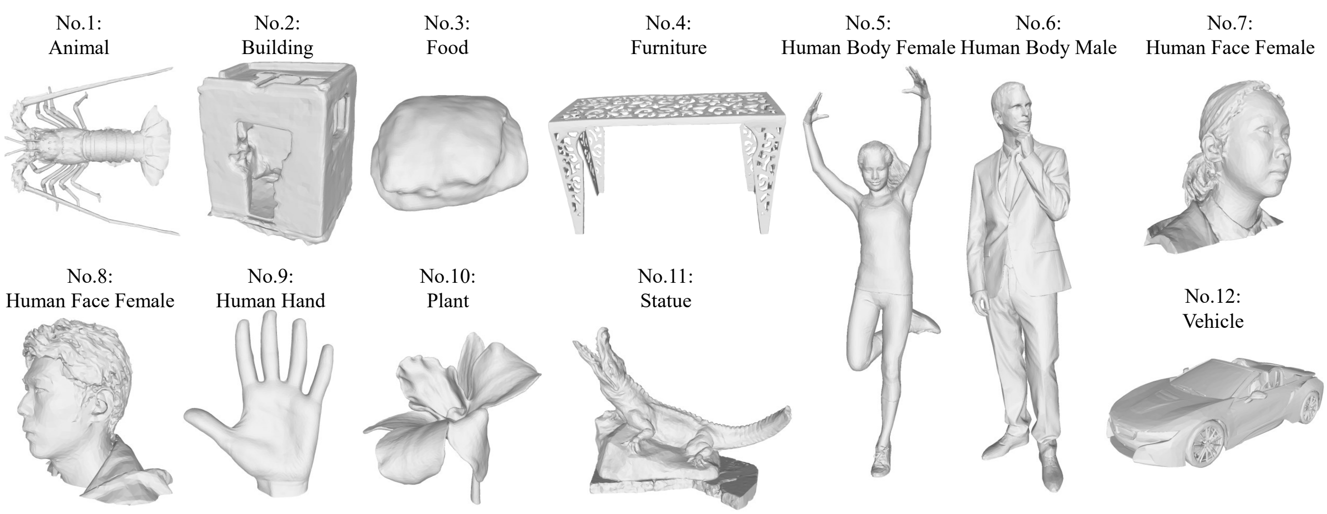


Figure 3. Objects in our provided *Shape Grading* dataset and what the object numbers correspond to in main paper Table 2.

4. Objects and Distortions in *Shape Grading*

Fig. 3 shows the objects in our proposed dataset *Shape Grading* and what the object numbers correspond to in main paper Table 2. We also show the distortion types that we used in our dataset and how we generate them in Tab. 1. Fig. 2 shows examples of distorted meshes of different distortion levels in our dataset.

5. Swiss System Tournament for Human Scoring

We do a Swiss system tournament for human scoring in main paper Section 4.1. The tournament has 6 rounds. To begin with, all 28 meshes are set to 0 points. In the first round, the 28 meshes are randomly sorted and we form the adjacent meshes into pairs (the 1st and 2nd meshes form a pair, the 3rd and 4th meshes form another pair, *etc*.). Together, we have 14 pairs. For each pair, we ask the subject which one is closer to the ground truth. The mesh that the subject picked will be added 1 point. From the 2nd to the 6th round, for each round, we first sort the meshes by their current score from low to high, and we also make pairs with adjacent meshes in the sorted mesh array, like what we did in the first round. The mesh closer to ground truth will be added 1 point. The scores of the meshes after 6 rounds are their scores graded by this subject. Fig. 4 shows the panel of our online human scoring page.

