A new mesh visual quality metric using saliency weighting-based pooling strategy

Xiang Feng[a,b,c,*], Wanggen Wan[a,c], Richard Yi Da Xu[b], Stuart Perry[b], Song Zhu[d], Zexin Liu[a,c]

[a] *School of Communication and Information Engineering, Shanghai University, Shanghai, China*
[b] *Faculty of Engineering and Information Technology, University of Technology Sydney, Australia*
[c] *Institute of Smart City, Shanghai University, Shanghai, China*
[d] *Department of Engineering Enzo Ferrari, University of Modena and Reggio Emilia, Modena, Italy*

Abstract

Several metrics have been proposed to assess the visual quality of 3D triangular meshes during the last decade. In this paper, we propose a mesh visual quality metric by integrating mesh saliency into mesh visual quality assessment. We use the Tensor-based Perceptual Distance Measure metric to estimate the local distortions for the mesh, and pool local distortions into a quality score using a saliency weighting-based pooling strategy. Three well-known mesh saliency detection methods are used to demonstrate the superiority and effectiveness of our metric. Experimental results show that our metric with any of three saliency maps performs better than state-of-the-art metrics on the LIRIS/EPFL general-purpose database. We generate a synthetic saliency map by assembling salient regions from individual saliency maps. Experimental results reveal that the synthetic saliency map achieves better performance than individual saliency maps, and the performance gain is closely correlated with the similarity between the individual saliency maps.

Keywords: Mesh visual quality assessment, Mesh saliency, Tensor-based Perceptual Distance Measure, Saliency weighting-based pooling, Synthetic saliency map

1. Introduction

With the advance of 3D acquisition techniques, 3D triangular mesh has become a standard digital representation of 3D object surface and is widely used in various human centered applications. A 3D triangular mesh is always subject to geometric distortions during common processing operations, such as compression, watermarking and smoothing. Since the geometric distortions may degrade the visual quality of 3D triangular meshes, it is critical to assess the perceptual quality of 3D triangular meshes. It is inappropriate to ask human subjects to evaluate the visual distortion of 3D triangular meshes in most practical applications since it is both time-consuming and tedious. Thus, it is necessary to develop computational metrics to assess the perceptual quality of 3D triangular meshes accurately. Some well-performing metrics have been proposed for mesh visual quality (MVQ) assessment, such as Mesh Structural Distortion Measure (MSDM) [1], Multiscale Mesh Structural Distortion Measure (MSDM2) [2], Fast Mesh Perceptual Distance (FPDM) [3], Dihedral Angle Mesh Error (DAME) [4], Tensor-based Perceptual Distance Measure (TPDM) [5], Dong [6].

As another important research area of visual perception, mesh saliency detection [7] has also attracted much attention in the community. Many computational saliency methods [8–12] have been proposed to detect perceptually important regions where human visual attention is focused on the mesh. Since the receptor of both mesh visual quality and mesh saliency is the human visual system, we believe that it is possible to improve the performance of MVQ metrics by incorporating mesh saliency. Actually, in the community of image quality assessment, there are already some works [13–17] that investigated incorporating either visual attention or computational visual saliency into image quality metrics (IQMs). Zhang et al. [18] presented a statistical evaluation to investigate the added value of integrating computational saliency into IQMs. They concluded that the computational saliency models can yield a performance gain statistically when integrating computational saliency into IQMs though the specific amount of performance gain depends on the combination of saliency model and IQM [18]. Compared with the works in image quality assessment, there are relatively fewer works that investigated the relationship between mesh saliency and mesh visual quality, not to mention the incorporation of mesh saliency in MVQ metrics. In [13–18], either visual attention or computational visual saliency was incorporated in image quality metrics to improve the performance based on the assumption that distortions occurring in more salient areas of an image are more visible and thus more annoying, which was finally verified by the experimental results. Since the ultimate assessors of both mesh quality and image quality

* Corresponding author.
Email address: fengxiang0727@shu.edu.cn (Xiang Feng)

are human visual system, in this paper we similarly assume that, in mesh visual quality assessment, distortions appearing in more salient regions of a mesh are more annoying. Based on this assumption, we propose a MVQ metric by integrating mesh saliency into MVQ assessment.

As mentioned in [7], many methods have been proposed to detect mesh saliency. But the problem is which saliency detection methods we should choose to perform the analysis of integrating mesh saliency into MVQ assessment. Kim et al. [19] conducted an user study with five 3D models based on eye-tracking experiment and quantified the correlation between the mesh saliency computed by the method [8] and fixation locations acquired from an eye-tracking experiment. However, to the best of our knowledge, until now there is not yet a publicly accessible ground-truth eye-tracking database that records fixation points of visual attention on 3D triangular meshes. Chen et al. [20] introduced a benchmark with pseudo-ground truth saliency on the mesh based on Schelling points, and used a regression model to predict mesh saliency with the benchmark. Tasse et al. [21] proposed three metrics to quantitatively evaluate 3D computational saliency models based on the benchmark [20]. The evaluation involves three 3D computational saliency models which were previously proposed in [9, 22, 23]. But there is a lack of comprehensive quantitative analysis to reveal the accuracy and reliability of state-of-the-art mesh saliency detection methods. In [8–12], the effectiveness of the mesh saliency detection methods was justified mostly through either application-guided evaluation [8–10] or subjective visual analysis [11, 12]. Since the three mesh saliency detection methods proposed in [8–10] were demonstrated to be capable of enhancing the results of graphics applications, such as mesh simplification and viewpoint selection, we use them [8–10] to evaluate the benefits of incorporating mesh saliency into MVQ metric in this paper. We firstly generate a distortion map with the TPDM metric [5], which is one of the best-performing MVQ metrics until now, then generate a saliency map with each of three mesh saliency detection methods [8–10], and finally derive the overall quality score for the mesh via saliency weighting-based pooling of local distortions.

The remainder of this paper is organized as follows: We review related work on MVQ metrics, mesh saliency detection methods and the incorporation of visual saliency in IQMs in Section 2. We introduce our proposed MVQ metric in Section 3. We give a brief description of three mesh saliency detection methods used in this paper and present an analysis of the saliency maps generated by three methods in Section 4. We present the experimental results and analysis in Section 5 and conclude the paper in Section 6.

2. Related work

In the last decade, some MVQ metrics have been designed to predict human judgement on the quality of 3D triangular mesh. Detailed reviews of MVQ metrics can be found in [24, 25]. The classical geometric distances, such as Hausdorff Distance and Root Mean Squared Error, are demonstrated to have weak correlation with human visual perception [25]. There is still no clear consensus on the suitability of image-based metrics in MVQ assessment. The literature [26] argues that image-based metrics [27, 28] are not suitable for evaluating the quality of meshes while the literature [29] suggests that image-based metrics can be used for evaluating the quality of distorted meshes of the same object under a single type of distortion. Some model-based perceptual metrics have been proposed for MVQ assessment by exploiting geometric features. Karni and Gotsman [30] measured the distance between the distorted mesh and the reference mesh by comparing both vertex coordinates and geometric Laplacian values of two meshes. Sorkine et al. [31] improved the method [30] by assigning a greater weight to geometric Laplacian values. Corsini et al. [32] developed two perceptual metrics, $3DWPM_1$ and $3DWPM_2$, based on the roughness difference between two meshes. Bian et al. [33] proposed a physically-inspired metric based on strain energy that induces the deformation to the reference mesh. Lavoué et al. proposed the MSDM metric [1] by extending structural similarity index [34] in image quality assessment to MVQ assessment. Later, a multiscale version MSDM2 [2] was proposed to address the issue of changed connectivity of distorted meshes based on the work [1]. Wang et al. [3] introduced the FMPD metric to compute the perceptual distortion between two meshes based on global roughness derived from the Laplacian of Gaussian curvature. Váša and Rus [4] developed the DAME metric by computing the differences of oriented dihedral angles between two meshes. Torkhani et al. [5] proposed the TPDM metric based on the measurement of the distance between curvature tensors of two meshes. Dong et al. [6] proposed a MVQ metric by integrating roughness distortion and structure similarity.

Liu et al. [7] provided a survey on mesh saliency detection methods and their applications in computer graphics. The mesh saliency detection methods are classified into two categories, namely local contrast-based methods and global contrast-based methods [7]. Interested reader can find a detailed description of advantages and drawbacks of state-of-the-art mesh saliency detection methods in [7]. Lee et al. [8] developed a mesh saliency detection method using a center-surround operator on Gaussian-weighted mean curvatures. Song et al. [9] proposed a method for detecting mesh saliency by analyzing the properties of the log-Laplacian spectrum of the mesh. Limper et al. [10] proposed a mesh saliency detection method, named Local Curvature Entropy, by applying Shannon entropy to the mean curvature of vertices of 3D meshes. Nouri et al. [11] proposed a local surface descriptor based on adapative patches to characterize the perceptual saliency of each vertex of the mesh. Tao et al. [12] proposed to detect mesh saliency via manifold ranking in a descriptor space that is composed of patch descriptors based on Zernike coefficients. In this paper, we use three well-known mesh saliency detection methods [8–10] and TPDM metric [5] to investigate the added value of utilizing mesh saliency in MVQ assessment.

