

RESEARCH

Open Access



A perceptual quality metric for dynamic triangle meshes

Zeynep Cipiloglu Yildiz^{1*} and Tolga Capin²

Abstract

A measure for assessing the quality of a 3D mesh is necessary in order to determine whether an operation on the mesh, such as watermarking or compression, affects the perceived quality. The studies on this field are limited when compared to the studies for 2D. In this work, we aim a full-reference perceptual quality metric for animated meshes to predict the visibility of local distortions on the mesh surface. The proposed visual quality metric is independent of connectivity and material attributes. Thus, it is not associated to a specific application and can be used for evaluating the effect of an arbitrary mesh processing method. We use a bottom-up approach incorporating both the spatial and temporal sensitivity of the human visual system. In this approach, the mesh sequences go through a pipeline which models the contrast sensitivity and channel decomposition mechanisms of the HVS. As the output of the method, a 3D probability map representing the visibility of distortions is generated. We have validated our method by a formal user experiment and obtained a promising correlation between the user responses and the proposed metric. Finally, we provide a dataset consisting of subjective user evaluation of the quality of public animation datasets.

Keywords: Visual quality assessment, Animation, Geometry, VDP CSF

1 Introduction

Recent advances in 3D mesh modeling, representation, and rendering have matured to the point that they are now widely used in several mass-market applications, including networked 3D games, 3D virtual and immersive worlds, and 3D visualization applications. Using a high number of vertices and faces allows a more detailed representation of a mesh, increasing the visual quality. However, this causes a performance loss because of the increased computations. Therefore, a tradeoff often emerges between the visual quality of the graphical models and processing time, which results in a need to estimate the quality of 3D graphical content.

Several operations on 3D models rely on a good estimate of 3D mesh quality. For example, network based applications require *3D model compression and streaming*, in which a tradeoff must be made between the visual quality and the transmission speed. Several applications require *level-of-detail (LOD) simplification* of 3D meshes

for fast processing and rendering optimization. *Watermarking* of 3D meshes requires evaluation of quality due to artifacts produced. *Indexing and retrieval of 3D models* require metrics for judging the quality of 3D meshes that are indexed. Most of these operations cause certain modifications to the 3D shape. For example, compression and watermarking schemes may introduce aliasing or even more complex artifacts; LOD simplification and denoising result in a kind of smoothing of the input mesh and can also produce unwanted sharp features.

Quality assessment of 3D meshes is generally understood as the problem of evaluation of a modified mesh with respect to its original form based on detectability of changes. Quality metrics are given a reference mesh and its processed version, and compute geometric differences to reach a quality value. Furthermore, certain operations on the input 3D mesh, such as simplification, reduce the number of vertices; and this makes it necessary to handle topographical changes in the input mesh.

Contributions Most of the existing 3D quality metrics have focused on static meshes, and they do not target animated 3D meshes. Detection of distortions on

*Correspondence: zeynep.cipiloglu@cbu.edu.tr

¹Faculty of Engineering, Celal Bayar University, Muradiye/Manisa, Turkey
Full list of author information is available at the end of the article

animated meshes is particularly challenging since temporal aspects of seeing are complex and only partially modeled. We propose a method to estimate the 3D spatiotemporal response, by incorporating temporal as well as spatial human visual system (HVS) processes. For this purpose, our method follows a 3D object-space approach by extending the image-space sensitivity models for 2D imagery in 3D space. These models, based on vast amount of empirical research on retinal images, allow us to follow a more principled approach to model the perceptual response to 3D meshes. The result of our perceptual quality metric is the probability of distortion detection as a 3D map, acquired by taking the difference between estimated visual response 3D map of both meshes (Fig. 1). Subjective evaluation of the proposed method demonstrates favorable results for our quality estimation method. The supplementary section of this paper provides a dataset which includes subjective evaluation results of several animated meshes.

2 Related work

Methods for quality assessment of triangle meshes can be categorized according to their approach to the problem and the solution space. Non-perceptual methods approach the problem geometrically, without taking human perception effects into account. On the other hand, perceptual methods integrate human visual system properties into computation. Moreover, solutions can further be divided into image-based and model-based solutions. Model-based approaches work in 3D object space, and use structural or attribute information of the mesh. Image-based solutions, on the other hand, work in 2D image space, and use rendered images to estimate the quality of the given mesh. Several quality metrics have been proposed; [6], [12], and [28] present surveys on the recently proposed 3D quality metrics.

2.1 Geometry-distance-based metrics

Several methods use geometrical information to compute a quality value of a single mesh or a comparison between meshes. Therefore, methods that fall into this category do not reflect the perceived quality of the mesh.

Model-based metrics The most straightforward object space solution is the Euclidean distance or root mean squared (RMS) distance between two meshes. This method is limited to comparing two meshes with the same number of vertices and connectivity. To overcome this constraint, more flexible geometric metrics have been proposed. One of the most commonly used geometric measure is Hausdorff distance [9]. The Hausdorff distance defines the distance between two surfaces as the maximum of all pointwise distances. This definition is one-sided ($D(AB) \neq D(BA)$). Extensions to this approach have been proposed, such as taking the average, root mean squared error, or combinations [34].

Image-based metrics The simplest view dependent approach is the root-mean-squared error of two rendered images, by comparing them pixel by pixel. This metric is highly affected by luminance, shifts and scales, therefore is not a good approach [6]. Peak signal-to-noise ratio (PSNR) is also a popular quality metric for natural images where RMS of the image is scaled with the peak signal value. Wang et al. [49] show that alternative pure mathematical quality metrics do not perform better than PSNR although results indicate that PSNR gives poor results on pictures of artificial and human-made objects.