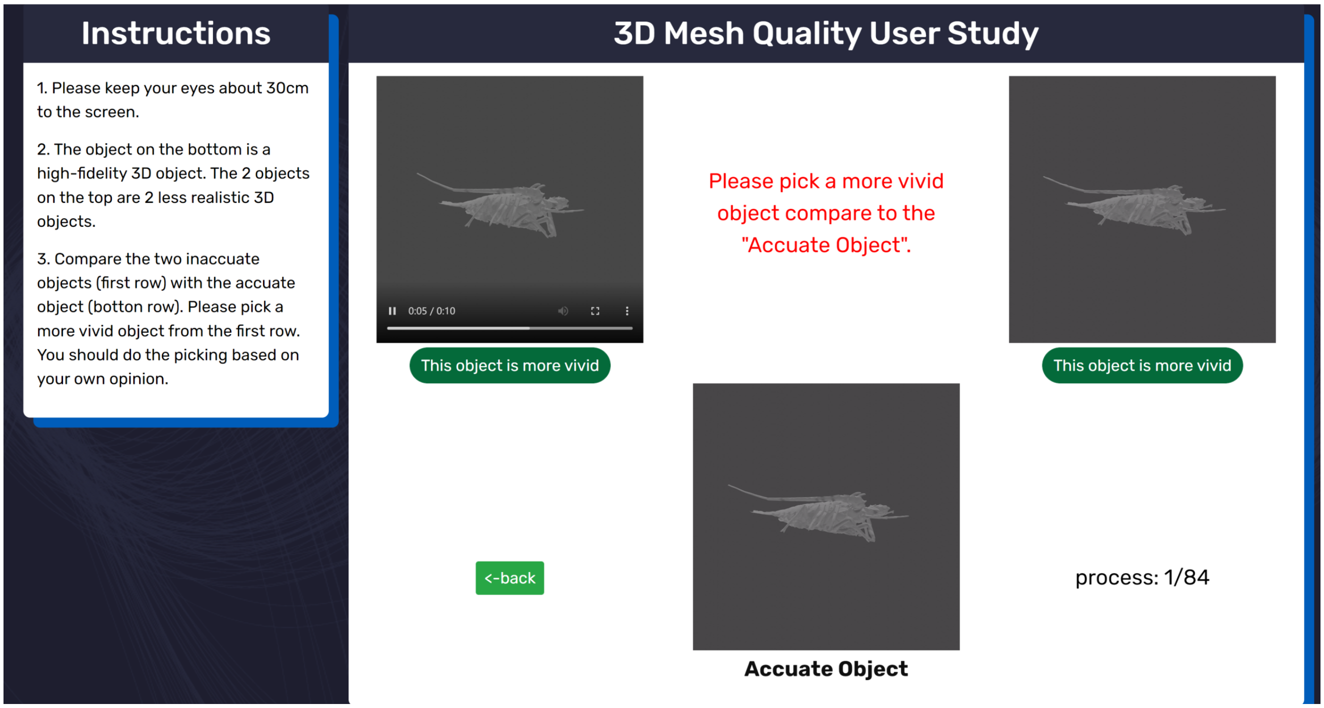


Figure 4. The panel of our online user study system. The instructions on the left contain simple instructions for the subjects. On the right side of the page, the top two videos are rendered from distorted meshes. The lower video is rendered from ground truth mesh.

6. More Examples and Evaluation Results

We show more examples in our dataset and evaluation results using different metrics in Fig. 5. Compared to previous methods, our provided metrics generally align better with the human evaluation of mesh shape similarity.

7. Implementation Details on Adapting SAUCD to Training Loss

We adapt SAUCD to a topology Laplacian version. Specifically, we replace the Laplacian matrix defined in the main paper Eq.(4) to $L = D - A$ defined in [4], where D is the degree matrix of the mesh graph, and A is the adjacency matrix of the mesh graph. By making the change, we can avoid calculating a different SVD decomposition in every training iteration when mesh vertex locations change. Our network is designed as Fig. 6. The input image first goes through a feature extraction CNNs network to get image features, and uses that feature to generate MANO [22] mesh. Then, we use features from CNNs network and 3 resolution levels of Graph Convolution Networks (GCN) to reconstruct the mesh details. In the main paper Fig. 8, we compare the results using only MVPE loss (w/o SAUCD loss column) and using both MVPE and SAUCD loss (w/ SAUCD loss column). In this experiment, we use EfficientNet [24] and GCN similar to [11].

8. Failure Cases

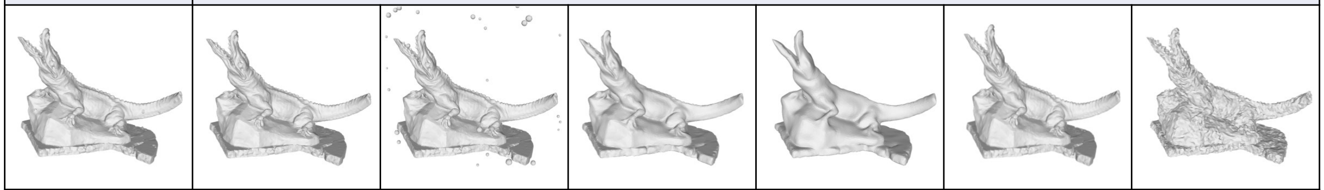
We also show a case that our metric does not provide accurate evaluations aligned with the human evaluation in Fig. 7.