Several works [13–17] have been done to investigate the added value of including visual attention or computational visual saliency in IQMs. Moorthy et al. [13] proposed weighting local quality measurement by visual fixation and demonstrated improved performance for image quality assessment. Liu and Heynderickx [14] included visual attention in the design of IQMs based on eye-tracking data and achieved performance gain with the modified metrics. Farias and Akamine [15] concluded that the performance gain depends on the precision of visual saliency model and the distortion type when incorporating computational visual saliency models into image quality metrics. Liu et al. [16] investigated the effect of image content on the performance gain when adding visual attention in image quality assessment. Zhang et al. [17] used the visual saliency as a feature to compute the local quality map of distorted image and employed visual saliency as a weighting function to reflect the importance of local image region. In the community of MVQ assessment, however, there are relatively fewer works that investigated the benefit of integrating visual saliency into MVQ metrics. Nouri et al. [35] proposed a MVQ metric, Saliency-based Mesh Quality Index (SMQI), by using multiscale saliency map to compute local statistics that reflect the structural information. The literature [35] reveals that there exists a link between mesh saliency and MVQ assessment. Though the SMQI method [35] also involves mesh saliency in the MVQ metric, our work in this paper differs from the SMQI method in several aspects. The SMQI method uses a saliency map generated by the mesh saliency detection method in [12] to compute local structural distortions, which are then pooled via weighted Minkowski summation. We firstly generate a distortion map with the TPDM metric [5] and a saliency map with each of three state-of-the-art mesh saliency detection methods [8–10], and then weight the local distortion by the saliency value for each vertex of the mesh before pooling local distortions into an overall quality score. Thus, the role of mesh saliency in MVQ metric in our work is different from that in the SMQI method [35]. Moreover, our method inherits the merit of detecting perceptual distortions that reflect the mechanism of human visual system, and the merit of detecting perceptually important regions that reflect the preference of human perception.

Our contributions can be summarized as follows: Firstly, we investigate the benefit of integrating mesh saliency into MVQ assessment and propose a MVQ metric using a saliency weighting-based pooling strategy. Experimental results demonstrate the superiority and effectiveness of our metric. Secondly, we analyze the influence of surface area in the metric on the performance. The performance comparison reveals that it is inappropriate to include the surface area in the metric for the LIRIS/EPFL general-purpose database [1]. Thirdly, we assemble salient regions from individual saliency maps to generate a synthetic saliency map for saliency weighting. Experimental results show that the synthetic saliency map achieves better performance than individual saliency maps when used in our metric, and the performance gain is closely correlated with the similarity between the individual saliency maps.

3. Our proposed mesh visual quality metric

In this section, we propose a mesh visual quality metric by integrating mesh saliency into mesh visual quality assessment. As we mentioned in Section 1, we are inspired by the works [13–18] in image quality assessment and assume that distortions appearing in more salient regions of a mesh are more annoying. We use a saliency weighting-based pooling strategy at the pooling step to emphasize the distortions on the salient regions.

Among state-of-the-art MVQ metrics [1–6], the TPDM metric [5] correlates well with the human perception of mesh quality and is one of the best-performing MVQ metrics so far. The TPDM metric consists of a two-step computation process: firstly constructing a distortion map for the mesh, and then pooling local distortions via Minkowski summation. In our metric, given a reference mesh and a distorted mesh, we firstly use the TPDM metric [5] to generate a distortion map for the reference mesh, then generate a saliency map for the reference mesh with a mesh saliency detection method, and finally compute an overall quality score for the distorted mesh via the saliency weighting-based pooling of local distortions. The flowchart of our proposed mesh visual quality metric is illustrated in Fig. 1.

We follow the first-step computation process of the TPDM metric [5] to compute the local distortion for each vertex of the reference mesh. The TPDM metric computes the perceptual difference between the reference mesh and the distorted mesh based on the distance between curvature tensors of two meshes. It establishes a correspondence between the reference mesh and the distorted mesh to allow changed connectivity of distorted meshes. It performs the vertex projection from the reference mesh M_r to the distorted mesh M_d using the AABB tree data structure. Each vertex v_i in the reference mesh corresponds to a point v'_i in the distorted mesh. There are three vertices $v'_{i,1}$, $v'_{i,2}$ and $v'_{i,3}$ on the triangular facet T'_i that contains the point v'_i.

A number of excellent methods [36, 37] have been proposed to estimate the curvature tensor for polyhedral surfaces. By following the TPDM metric, we use the method proposed in [36] to estimate the curvature tensor of each vertex on the meshes M_r and M_d. Let $\mathscr{T}_{v_i}$ and $\mathscr{T}_{v'_{i,k}}$ $(1 \leq \mathrm{k} \leq 3)$ denote the curvature tensors of the vertices v_i and $v'_{i,k}$ respectively. The correspondence relationship between the principal curvature directions / amplitudes of $\mathscr{T}_{v_i}$ and $\mathscr{T}_{v'_{i,k}}$ is established based on the minimum angular distance criterion. For the minimum principal curvature direction γ_{min} of $\mathscr{T}_{v_i}$, the principal curvature direction γ'_1 of $\mathscr{T}_{v'_{i,k}}$ that has the smallest angular distance to γ_{min} is found as the corresponding direction. Accordingly, the minimum curvature amplitude κ_{min} of $\mathscr{T}_{v_i}$ corresponds to the curvature amplitude κ'_1 of $\mathscr{T}_{v'_{i,k}}$ that is associated to γ'_1.

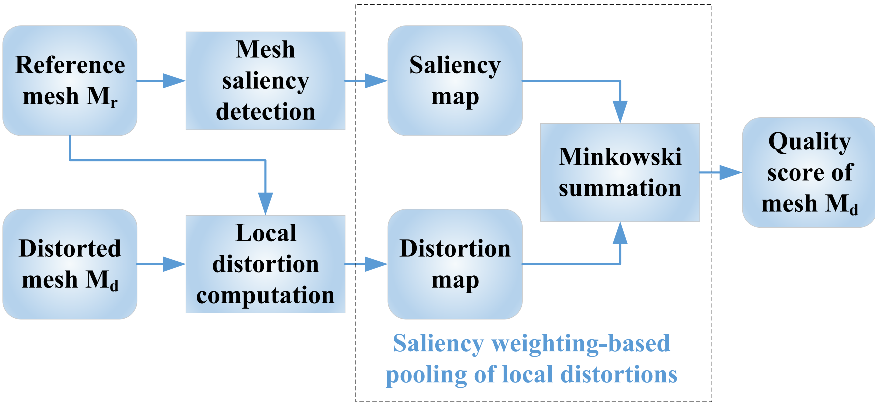


Fig. 1. Flowchart of our proposed mesh visual quality metric

By using the criterion, for the maximum principal curvature direction γ_{max} and maximum curvature amplitude κ_{max} of $\mathscr{T}_{v_i}$, the corresponding principal curvature direction γ_2' and curvature amplitude κ_2' of $\mathscr{T}_{v'_{i,k}}$ can be found in a similar way. Then the local distance $LPD_{v_i,v'_{i,k}}$ between the vertex v_i in the reference mesh and the vertex $v'_{i,k}$ of triangular facet T_i' in the distorted mesh is computed as:

$$LPD_{v_i,v'_{i,k}} = RW_i^{(\gamma)} \cdot RW_i^{(\kappa)} \cdot \left(\frac{\theta_{min}}{(\pi/2)} \delta_{\kappa_{min}} + \frac{\theta_{max}}{(\pi/2)} \delta_{\kappa_{max}} \right), \tag{1}$$

where θ_{min} is the angle between the principal curvature directions γ_{min} and γ_1', θ_{max} is the angle between the principal curvature directions γ_{max} and γ_2', $\delta_{\kappa_{min}}$ is the Michelson-like contrast of the curvature amplitudes κ_{min} and κ_1', and $\delta_{\kappa_{max}}$ is the Michelson-like contrast of the curvature amplitudes κ_{max} and κ_2'. $RW_i^{(\kappa)}$ and $RW_i^{(\gamma)}$ are the roughness-based coefficients [5]. On one hand, the principal curvature directions in the 1-ring neighborhood of v_i are projected on the tangent plane of v_i, and then a local roughness value LR_i^{γ} of v_i is computed as the sum of two angular standard deviations of the projected minimum and maximum curvature directions. After mapping all the local roughness values LR_i^{γ} to [0.1, 1.0], LR_i^{γ} is taken as the coefficient RW_i^{γ}. On the other hand, another local roughness value LR_i^{κ} of v_i is computed by normalizing the Laplacian of mean curvature amplitudes in the 1-ring neighborhood of v_i by the mean curvature of v_i. After mapping all the local roughness values LR_i^{κ} to [0.1, 1.0], LR_i^{κ} is taken as the coefficient $RW_i^{(\kappa)}$. A detailed description of $RW_i^{(\gamma)}$ and $RW_i^{(\kappa)}$ can be found in [5]. Let $b_k(v_i')$ denote the k-th barycentric coordinate of point v_i' within the triangular facet T_i'. The local distortion d_i of vertex v_i is computed through barycentric interpolation of three local distances between vertex v_i and vertices $v'_{i,1}$, $v'_{i,2}$ and $v'_{i,3}$ respectively:

$$d_i = \sum_{k=1}^{3} b_k(v_i') LPD_{v_i,v'_{i,k}}. \tag{2}$$

We compute the overall quality score of the distorted mesh M_d via saliency weighting-based pooling of local distortions. We firstly use the Minkowski exponent p to highlight the contributions of severe distortions to the quality judgement, then weight the local distortion by the saliency value for each vertex to emphasize the distortions on salient regions, and finally pool the weighted local distortions into an overall quality score. Our proposed MVQ metric TPDMVS is shown in Eq. (3):

$$TPDMVS = \left(\frac{1}{N} \sum_{i=1}^{N} s_i d_i^p \right)^{\frac{1}{p}}, \tag{3}$$

where s_i is the saliency value of vertex v_i and d_i is the local distortion of vertex v_i computed through Eq. (2). The Minkowski exponent p is set as $p = 4$. The Minkowski pooling method has been used in several MVQ metrics [1, 2, 5], where the Minkowski exponent p was chosen empirically in order to achieve the best performance. A typical value of p lies in the range [2.0, 4.0] as suggested in [2]. We investigated the influence of the value of p on the performance in a preliminary experiment and found that the overall best performance is achieved when p is set to 4. N is the number of vertices of the reference mesh. We generate a saliency map s, either individual saliency map or synthetic saliency map, for the reference mesh using the saliency methods [8–10] as we describe in Section 4 and Section 5. The saliency map is normalized so that the saliency value s_i of each vertex v_i of the mesh lies in the range [0, 1].