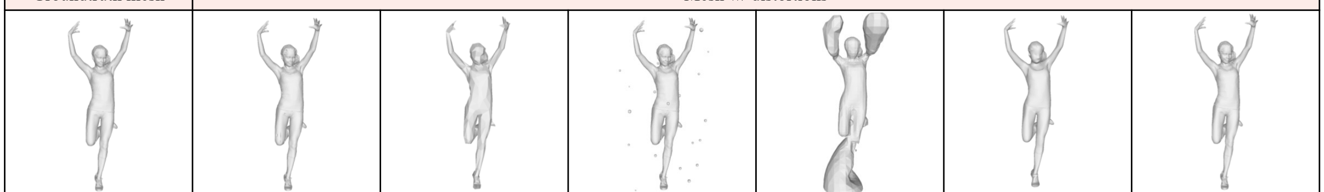
9. Discussions of Future Works

In future work, we plan to dig deeper into understanding human sensitivity to frequency changes. To enhance the robustness and applicability of our approach, we plan to expand our dataset to include a wider range of distortions and objects. While our current methods are effective on general 3D meshes, we recognize the importance of developing specialized versions for particular areas of 3D reconstruction, such as human body [12, 13, 18], human face [6], human hand [19], or volumetric representations [14–17]. Furthermore, the frequency method holds promise for extension into 2D domains, including image classification/segmentation/generation [29], as well as video analysis/generation [3, 27, 28]. These future works will not only refine our understanding of human perception alignment but also broaden the potential applications of our research in various fields.

References

[1] Mathieu Blondel, Olivier Teboul, Quentin Berthet, and Josip Djolonga. Fast differentiable sorting and ranking. In *ICML*, pages 950–959, 2020. 2

Groundtruth mesh	Mesh w/ distortions					
						
User study↑	4.70	3.93	3.90	2.38	4.81	2.37
Ours↓	0.30	0.42	0.41	1.35	0.27	0.89
Ours extended↓	0.23	0.31	0.35	1.20	0.21	0.73
Chamfer Distance↓	0.07	29.52	3.10	5.02	12.62	4.83
IoU↑	1.00	0.24	0.97	0.95	0.68	0.94
F-score↑	1.00	0.93	1.00	1.00	0.95	1.00
SSFID↓	0.00	1.38	0.01	0.02	0.05	0.08
UHD↓	12.60	0.00	30.13	37.96	36.93	14.12

Groundtruth mesh	Mesh w/ distortions					
						
User study↑	4.40	2.63	4.03	0.51	3.68	4.66
Ours↓	0.46	1.08	0.51	1.22	0.89	0.48
Ours extended↓	0.36	0.92	0.43	1.14	0.90	0.38
Chamfer Distance↓	0.006	1.32	1.86	2.45	0.44	0.63
IoU↑	1.00	0.87	0.24	0.09	0.89	0.92
F-score↑	1.00	0.95	0.94	0.85	1.00	1.00
SSFID↓	0.0002	0.04	0.44	8.57	0.03	0.02
UHD↓	1.03	6.72	0.51	1.22	0.89	0.48