Note that we do not include the surface area in our metric while the TPDM metric [5] uses surface area to weight local distortion for each vertex. We provide an analysis of the influence of surface area on the performance of the metric in Section 5.3.

4. Mesh saliency detection methods

Many computational methods have been proposed to detect mesh saliency [7–12]. In this paper, we employ three well-known mesh saliency detection methods [8–10] to investigate the benefit of integrating mesh saliency into MVQ metric since they were demonstrated to be effective in graphics applications. We generate a saliency map for the reference mesh with each method. We denote the method in [8] as MS, the method in [9] as MSSP and the method in [10] as MSLCE. A detailed description of each method can be found in [8–10].

4.1. Mesh saliency (MS)

In [8], Lee et al. proposed a mesh saliency detection method MS using center-surround operators on Gaussian-weighted curvatures. The MS saliency method uses Taubin's method [37] to generate a mean curvature map $\mathscr{C}$ that maps from each vertex v of the mesh to its mean curvature $\mathscr{C}(v)$. Let $\mathscr{N}(v,\sigma) = \{x| \, ||x-v|| < \sigma, x \textit{ is a mesh point}\}$ denote the neighbourhood points for vertex v within Euclidean distance σ. The Gaussian-weighted average of mean curvature of vertex v, $G(\mathscr{C}(v),\sigma)$, is computed from the neighbourhood points. The saliency $\mathscr{S}(v)$ of vertex v is derived as the absolute difference between the Gaussian-weighted averages that are computed at fine and coarse scales. The saliency of vertex v at scale level t is defined as

$$\mathscr{S}_t(v) = |G(\mathscr{C}(v),\sigma_t) - G(\mathscr{C}(v),2\sigma_t)|, \tag{4}$$

where σ_t is the standard deviation of the Gaussian filter at scale t.

After each saliency map $\mathscr{S}_t$ at each scale level is normalized, the maximum saliency value M_t and the average $\bar{m}_t$ of local maxima excluding the global maximum at scale t are computed. Then the normalized saliency map $\mathscr{S}_t$ is multiplied by the factor $(M_t - \bar{m}_t)^2$. Finally, the final saliency map s of the mesh is derived by adding the saliency maps at all scales after applying a non-linear suppression operator $\mathscr{O}$ to each saliency map at each scale: $s = \sum_t \mathscr{O}(\mathscr{S}_t)$, where the suppression operator $\mathscr{O}$ suppresses the saliency maps with a large number of similar peaks while promoting the saliency maps with a small number of high peaks, and thus will reduce the number of salient vertices on the mesh.

4.2. Mesh saliency via spectral processing (MSSP)

Song et al. proposed a method MSSP to detect mesh saliency by analyzing the spectral properties of mesh [9]. The MSSP method firstly decomposes the geometric Laplacian matrix L of mesh M via eigenvalue decomposition: $L = B\Lambda B^T$, where Λ denotes a diagonal matrix whose entries are eigenvalues of L, and B denotes an orthogonal matrix whose columns are the eigenvectors of L. Let R denote a diagonal matrix whose entries are exponentials of the elements of spectral irregularity matrix, and W denote the distance-weighted adjacency matrix. A matrix S in spatial domain is generated via $S = BRB^T \cdot W$, where "$\cdot$" denotes the element-by-element multiplication. A saliency value $S(v_i)$ for vertex v_i is generated by summing all the elements in i-th row of matrix S. Then the spectral saliency value $S(v_i,t)$ of vertex v_i at scale t is computed in the Difference of Gaussian scale space. Let $k(i)$ denote the multiplicative factor computed from the one-ring neighbour vertices of vertex v_i. The scale saliency value $\tilde{S}(v_i,t)$ of vertex v_i at scale t is computed as the absolute difference between $S(v_i,k(i)t)$ and $S(v_i,t)$.

Since the eigenvalue decomposition of Laplacian matrix has a high computational complexity with respect to the number of vertices of the mesh, QSlim [38] is typically employed to simplify the original high-resolution mesh M to a low-resolution mesh M'. The saliency map $\tilde{S}'_t$ of the simplified mesh M' at each scale t is computed and then the saliency map $\tilde{S}_t$ of mesh M at scale t is obtained by mapping $\tilde{S}'_t$ to the mesh M using a k-d tree. After the saliency map $\tilde{S}_t$ of mesh M at each scale is obtained, a saliency map $\tilde{S}$ of mesh M is computed by adding the saliency maps $\tilde{S}_t$ at all scales and then smoothed using Laplacian smoothing. The final saliency map s of mesh M is produced by performing a logarithmic operation on $\tilde{S}$: $s = \log \tilde{S}$.

4.3. Mesh saliency analysis via local curvature entropy (MSLCE)

Limper et al. proposed a method MSLCE [10] to detect mesh saliency via computing local curvature entropy for each vertex of the mesh within the geodesic neighborhood. The mean curvature $\mathscr{C}(v_i)$ for each vertex v_i of the mesh is firstly computed in the same way as in [8]. By considering the neighbourhood vertices $\mathscr{N}(v_i,r) = \{v'_0, v'_1, \cdots, v'_m\}$ of vertex v_i within geodesic distance r, the curvature values of $\mathscr{N}(v_i,r)$ are partitioned into n_1 bins using a uniform sampling, which results in a set of discrete symbols $\{\rho_0, \rho_1, \cdots, \rho_{n_1}\}$. Let A_k denote the surface area of each vertex v'_k within $\mathscr{N}(v_i,r)$. The probability of symbol ρ_j ($0 \leq \mathrm{j} \leq n_1$) within local neighbourhood of vertex v_i is computed by the surface area and the affiliation of each neighbourhood vertex.

By applying Shannon entropy to the set of symbols ρ_j, the saliency value of vertex v_i is computed as its local curvature entropy. In order to detect salient regions at multiple scales, the radius parameter r is varied up to a maximum value r_{max}. The saliency maps are computed at multiple levels $l_0, \cdots, l_{t_0-1}$, where the radius parameter for each level l_t is defined as $r_t = 2^{-t} r_{max}$. A final saliency map s is generated for the mesh by combining the saliency maps at all levels using an average weighting scheme.

4.4. Analysis of mesh saliency detection methods

In this section, we perform an analysis of three mesh saliency detection methods [8–10] with the Dinosaur model and the RockerArm model in the LIRIS/EPFL general-purpose database [1]. We generate a normalized saliency map for the reference mesh of each model with each mesh saliency detection method, and provide a visual illustration of each saliency map in Fig. 2. The colormap is used to map the saliency value to RGB color for each vertex of the mesh. As indicated by Fig. 2(e), for each vertex in the mesh, the red color represents a high saliency value, the green color represents a median saliency value, and the blue color represents a low saliency value. When the saliency value of

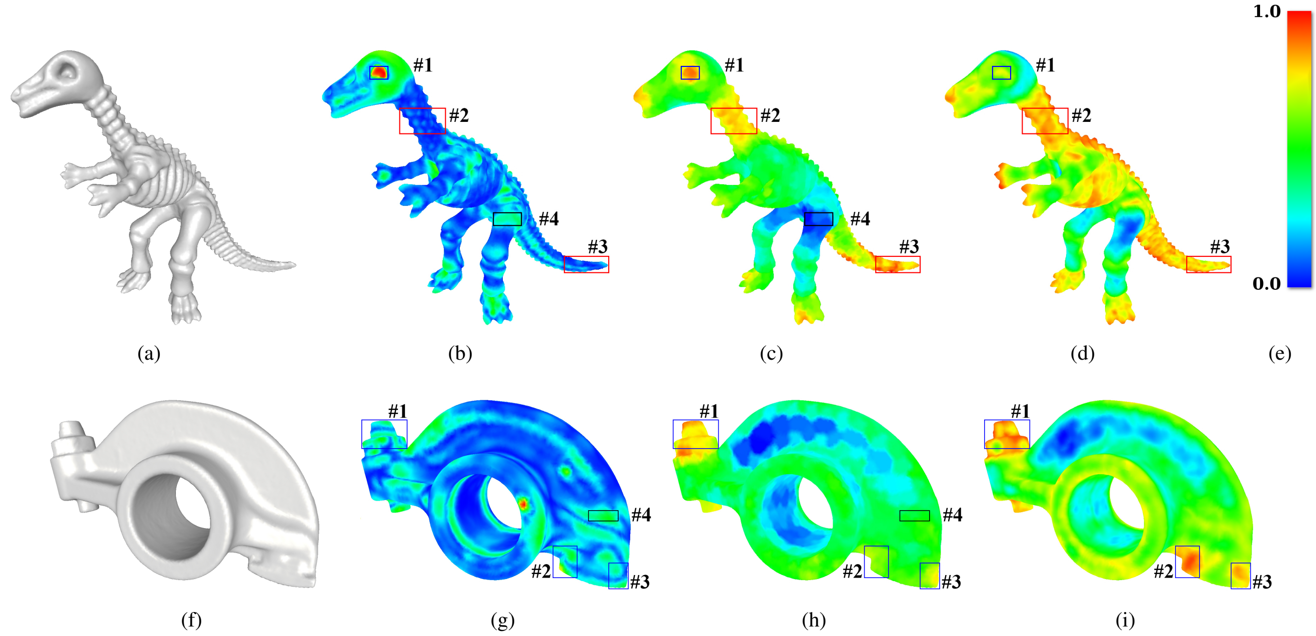


Fig. 2. Visual illustration of individual saliency maps on two models. (a) Reference mesh of the Dinosaur model. (b)-(d) Saliency map of MS, MSSP and MSLCE respectively on the Dinosaur model. (e) Rainbow colormap. (f) Reference mesh of the RockerArm model. (g)-(i) Saliency map of MS, MSSP and MSLCE respectively on the RockerArm model.

a vertex is higher than the mean value of the saliency map of the mesh, we consider the vertex as salient in the mesh.

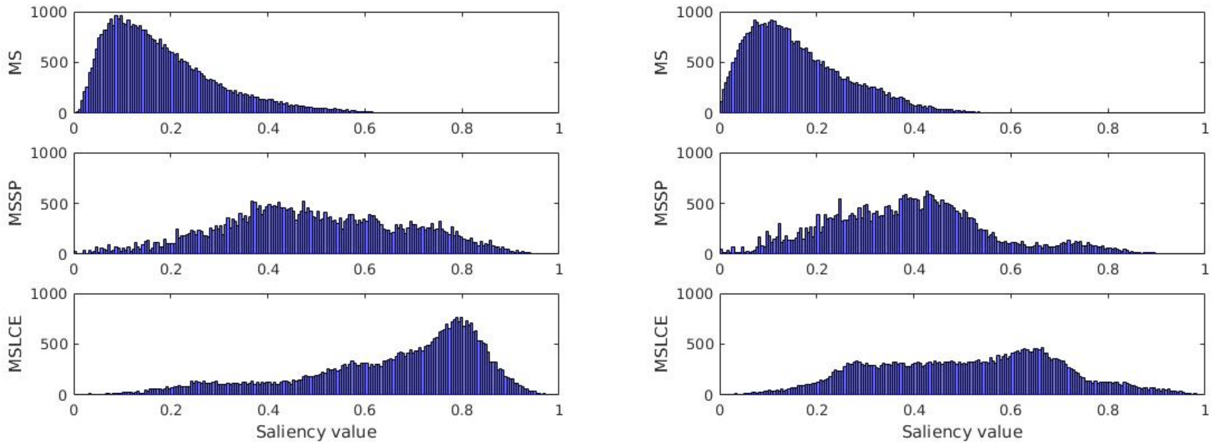
From Fig. 2, we observe that, on the same model, the saliency map of MSLCE is overall warmer than the saliency map of MSSP while the saliency map of MSSP is overall warmer than the saliency map of MS. We also observe that three saliency methods detect some common vertices as salient at some regions though the salient vertices that each saliency method [8–10] detects are not exactly the same. Particularly, there is a relatively higher similarity between the saliency maps of MSSP and MSLCE since MSSP and MSLCE detect more common vertices as salient among the three saliency methods. On the Dinosaur model, all the three saliency methods detect the vertices at the #1 region (the left eye region) as salient, as shown in the blue rectangles of Fig. 2(b) - Fig. 2(d). Besides, at some other regions, such as the #2 region (the neck region) and the #3 region (the tail region) as shown in the red rectangles of Fig. 2(b) - Fig. 2(d), both MSSP and MSLCE detect the vertices as salient which however are detected as non-salient by MS. On the RockerArm model, at the #1, #2, and #3 regions as shown in the blue rectangles of Fig. 2(g) - Fig. 2(i), both MSSP and MSLCE detect generally high saliency while MS detects high saliency only at some parts of these regions and low saliency at the remaining part of these regions.

In order to observe the statistical distribution characteristics of each saliency map, we plot a histogram of each saliency map generated by three saliency methods on two models in Fig. 3. We list the statistical characteristics of three individual saliency maps on the Dinosaur model and the RockerArm model respectively in Table 1 and Table 2, where *Mean* and *Std* represent the mean and standard deviation of the saliency map. We sort the saliency map in ascending order. Then Q_1, Q_2 and Q_3 stand for the first quartile, the second quartile, and the third quartile of the sorted saliency map respectively. We observe that three saliency maps show different statistical distributions on the same model. When comparing the statistical characteristics of three saliency maps in terms of Q_1, Q_2, Q_3 and *Mean*, on either the Dinosaur model or the RockerArm model, MSLCE always has greater value than MSSP while MSSP always has greater value than MS. Thus, the saliency map of MSLCE has overall greater values than the saliency map of MSSP while the saliency map of MSSP has overall greater values than the saliency map of MS. This conclusion is consistent with the visual illustration in Fig. 2.

Table 1. Statistical characteristics of three individual saliency maps on the Dinosaur model

Saliency map	Q_1	Q_2	Q_3	*Mean*	*Std*
MS	0.0959	0.1574	0.2442	0.1859	0.1236
MSSP	0.3651	0.4821	0.6316	0.4938	0.1880
MSLCE	0.5497	0.7059	0.7958	0.6526	0.1884

We use the Pearson linear correlation coefficient (PLCC) to measure the similarity between two saliency maps on each model. The PLCC has been used to evaluate the similarity



(a) (b)

Fig. 3. Histograms of saliency maps of three saliency methods on two models. (a) Dinosaur model. (b) RockerArm model

Table 2. Statistical characteristics of three individual saliency maps on the RockerArm model

Saliency map	Q_1	Q_2	Q_3	*Mean*	*Std*
MS	0.0835	0.1411	0.2251	0.1642	0.1065
MSSP	0.2744	0.3896	0.4864	0.3935	0.1679
MSLCE	0.3588	0.5202	0.6527	0.5098	0.1888

Table 3. PLCC values (%) for each pair of saliency maps on two models

	Dinosaur model	RockerArm model
MS vs. MSSP	-1.95	36.34
MS vs. MSLCE	-19.92	34.13
MSSP vs. MSLCE	63.66	79.80

between two saliency maps in the image saliency detection [7, 39, 40]. We list the PLCC values for each pair of saliency maps on two models in Table 3. The PLCC value lies in the range [-1, 1], and a greater PLCC value indicates a higher similarity between two saliency maps. We observe that the rank of three PLCC values is the same for two models though there is a significant difference in the PLCC values between two models. On either the Dinosaur model or the RockerArm model, the PLCC value between the saliency maps of MS and MSLCE is smallest, the PLCC value between the saliency maps of MSSP and MSLCE is greatest, and the PLCC value between the saliency maps of MS and MSSP is median. This indicates that, relatively speaking, the similarity between the saliency maps of MSSP and MSLCE is greatest, the similarity between the saliency maps of MS and MSLCE is lowest, and the similarity between the saliency maps of MS and MSSP is median.

5. Experimental results and analysis

5.1. Experiment protocol

In this paper, we use the LIRIS/EPFL general-purpose database [1] as a test bed to validate the superiority and effectiveness of our MVQ metric. The LIRIS/EPFL general-purpose database consists of four models, and for each model there are one reference mesh and 21 distorted meshes. The distorted meshes are generated by applying either noise addition or smoothing distortion with different strengths either locally or globally to the reference mesh. The observer was asked to remember the mesh that was considered to have the worst quality among the distorted meshes. Then the observer provided an opinion score that reflects the degree of perceived distortion for each mesh of each model, including the reference mesh and distorted meshes. The opinion score ranges from 0 (best quality) to 10 (worst quality). Twelve observers participated in the subjective evaluation. Finally, a normalized Mean Opinion Score (MOS) was computed for each mesh by averaging the opinion scores of all the observers.

We use our metric TPDMVS to compute objective quality scores for the meshes in the LIRIS/EPFL general-purpose database. We evaluate the performance of our metric by measuring the correlation between the quality scores and MOSs with two coefficients: Pearson linear correlation coefficient (PLCC) that measures the prediction accuracy of quality metric and Spearman rank-order correlation coefficient (SROCC) that measures the prediction monotonicity of quality metric [27, 41]. Both values of PLCC and SROCC range from -1 to 1, where -1 indicates fully negative correlation, 1 indicates fully positive correlation, and 0 indicates no correlation. Since the nonlinear quality rating compression may exist at the extremes of the test range during the subjective testing, there is typically a nonlinearity between the subjective ratings and objective predictions [42]. Thus, in many works on both mesh quality metrics and image quality metrics [1, 3, 5, 6, 43], a psychometric fitting was performed between the objective quality scores and MOS values to remove the nonlinearity. In this paper, we also conduct a psychometric fitting to remove the nonlinearity between the set of objective quality scores and the set of MOS values before computing the correlation coefficients. We apply the cumulative Gaussian function [5, 44] for psychometric fitting:

$$g(a,b,Q) = \frac{1}{\sqrt{2\pi}} \int_{a+bQ}^{+\infty} e^{-(t^2/2)} dt, \tag{5}$$

where Q is the objective quality score. Each mesh in the LIRIS/EPFL general-purpose database [1] has a MOS value and a calculated objective quality score, both of which constitute a sample pair. We conduct the psychometric fitting on the sample pairs using the nonlinear least squares method and thus obtain the values for parameters a and b. In this paper, we use the curve fitting toolbox of Matlab to implement the psychometric fitting. After obtaining the values for a and b, we transform the set of objective quality values to a set of predicted MOS values, and then compute the correlation coefficients between the predicted MOS values and the actual MOS values to evaluate the performance of the metric. Note that g is assigned the actual MOS value during the psychometric fitting and will be the predicted MOS value after the values of a and b are determined.

We provide the correlation coefficients of our metric in three cases. In each case, we use one of the three saliency methods described in Section 4 to generate a saliency map s for each reference mesh in the LIRIS/EPFL general-purpose database and then generate quality scores for the distorted meshes using the saliency map s in our metric through Eq. (3). Note that the MS saliency method [8] takes a long time to compute the saliency map particularly for the high-resolution mesh. Thus, in the case of MS saliency method [8], we use QSlim [38] to simplify the original mesh M to a simplified mesh M', and then generate a saliency map s' for M'. The saliency map s of mesh M is finally obtained using a closest point matching strategy as in [9].