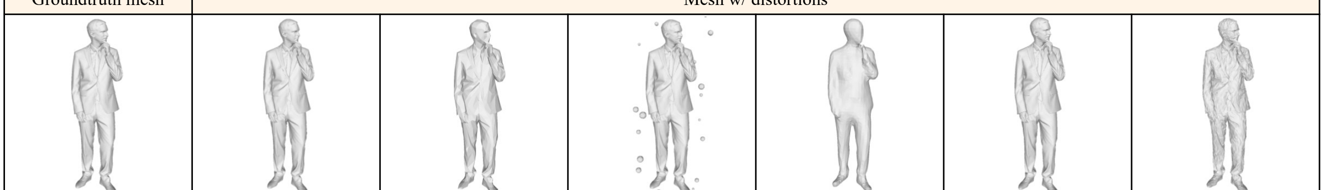
Groundtruth mesh	Mesh w/ distortions					
						
User study↑	4.64	2.74	4.10	1.87	4.67	3.01
Ours↓	0.52	7.02	0.53	1.25	0.55	0.94
Ours extended↓	0.70	1.43	0.76	1.99	0.80	1.19
Chamfer Distance↓	0.01	1.26	2.12	1.13	0.69	0.30
IoU↑	1.00	0.96	0.29	0.81	0.93	0.97
F-score↑	1.00	0.97	0.93	1.00	1.00	1.00
SSFID↓	0.0001	0.01	0.59	0.11	0.03	0.004
UHD↓	1.45	1.02	0.53	1.25	0.55	0.94

Figure 5. Examples in our dataset and their evaluation results using different metrics. ↓ means lower is better. ↑ means higher is better. For each object, the mesh on the top-left is the ground truth mesh, and the rest meshes are distorted meshes. The table below the meshes contains the scores they get from different metrics or from our user study. As shown in the figure, our metric aligns better with user study scores and human perception.

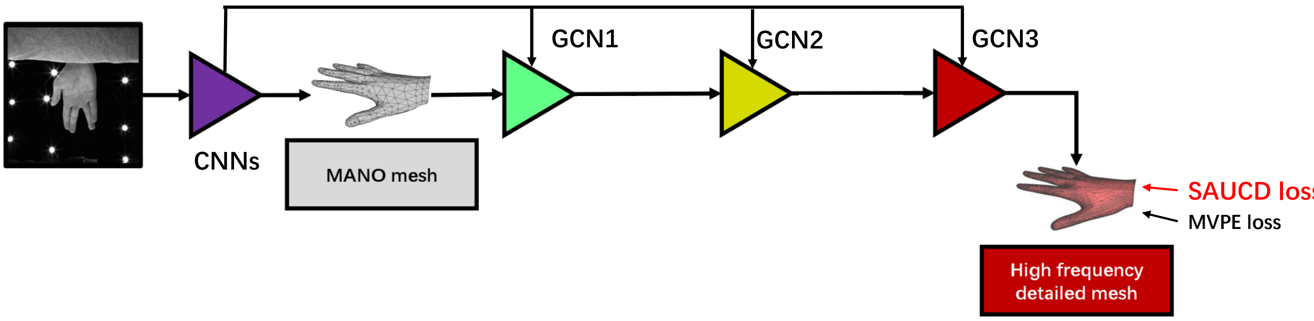


Figure 6. Network architecture used when adapting SAUCD to training loss.

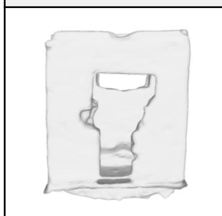
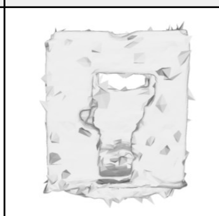
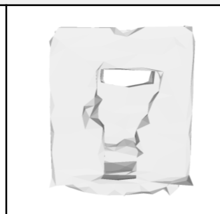
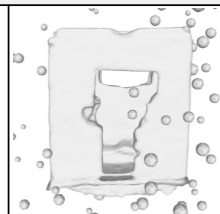
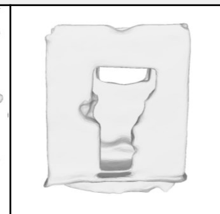
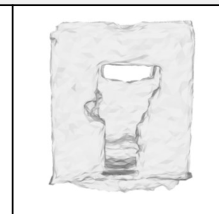
Groundtruth mesh	Mesh w/ distortions				
					
User study↑	1.23	3.74	2.36	4.57	2.63
Ours↓	0.72	1.13	0.80	0.92	0.65
Ours extended↓	1.06	1.96	1.12	1.34	1.02

Figure 7. Failure cases. We show a case in which our metric does not provide accurate evaluations aligned with the human evaluation.