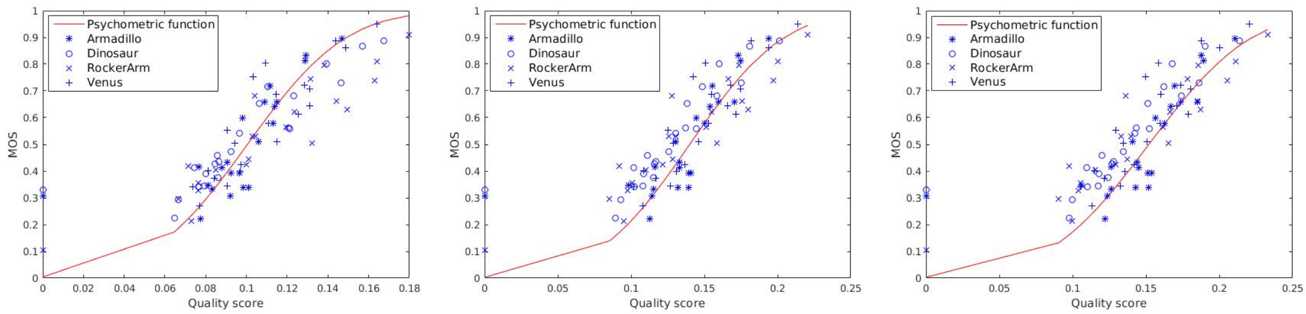
5.2. Performance comparison

We compare our metric TPDMVS with state-of-the-art MVQ metrics, including Hausdorff Distance (HD) [45], Root Mean Square Error (RMS) [45], GL1 [30], GL2 [31], SF [33], $3DWPM_1$ [32], $3DWPM_2$ [32], MSDM [1], MSDM2 [2], FMPD [3], DAME [4], TPDM [5], Dong [6]. We obtain the results of existing metrics shown in Table 4 from literatures [3–5, 24, 25] and the erratum of MVQ metrics [46]. The performance values of the TPDM metric are generated with the code released online [5], which are officially confirmed by the authors. Table 4 lists the values of PLCC and SROCC for our metric with the three saliency methods [8–10] as well as state-of-the-art metrics on the LIRIS/EPFL general-purpose database. TPDMVS(MS) indicates the performance of our metric with the MS saliency method [8], TPDMVS(MSSP) indicates the performance of our metric with the MSSP saliency method [9], and TPDMVS(MSLCE) indicates the performance of our metric with the MSLCE saliency method [10]. From Table 4, we observe that our metric with each saliency method achieves significant performance gain over the TPDM metric [5] and achieves the best performance among all the metrics in Table 4. This indicates that incorporating mesh saliency in mesh quality metric can improve the performance of quality prediction, and thus supports the assumption that we made in Section 1.

From Table 4, we also observe that our metric shows similar performances for three saliency methods despite the significant differences in the generated saliency maps as illustrated in Fig. 2 and Fig. 3. The reason may be that the performance of the TPDM metric [5] is already relatively high as shown in Table 4 and there is a performance bottleneck for the LIRIS/EPFL general-purpose database [1] that consists of a small number of meshes. Note that any of the existing subjective image quality databases [34, 47–50] consists of hundreds or even thousands of image samples while the LIRIS/EPFL general-purpose database which is the largest available subjective mesh quality database consists of only 88 mesh samples. Even though it is hard to achieve further performance gain over the TPDM metric, our proposed metric by incorporating mesh saliency still achieves a performance improvement and the performances for three saliency maps are similar. As pointed out in [18], how human attention affects the perception of visual quality is still unknown and there is a lack of solid theoretical basis for the investigation on the relationship between human attention and visual quality. Thus, it is still difficult to explain in a theoretical way how much the performance improvement would be when incorporating human attention or visual saliency in a visual quality metric. In this paper, we have demonstrated the added value of mesh saliency empirically by incorporating three well-known saliency methods [8–10] in the mesh quality metric in a similar way as previous scholars did in the community of image quality assessment [13–18].

For each saliency method, we use our metric to compute quality scores for all the meshes in the LIRIS/EPFL general-purpose database [1] and then perform psychometric fitting between the quality scores and MOSs using the cumulative Gaussian psychometric function in Eq. (5). We plot the psychometric function curves with scatter plots of *QualityScore-MOS* pairs for three saliency methods in Fig. 4, where we observe that the *QualityScore-MOS* pairs are fitted well by the psychometric function curve for each saliency method.

In order to demonstrate the generalization capability of our metric on a variety of models, we use our metric TPDMVS(MS) to compute the quality scores of some representative distorted models in the LIRIS/EPFL general-purpose database [1]. For each of the four 3D objects in the LIRIS/EPFL general-purpose database, we select four distorted models with various distortion levels which are generated by applying the smoothing filter or adding noise with different strengths either locally or globally on the reference model. As stated in [1], these distortions reflect the distortions



(a) (b) (c)

Fig. 4. The psychometric function curves with scatter plots of quality scores versus MOSs for the meshes in the LIRIS/EPFL general-purpose database for each saliency method. (a) MS saliency method. (b) MSSP saliency method. (c) MSLCE saliency method.

that generally appear in common mesh processing operations, such as mesh simplification, mesh compression, and mesh watermarking. We illustrate the reference model and distorted models of each 3D object in Fig. 5 and provide a description for each distorted model on how the distortion is applied on the reference model in Table 5. At the subcaptions of Fig. 5, we provide the MOS value and the quality score (QS) computed by our metric TPDMVS(MS) for each distorted model. We denote the distorted models of Venus as V_1, V_2, V_3, V_4, the distorted models of RockerArm as R_1, R_2, R_3, R_4, the distorted models of Armadillo as A_1, A_2, A_3, A_4, and the distorted models of Dinosaur as D_1, D_2, D_3, D_4, respectively. From Fig. 5, we observe that the MOS values of four distorted models have exactly the same rankings with the QS values of four distorted models for each 3D object despite the variations in the distortion type, distortion area and distortion strength in the distorted models. This indicates that our metric has a good generalization capability in evaluating the visual quality of different models with various distortions. Note that though we use the MS saliency method [8] to demonstrate the generalization capability of our metric, we can find a similar consistency between the MOS values and QS values of the distorted models when using the other two saliency methods [9, 10] in our metric.

5.3. Analysis of the influence of surface area

In [5], the surface area is used as a weighting coefficient for the local distortion of each vertex in the TPDM metric. However, we do not include surface area in our metric in Eq. (3). The LIRIS/EPFL general-purpose database [1] involves two types of distortion: noise addition and smoothing. The smoothing operation usually introduces perceptually more significant distortion on the rough regions than on the smooth regions. The surface areas on the rough regions are generally smaller than the surface areas on the smooth regions because the rough regions generally need small-area triangles to characterize highly curved shape while the smooth regions typically consist of large-area triangles to characterize flat shape. Thus, in the case of smoothing distortion, weighting the local

Table 4. PLCC and SROCC (%) of our metric with three saliency methods as well as state-of-the-art metrics on the LIRIS/EPFL general-purpose database

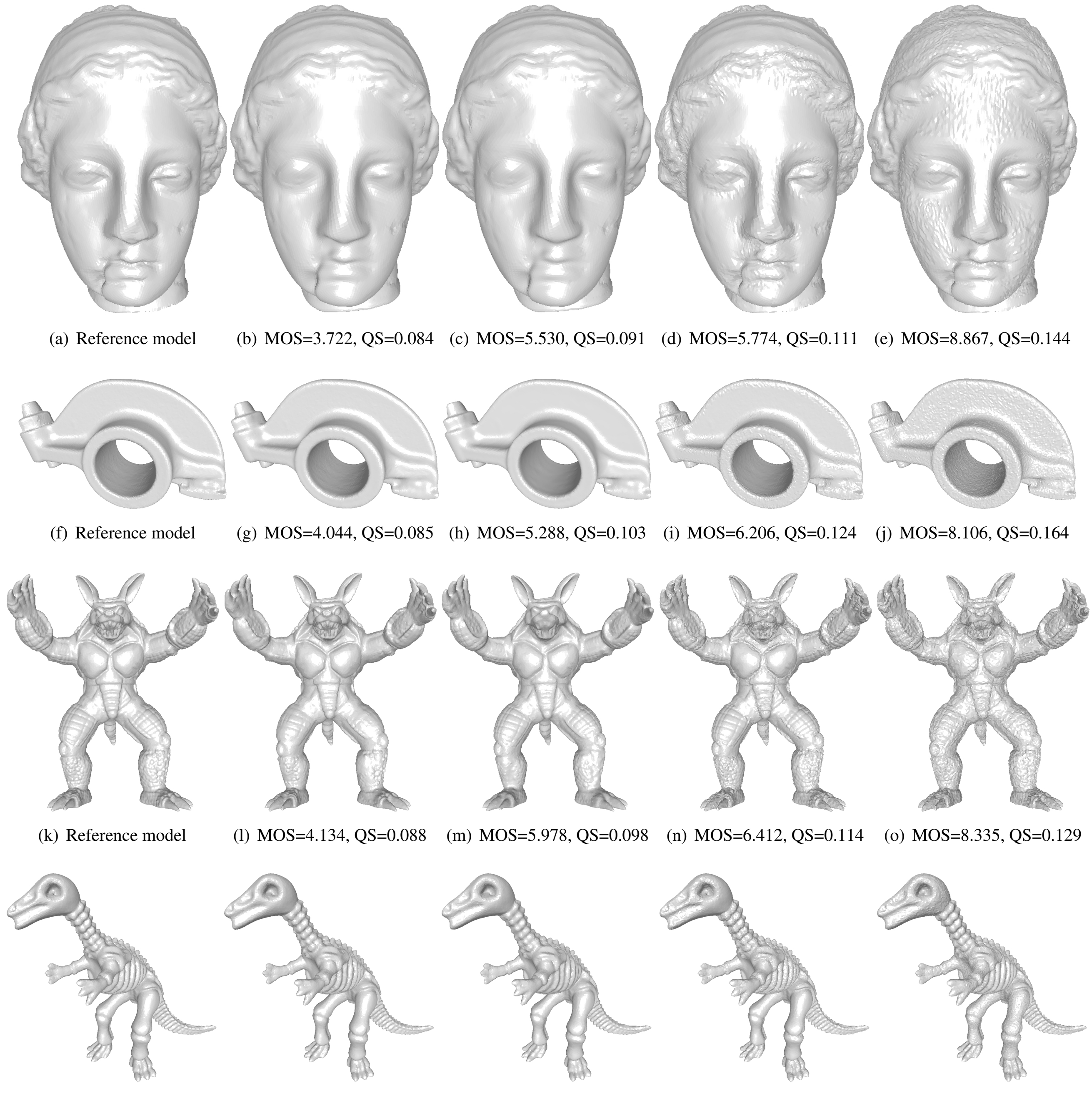
Metrics	PLCC	SROCC
HD	11.4	13.8
RMS	28.1	26.8
GL1	35.5	33.1
GL2	42.4	39.3
SF	7.0	15.7
$3DWPM_1$	61.8	69.3
$3DWPM_2$	49.6	49.0
MSDM	75.0	73.9
MSDM2	81.4	80.4
FMPD	83.5	81.9
DAME	75.2	76.6
TPDM	84.1	84.3
Dong	87.7	86.6
TPDMVS(MS)	89.0	89.3
TPDMVS(MSSP)	89.6	89.2
TPDMVS(MSLCE)	89.4	89.3

distortion by the surface area will lead to overemphasis on the local distortions on the smooth regions and then result in overestimation of quality degradation of the mesh. Finally, the correlation between the quality scores and MOSs of the meshes in the entire database may decline to some extent. If the surface area is used as a weighting coefficient for the local distortion, the metric incorporating the surface area will be

$$TPDMVS\text{-}W = \left(\sum_{i=1}^{N} w_i s_i d_i^p \right)^{\frac{1}{p}}, \qquad (6)$$

where $w_i = a_i / \sum_{i=1}^{N} a_i$ is the surface area weighting coefficient of vertex v_i with a_i one-third of the total areas of all the incident facets of vertex v_i in the reference mesh.

We use the TPDMVS-W metric with three saliency methods to generate quality scores for the meshes and provide a performance comparison among the TPDM metric [5],



(a) Reference model (b) MOS=3.722, QS=0.084 (c) MOS=5.530, QS=0.091 (d) MOS=5.774, QS=0.111 (e) MOS=8.867, QS=0.144

(f) Reference model (g) MOS=4.044, QS=0.085 (h) MOS=5.288, QS=0.103 (i) MOS=6.206, QS=0.124 (j) MOS=8.106, QS=0.164

(k) Reference model (l) MOS=4.134, QS=0.088 (m) MOS=5.978, QS=0.098 (n) MOS=6.412, QS=0.114 (o) MOS=8.335, QS=0.129

(p) Reference model (q) MOS=3.429, QS=0.079 (r) MOS=4.278, QS=0.084 (s) MOS=6.540, QS=0.106 (t) MOS=8.011, QS=0.139

Fig. 5. MOS values versus quality scores of some representative distorted models in the LIRIS/EPFL general-purpose database. (a)-(e) The reference model and four distorted models V_1, V_2, V_3, V_4 of Venus. (f)-(j) The reference model and four distorted models R_1, R_2, R_3, R_4 of RockerArm. (k)-(o) The reference model and four distorted models A_1, A_2, A_3, A_4 of Armadillo. (p)-(t) The reference model and four distorted models D_1, D_2, D_3, D_4 of Dinosaur.

the TPDMVS-W metric and the TPDMVS metric on the LIRIS/EPFL general-purpose database in Table 6. From Table 6, we observe that, for each saliency method, the TPDMVS metric always achieves better performance than the TPDMVS-W metric while the TPDMVS-W metric always achieves better performance than the TPDM metric. The comparison validates the effectiveness of the saliency weighting-based pooling strategy and also reveals that it is inappropriate to include the surface area in the metric for the LIRIS/EPFL general-purpose database.

Table 5. Descriptions on the generation of the distorted models from the reference models

Model		MOS	QS	Distortions
Venus	V_1	3.722	0.084	Applying the Taubin smoothing filter with 20 iterations on the rough areas
	V_2	5.530	0.091	Applying the Taubin smoothing filter with 30 iterations on the rough areas
	V_3	5.774	0.111	Adding noise on the intermediately rough areas
	V_4	8.867	0.144	Adding noise on the smooth areas
RockerArm	R_1	4.044	0.085	Applying the Taubin smoothing filter with 20 iterations on the rough areas
	R_2	5.288	0.103	Applying the Taubin smoothing filter with 15 iterations uniformly on the surface
	R_3	6.206	0.124	Adding noise on the rough areas
	R_4	8.106	0.164	Adding noise uniformly on the surface
Armadillo	A_1	4.134	0.088	Applying the Taubin smoothing filter with 10 iterations on the intermediately rough areas
	A_2	5.978	0.098	Applying the Taubin smoothing filter with 15 iterations on the rough areas
	A_3	6.412	0.114	Adding noise on the rough areas
	A_4	8.335	0.129	Adding noise uniformly on the surface
Dinosaur	D_1	3.429	0.079	Applying the Taubin smoothing filter with 20 iterations on the rough areas
	D_2	4.278	0.084	Applying the Taubin smoothing filter with 30 iterations on the rough areas
	D_3	6.540	0.106	Adding noise on the intermediately rough areas
	D_4	8.011	0.139	Adding noise on the smooth areas

Table 6. Performance comparison among the TPDM, TPDMVS-W and TPDMVS metrics on the LIRIS/EPFL general-purpose database

Metric		PLCC	SROCC
TPDM		84.1	84.3
MS	TPDMVS-W	87.5	88.3
	TPDMVS	89.0	89.3
MSSP	TPDMVS-W	89.0	88.5
	TPDMVS	89.6	89.2
MSLCE	TPDMVS-W	88.2	87.5
	TPDMVS	89.4	89.3

5.4. Synthetic saliency maps

As we analyzed in Section 4.4, there is a significant difference among the saliency maps generated by the three saliency methods [8–10]. When some vertices are detected as salient by one saliency method, they may be detected as non-salient by the other two saliency methods. In spite of the difference among three saliency maps, each saliency method leads to performance gain when used in our metric, as we described in Section 5.2. Therefore, we come up with a question naturally: is it possible to further improve the performance using the synthetic saliency map generated by assembling the salient regions from different saliency maps? We firstly assume that better performance can be obtained if the salient regions from individual saliency maps are assembled together. In order to validate the assumption, we firstly merge the saliency maps by selecting the relatively higher saliency value for each vertex of the mesh and then observe if there is any performance gain over each individual saliency map when using the synthetic saliency map in our metric. Since three saliency maps have different statistical distributions, we standardize each saliency map s by transforming it to have mean of zero and standard deviation of one:

$$s_i' = (s_i - s_{mean})/s_{std}, \tag{7}$$

where s_i is the saliency value for vertex v_i before standardization, s_i' is the saliency value after standardization, s_{mean} and s_{std} are the mean and standard deviation of the saliency map s respectively. We use the *max* function to assign the higher saliency value from the standardized saliency maps as the saliency value for each vertex. Let $s^{a'}$ and $s^{b'}$ denote two standardized saliency maps obtained via Eq. (7), the synthetic saliency map is generated by applying the *max* function to each element value of saliency maps $s^{a'}$ and $s^{b'}$

$$s_i^{m'} = max(s_i^{a'}, s_i^{b'}), \tag{8}$$

where $s_i^{a'}$ and $s_i^{b'}$ are the saliency values for vertex v_i in the saliency maps $s^{a'}$ and $s^{b'}$ respectively, and $s_i^{m'}$ is the saliency value for vertex v_i in the synthetic saliency map. The saliency values in the synthetic saliency map are normalized into the range [0, 1] before the synthetic saliency map is used in our metric.

Table 7. Statistical characteristics of the synthetic saliency maps on the Dinosaur model

Saliency map	Q_1	Q_2	Q_3	*Mean*	*Std*
MS-MSSP	0.1637	0.2397	0.3277	0.2504	0.1171
MS-MSLCE	0.1969	0.2596	0.3028	0.2555	0.1030
MSSP-MSLCE	0.4497	0.5795	0.6716	0.5527	0.1723
MS-MSSP-MSLCE	0.2117	0.2711	0.3336	0.2741	0.1061

We provide a visual illustration of the synthetic saliency maps on the Dinosaur model and the RockerArm model in the LIRIS/EPFL general-purpose database [1] in Fig. 6. MS-MSSP indicates the synthetic saliency map by merging the saliency maps of MS and MSSP, MS-MSLCE indicates the

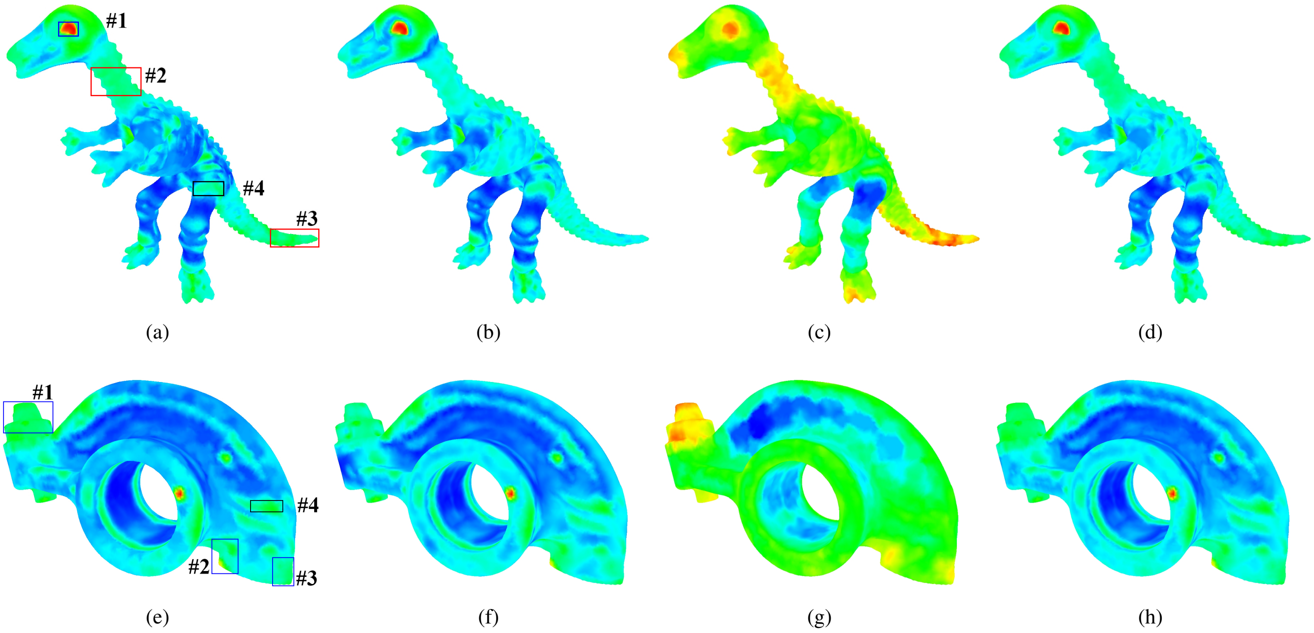


Fig. 6. Visual illustration of synthetic saliency maps on two models. (a)-(d) Synthetic saliency maps MS-MSSP, MS-MSLCE, MSSP-MSLCE, MS-MSSP-MSLCE respectively on the Dinosaur model. (e)-(h) Synthetic saliency maps MS-MSSP, MS-MSLCE, MSSP-MSLCE, MS-MSSP-MSLCE respectively on the RockerArm model.