[2] Robert Bridson. Fast poisson disk sampling in arbitrary dimensions. *SIGGRAPH sketches*, page 1, 2007. 4

[3] Hang Chen, Xinyu Yang, and Xiang Li. Learning a general clause-to-clause relationships for enhancing emotion-cause pair extraction. *arXiv preprint arXiv:2208.13549*, 2022. 6

[4] Fan RK Chung. *Spectral graph theory*. American Mathematical Soc., 1997. 6

[5] Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. MeshLab: an Open-Source Mesh Processing Tool. In *Eurographics Italian Chapter Conference*, 2008. 4

[6] Zhongpai Gao. Learning continuous mesh representation with spherical implicit surface. In *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–8. IEEE, 2023. 6

[7] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *SIGGRAPH*, pages 209–216, 1997. 4

[8] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge University Press, 2012. 2

[9] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Eurographics Symposium on Geometry Processing*, 2006. 4

[10] Maurice George Kendall et al. The advanced theory of statistics. *The advanced theory of statistics*, 1946. 2

[11] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, pages 4501–4510, 2019. 6

[12] Zhong Li, Yu Ji, Wei Yang, Jinwei Ye, and Jingyi Yu. Robust 3d human motion reconstruction via dynamic template construction. In *3DV*, pages 496–505, 2017. 6

[13] Zhong Li, Minye Wu, Wangyiteng Zhou, and Jingyi Yu. 4d human body correspondences from panoramic depth maps. In *CVPR*, pages 2877–2886, 2018. 6

[14] Zhong Li, Lele Chen, Celong Liu, Fuyao Zhang, Zekun Li, Yu Gao, Yuanzhou Ha, Chenliang Xu, Shuxue Quan, and Yi Xu. Animated 3d human avatars from a single image with gan-based texture inference. *Computers & Graphics*, pages 81–91, 2021. 6

[15] Zhong Li, Liangchen Song, Celong Liu, Junsong Yuan, and Yi Xu. Neulf: Efficient novel view synthesis with neural 4d light field. *arXiv preprint arXiv:2105.07112*, 2021.

[16] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. *arXiv preprint arXiv:2312.16812*, 2023.

[17] Zhong Li, Liangchen Song, Zhang Chen, Xiangyu Du, Lele Chen, Junsong Yuan, and Yi Xu. Relit-neulf: Efficient relighting and novel view synthesis via neural 4d light field. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7007–7016, 2023. 6

[18] Tianyu Luan, Yali Wang, Junhao Zhang, Zhe Wang, Zhipeng Zhou, and Yu Qiao. Pc-hmr: Pose calibration for 3d human

mesh recovery from 2d images/videos. In *AAAI*, pages 2269–2276, 2021. 6

[19] Tianyu Luan, Yuanhao Zhai, Jingjing Meng, Zhong Li, Zhang Chen, Yi Xu, and Junsong Yuan. High fidelity 3d hand shape reconstruction via scalable graph frequency decomposition. In *CVPR*, pages 16795–16804, 2023. 6

[20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 32, 2019. 2

[21] Karl Pearson. Notes on the history of correlation. *Biometrika*, pages 25–45, 1920. 2

[22] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *SIGGRAPH*, 2017. 6

[23] Charles Spearman. Correlation calculated from faulty data. *British journal of psychology*, page 271, 1910. 2

[24] Mingxing Tan and Quoc V Le. Efficientnet: Improving accuracy and efficiency through automl and model scaling. *arXiv preprint arXiv:1905.11946*, 2019. 6

[25] Gabriel Taubin. Curve and surface smoothing without shrinkage. In *ICCV*, pages 852–857, 1995. 4

[26] Stephen Wolfram. *Mathematica: a system for doing mathematics by computer*. Addison Wesley Longman Publishing Co., Inc., 1991. 4

[27] Yuanhao Zhai, Le Wang, Wei Tang, Qilin Zhang, Junsong Yuan, and Gang Hua. Two-stream consensus network for weakly-supervised temporal action localization. In *ECCV*, pages 37–54, 2020. 6

[28] Yuanhao Zhai, Mingzhen Huang, Tianyu Luan, Lu Dong, Ifeoma Nwogu, Siwei Lyu, David Doermann, and Junsong Yuan. Language-guided human motion synthesis with atomic actions. In *ACM MM*, pages 5262–5271, 2023. 6

[29] Yuanhao Zhai, Tianyu Luan, David Doermann, and Junsong Yuan. Towards generic image manipulation detection with weakly-supervised self-consistency learning. In *ICCV*, pages 22390–22400, 2023. 6