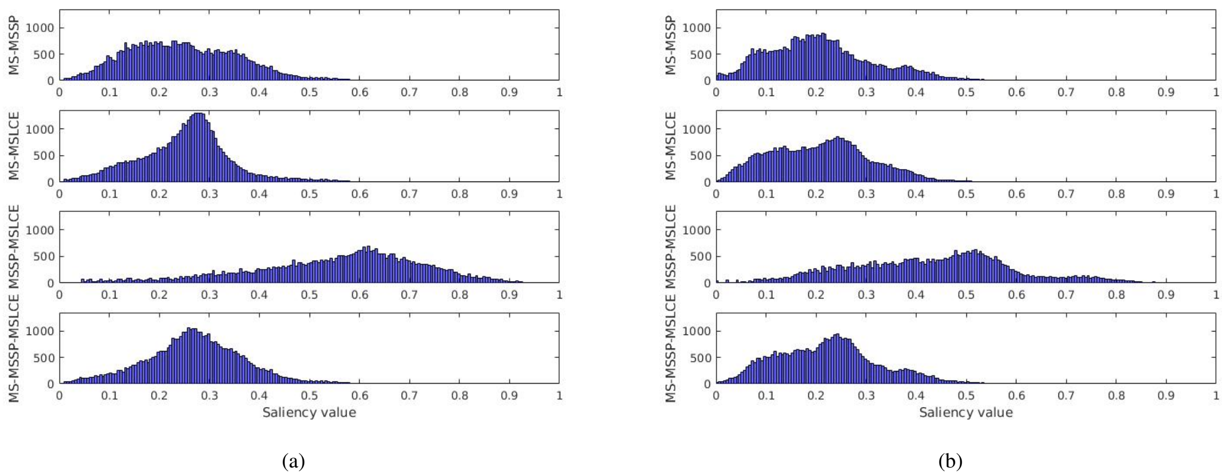


Fig. 7. Histograms of synthetic saliency maps on two models. (a) Dinosaur model. (b) RockerArm model

Table 8. Statistical characteristics of the synthetic saliency maps on the RockerArm model

Saliency map	Q_1	Q_2	Q_3	*Mean*	*Std*
MS-MSSP	0.1336	0.2001	0.2700	0.2105	0.1066
MS-MSLCE	0.1311	0.2110	0.2755	0.2107	0.1025
MSSP-MSLCE	0.3128	0.4416	0.5370	0.4328	0.1659
MS-MSSP-MSLCE	0.1483	0.2233	0.2831	0.2247	0.1028

synthetic saliency map by merging the saliency maps of MS and MSLCE, MSSP-MSLCE indicates the synthetic saliency map by merging the saliency maps of MSSP and MSLCE, and MS-MSSP-MSLCE indicates the synthetic saliency map by merging the saliency maps of MS, MSSP, and MSLCE. In order to determine if a vertex is salient on the mesh for each synthetic saliency map, we plot a histogram of each synthetic saliency map on two models in Fig. 7 and list the statistical characteristics of the synthetic saliency maps on the Dinosaur model and the RockerArm model respectively in Table 7 and Table 8. From Fig. 6, we observe that the synthetic saliency map MSSP-MSLCE is overall warmer than the other three

Table 9. Performance comparison between the individual saliency maps and the synthetic saliency maps on the LIRIS/EPFL general-purpose database

Saliency map	PLCC	SROCC
MS	89.0	89.3
MSSP	89.6	89.2
MSLCE	89.4	89.3
MS-MSSP	89.8	90.8
MS-MSLCE	90.1	91.2
MSSP-MSLCE	89.7	89.5
MS-MSSP-MSLCE	89.9	91.2

synthetic saliency maps on two models. This observation is consistent with the histograms of synthetic saliency maps in Fig. 7, where the saliency values of MSSP-MSLCE are generally greater than the saliency values of the other three synthetic saliency maps on either the Dinosaur model or the RockerArm model. When comparing the statistical characteristics of the synthetic saliency maps in terms of Q_1, Q_2, Q_3 and *Mean* in Table 7 and Table 8, we also observe that MSSP-MSLCE always has significantly greater value than the other three synthetic saliency maps on both models.

By comparing Fig. 2 and Fig. 6, we observe that the salient regions on each individual saliency map are preserved well on the synthetic saliency maps. We use the synthetic saliency map MS-MSSP to elaborate the preservation of salient regions on the synthetic saliency map on two models, and a similar phenomenon can also be observed for both MS-MSLCE and MSSP-MSLCE.

- On the Dinosaur model, MS detects high saliency at the #1 region (in the blue rectangle) and the #4 region (in the black rectangle), and low saliency at the #2 and #3 regions (in the red rectangles) as shown in Fig. 2(b). MSSP detects high saliency at the #1, #2 and #3 regions, and low saliency at the #4 region as shown in Fig. 2(c). Finally, the synthetic saliency map MS-MSSP shows high saliency at the #1, #2, #3 and #4 regions in Fig. 6(a).

- On the RockerArm model, MS detects high saliency at the #4 region (in the black rectangle) and low saliency at some parts of the #1, #2, and #3 regions (in the blue rectangles) as shown in Fig. 2(g). MSSP detects generally high saliency at the #1, #2, and #3 regions and median saliency at the #4 region as shown in Fig. 2(h). Finally, the synthetic saliency map MS-MSSP shows high saliency at the #1, #2, #3, and #4 regions as shown in Fig. 6(e).

We provide a performance comparison between the individual saliency maps and the synthetic saliency maps on the LIRIS/EPFL general-purpose database [1] in Table 9. From Table 9, we observe that all the synthetic saliency maps achieve performance gain over each individual saliency map, and MS-MSLCE has the best performance among all the synthetic saliency maps. Among the three synthetic saliency maps that merge only two individual saliency maps, the performance gain achieved by MS-MSLCE over corresponding individual saliency maps (MS and MSLCE) is the greatest while the performance gain achieved by MSSP-MSLCE over corresponding individual saliency maps (MSSP and MSLCE) is the least. As we analyzed in Section 4.4, the similarity between the saliency maps of MS and MSLCE is the lowest while the similarity between the saliency maps of MSSP and MSLCE is the highest. So we conclude that there is a close correlation between the performance gain of the synthetic saliency map over individual saliency maps and the similarity between the individual saliency maps. Specifically, our analysis based on three saliency methods indicates that the lower the similarity between two individual saliency maps is, the greater the performance gain of the synthetic saliency map over the individual saliency maps will be. From Table 9, we also observe that MS-MSSP-MSLCE does not achieve better performance than MS-MSLCE. The reason is that there is already a high similarity between the saliency maps of MSSP and MSLCE, and thus it is hard to achieve performance gain over MS-MSLCE by further merging the synthetic saliency map MS-MSLCE with the saliency map of MSSP. Due to a lack of sufficient knowledge of human visual system [13–18], a perfect theoretic interpretation for the performance gain of the synthetic saliency map over individual saliency maps is not yet available. However, we believe that our work in this paper will facilitate the investigation on how human attention or visual saliency affects the perception of mesh quality and on the correlation analysis among different mesh saliency methods.

Based on the aforementioned analysis, we draw the following conclusions: (1) After standardizing two individual saliency maps and applying the *max* function to the standardized saliency maps, the salient regions of each individual saliency map will be preserved in the synthetic saliency map. (2) The synthetic saliency map achieves better performance than each individual saliency map when used in our metric. (3) There is a close correlation between the performance gain of the synthetic saliency map over the individual saliency maps and the similarity between individual saliency maps. If the similarity between two individual saliency maps is lower, the performance gain of the synthetic saliency map over the individual saliency maps will be greater.

6. Conclusion

In this paper, we have proposed a mesh visual quality metric using a saliency weighting-based pooling strategy. We have demonstrated the superiority and effectiveness of our metric with three well-known mesh saliency detection methods. The performance comparison shows that our metric with any of the three saliency maps achieves better performance than state-of-the-art MVQ metrics. The experimental result reveals that it is inappropriate to include the surface area in the metric for the LIRIS/EPFL general-purpose database.

Our analysis shows that there is a significant difference in the statistical distribution for the saliency maps generated by three mesh saliency detection methods. We generate a synthetic saliency map by assembling salient regions from individual saliency maps. The experimental results show that the synthetic saliency map achieves better performance than the individual saliency maps when used in our metric, and the performance gain of the synthetic saliency map over the individual saliency maps will be greater if the similarity between the individual saliency maps is lower. Our work on the incorporation of mesh saliency into MVQ assessment in this paper will benefit the design of better perceptual mesh quality metrics. The proposed metric can be used to guide the algorithm design in other mesh processing operations, such as mesh smoothing, mesh simplification and mesh watermarking, in order to achieve the optimal algorithm performance with least visual degradations. One typical practical application of our metric is to evaluate the visual quality of the transmitted 3D models over the network at the receiver ends or client terminals efficiently. The visual quality data can be used as a feedback for the content and service providers to optimize the quality of user experience. One of our future projects involves the following works: to build a large database that consists of more geometric models, to investigate a more advanced feature representation that reflects the local distortions of a mesh better, and to explore the relationship between mesh saliency and mesh quality assessment in a theoretical way. It will also be interesting to integrate visual attention instead of mesh saliency into MVQ assessment when the eye-tracking data of mesh becomes available in the future.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 61373084, No. 61711530245) and the Key Project of Shanghai Science and Technology Commission (No.17511106802).

References

[1] G. Lavoué, E. D. Gelasca, F. Dupont, A. Baskurt, T. Ebrahimi, Perceptually driven 3D distance metrics with application to watermarking, in: SPIE Optics+ Photonics, International Society for Optics and Photonics, 2006, pp. 63120L–63120L.

[2] G. Lavoué, A multiscale metric for 3D mesh visual quality assessment, Computer Graphics Forum 30 (2011) 1427–1437.

[3] K. Wang, F. Torkhani, A. Montanvert, A fast roughness-based approach to the assessment of 3D mesh visual quality, Computers & Graphics 36 (2012) 808–818.

[4] L. Váša, J. Rus, Dihedral angle mesh error: a fast perception correlated distortion measure for fixed connectivity triangle meshes, Computer Graphics Forum 31 (2012) 1715–1724.

[5] F. Torkhani, K. Wang, J.-M. Chassery, A curvature-tensor-based perceptual quality metric for 3D triangular meshes, Machine Graphics & Vision 23 (2014) 1–25.

[6] L. Dong, Y. Fang, W. Lin, H. S. Seah, Perceptual quality assessment for 3D triangle mesh based on curvature, IEEE Transactions on Multimedia 17 (2015) 2174–2184.

[7] X. Liu, L. Liu, W. Song, Y. Liu, L. Ma, Shape context based mesh saliency detection and its applications: A survey, Computers & Graphics 57 (2016) 12–30.

[8] C. H. Lee, A. Varshney, D. W. Jacobs, Mesh saliency, in: ACM transactions on graphics (TOG), volume 24, ACM, 2005, pp. 659–666.

[9] R. Song, Y. Liu, R. R. Martin, P. L. Rosin, Mesh saliency via spectral processing, ACM Transactions on Graphics (TOG) 33 (2014) 6.

[10] M. Limper, A. Kuijper, D. W. Fellner, Mesh saliency analysis via local curvature entropy, in: Proceedings of the 37th Annual Conference of the European Association for Computer Graphics: Short Papers, Eurographics Association, 2016, pp. 13–16.

[11] A. Nouri, C. Charrier, O. Lézoray, Multi-scale mesh saliency with local adaptive patches for viewpoint selection, Signal Processing: Image Communication 38 (2015) 151–166.

[12] P. Tao, J. Cao, S. Li, X. Liu, L. Liu, Mesh saliency via ranking unsalient patches in a descriptor space, Computers & Graphics 46 (2015) 264–274.

[13] A. K. Moorthy, A. C. Bovik, Visual importance pooling for image quality assessment, IEEE journal of selected topics in signal processing 3 (2009) 193–201.

[14] H. Liu, I. Heynderickx, Visual attention in objective image quality assessment: Based on eye-tracking data, IEEE Transactions on Circuits and Systems for Video Technology 21 (2011) 971–982.

[15] M. C. Farias, W. Y. Akamine, On performance of image quality metrics enhanced with visual attention computational models, Electronics letters 48 (2012) 631–633.

[16] H. Liu, U. Engelke, J. Wang, P. Le Callet, I. Heynderickx, How does image content affect the added value of visual attention in objective image quality assessment?, IEEE Signal Processing Letters 20 (2013) 355–358.

[17] L. Zhang, Y. Shen, H. Li, Vsi: A visual saliency-induced index for perceptual image quality assessment, IEEE Transactions on Image Processing 23 (2014) 4270–4281.

[18] W. Zhang, A. Borji, Z. Wang, P. Le Callet, H. Liu, The application of visual saliency models in objective image quality assessment: A statistical evaluation, IEEE transactions on neural networks and learning systems 27 (2016) 1266–1278.

[19] Y. Kim, A. Varshney, D. W. Jacobs, F. Guimbretière, Mesh saliency and human eye fixations, ACM Transactions on Applied Perception (TAP) 7 (2010) 12.

[20] X. Chen, A. Saparov, B. Pang, T. Funkhouser, Schelling points on 3D surface meshes, ACM Transactions on Graphics (TOG) 31 (2012) 29.

[21] F. P. Tasse, J. Kosinka, N. A. Dodgson, Quantitative analysis of saliency models, in: SIGGRAPH ASIA 2016 Technical Briefs, ACM, 2016, p. 19.

[22] E. Shtrom, G. Leifman, A. Tal, Saliency detection in large point sets, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 3591–3598.

[23] F. Ponjou Tasse, J. Kosinka, N. Dodgson, Cluster-based point set saliency, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 163–171.

[24] G. Lavoué, M. Corsini, A comparison of perceptually-based metrics for objective evaluation of geometry processing, IEEE Transactions on Multimedia 12 (2010) 636–649.

[25] M. Corsini, M.-C. Larabi, G. Lavoué, O. Petřík, L. Váša, K. Wang, Perceptual metrics for static and dynamic triangle meshes, Computer Graphics Forum 32 (2013) 101–125.

[26] B. E. Rogowitz, H. E. Rushmeier, Are image quality metrics adequate to evaluate the quality of geometric objects?, in: Human Vision and Electronic Imaging, 2001, pp. 340–348.

[27] Z. Wang, A. C. Bovik, Modern image quality assessment, Synthesis Lectures on Image, Video, and Multimedia Processing 2 (2006) 1–156.

[28] G. Lavoué, R. Mantiuk, Quality assessment in computer graphics, in: Visual Signal Quality Assessment, Springer, 2015, pp. 243–286.

[29] G. Lavoué, M. C. Larabi, L. Váša, On the efficiency of image metrics for evaluating the visual quality of 3D models, IEEE transactions on visualization and computer graphics 22 (2016) 1987–1999.

[30] Z. Karni, C. Gotsman, Spectral compression of mesh geometry, in: Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co.,

2000, pp. 279–286.
[31] O. Sorkine, D. Cohen-Or, S. Toledo, High-pass quantization for mesh encoding., in: Symposium on Geometry Processing, volume 42, 2003.
[32] M. Corsini, E. D. Gelasca, T. Ebrahimi, M. Barni, Watermarked 3D mesh quality assessment, IEEE Transactions on Multimedia 9 (2007) 247–256.
[33] Z. Bian, S.-M. Hu, R. R. Martin, Evaluation for small visual difference between conforming meshes on strain field, Journal of Computer Science and Technology 24 (2009) 65–75.
[34] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE transactions on image processing 13 (2004) 600–612.
[35] A. Nouri, C. Charrier, O. Lézoray, Full-reference saliency-based 3D mesh quality assessment index, in: Image Processing (ICIP), 2016 IEEE International Conference on, IEEE, 2016, pp. 1007–1011.
[36] D. Cohen-Steiner, J.-M. Morvan, Restricted Delaunay triangulations and normal cycle, in: Proceedings of the nineteenth annual symposium on Computational geometry, ACM, 2003, pp. 312–321.
[37] G. Taubin, Estimating the tensor of curvature of a surface from a polyhedral approximation, in: Computer Vision, 1995. Proceedings., Fifth International Conference on, IEEE, 1995, pp. 902–907.
[38] M. Garland, P. S. Heckbert, Surface simplification using quadric error metrics, in: Proceedings of the 24th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co., 1997, pp. 209–216.
[39] N. Ouerhani, R. Von Wartburg, H. Hugli, R. Müri, Empirical validation of the saliency-based model of visual attention, ELCVIA: electronic letters on computer vision and image analysis 3 (2004) 13–24.
[40] C. Lang, G. Liu, J. Yu, S. Yan, Saliency detection by multitask sparsity pursuit, IEEE Transactions on Image Processing 21 (2012) 1327–1338.
[41] Z. Wang, A. C. Bovik, Reduced-and no-reference image quality assessment, IEEE Signal Processing Magazine 28 (2011) 29–40.
[42] Video Quality Experts Group, Final report from the video quality experts group on the validation of objective models of video quality assessment, Phase II (FR_TV2), `https://www.its.bldrdoc.gov/media/4150/vqegii_final_report.doc` (2003).
[43] J. Wu, W. Lin, G. Shi, A. Liu, Perceptual quality metric with internal generative mechanism, IEEE Transactions on Image Processing 22 (2013) 43–54.
[44] P. Engeldrum, Psychometric scaling, a toolkit for imaging systems development, Imcotek Press, Winchester, USA (2000) 1–200.
[45] P. Cignoni, C. Rocchini, R. Scopigno, Metro: Measuring error on simplified surfaces, Computer Graphics Forum 17 (1998) 167–174.
[46] G. Lavoué, Erratum of the results of mesh visual quality metrics, `http://liris.cnrs.fr/glavoue/travaux/Erratum.html` (2018).
[47] P. Le Callet, F. Autrusseau, Subjective quality assessment irccyn/ivc database, http://www.irccyn.ec-nantes.fr/ivcdb/ (2005).
[48] Y. Horita, K. Shibata, Z. P. Saddad, Subjective quality assessment toyama database, http://mict.eng.u-toyama.ac.jp/mict (2008).
[49] N. Ponomarenko, V. Lukin, A. Zelensky, K. Egiazarian, M. Carli, F. Battisti, Tid2008-a database for evaluation of full-reference visual quality assessment metrics, Advances of Modern Radioelectronics 10 (2009) 30–45.
[50] E. C. Larson, D. M. Chandler, Most apparent distortion: full-reference image quality assessment and the role of strategy, Journal of Electronic Imaging 19 (2010) 1–21.