2.2 Perceptually based metrics

Perceptually aware quality metrics or modification methods integrate computational models or characteristics of the human visual system into the algorithm. Lin and Kuo [31] present a recent survey on perceptual visual quality metrics; however, as this survey indicates, most of the studies in this field focus on 2D image or video quality. A large number of factors affect the visual appearance of a scene, and several studies only focus on a subset of features of the given mesh.

Model-based perceptual metrics Curvature is a good indicator of structure and roughness which highly affect visual experience. A number of studies focus on the

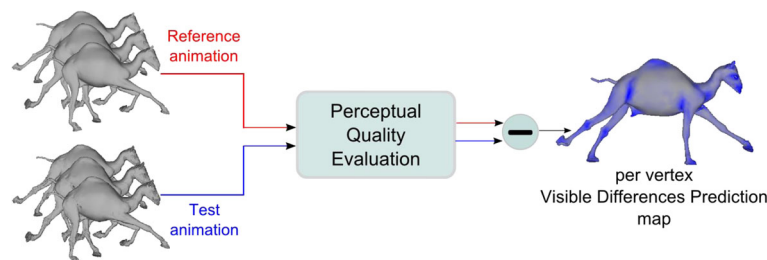


Fig. 1 Overview of the perceptual quality evaluation for dynamic triangle meshes

relation between curvature-linked characteristics and perceptual guide, and integrate curvature in quality assessment or modification algorithms. Karni and Gotsman [22] introduce a metric (GL1) by calculating roughness for mesh compression using Geometric Laplacian of every vertex. The Laplacian operator takes into account the geometry and topology. This simplification scheme uses variances in dihedral angles between triangles to reflect local roughness and weigh mean dihedral angles according to the variance. Sorkine et al. [41] modifies this metric by using slightly different parameters to obtain the metric called GL2.

Following the widely-used structural similarity concept in 2D image quality assessment, Lavouè [26] proposes a local mesh structural distortion measure called MSDM which uses curvature for structural information. MDSM2 [25] method improves this approach in several aspects: The new metric is multiscale and symmetric, the curvature calculations are slightly different to improve robustness, and there is no connectivity constraints.

Spatial frequency is linked to variance in 3D discrete curvature, and studies have used this curvature as a 3D perceptual measure [24], [29]. Roughness of a 3D mesh has also been used to measure quality of watermarked meshes [19], [11]. In [11], two objective metrics (3DWPM1 and 3DWPM2) derived from two definitions of surface roughness are proposed as the change in roughness between the reference and test meshes. Pan et al. [37] use the vertex attributes in their proposed quality metric.

Another metric developed for 3D mesh quality assessment is called FMPD which is based on local roughness estimated from Gaussian curvature [48]. Torkhani and colleagues [44] propose another metric (TPDM) based on curvature tensor difference of the meshes to be compared. Both of these metrics are independent of connectivity and designed for static meshes. Dong et al. [16] propose a novel roughness-based perceptual quality assessment method. The novelty of the metric lies in the incorporation of structural similarity, visual masking, and saturation effect which are highly employed in quality assessment methods separately. This metric is also similar to ours in the sense that it uses a HVS pipeline but it is designed for static meshes with connectivity constraints. Besides, they capture structural similarity which is not handled in our method.

Alternatively, Nader et al. [36] propose a just noticeable distortion (JND) profile for flat-shaded 3D surfaces in order to quantify the threshold for the change in vertex position to be detected by a human observer, by defining perceptual measures for local contrast and spatial frequency in 3D domain. Guo et al. [20] evaluate the local visibility of geometric artifacts on static meshes by means of a series of user experiments. In these experiments,

users paint the local distortions on the meshes and the prediction accuracies of several geometric attributes (curvatures, saliency, dihedral angle, etc.) and quality metrics such as Hausdorff distance, MSDM2, and FMPD are calculated. According to the results, curvature-based features outperform the others. They also provide a local distortion dataset as a benchmark.

A perceptually based metric for evaluating dynamic triangle meshes is the STED error [46]. The metric is based on the idea that perception of distortion is related to local and relative changes rather than global and absolute changes [12]. The spatial part of the error metric is obtained by computing the standard deviation of relative edge lengths within a topological neighborhood of each vertex. Similarly, the temporal error is computed by creating virtual temporal edges connecting a vertex to its position in the subsequent frame. The hypotenuse of the spatial and temporal components then gives the STED error. Another attempt for perceptual quality evaluation of dynamic meshes is by Torkhani et al. [45]. Their metric is a weighted mean square combination of three distances: speed-weighted spatial distortion measure, vertex speed-related contrast, and vertex moving direction related contrast. Experimental studies show that the metric performs quite well; however, it requires fixed connectivity meshes. They also provide a publicly available dataset and a comparative study to benchmark existing image and model based metrics.

Image-based perceptual metrics Human visual system characteristics are also used in image-space solutions. These metrics generally use the contrast sensitivity function (CSF), an empirically driven function that maps human sensitivity to spatial frequency. Daly's widely used visible difference predictor [14] gives the perceptual difference between two images. Longhurst and Chalmers [32] study VDP to show favorable image-based results with rendered 3D scenes. Lubin proposes a similar approach with Sarnoff Visual Discrimination Model (VDM) [33], which operates in spatial domain, as opposed to VDP's approach in frequency domain. Li et al. [30] compare VDP and Sarnoff VDM with their own implementation of the algorithms. Analysis of the two algorithms shows that the VDP takes place in feature space and takes advantage of FFT algorithms, but a lack of evidence of these feature space transformations in the HVS gives VDM an advantage.

Bolin et al. [5] incorporate color properties in 3D global illumination computations. Studies show that this approach gives accurate results [50]. Minimum detectable difference is studied as a perceptual metric [39] that handles luminance and spatial processing independently. Another approach for computer generated images is visual equivalence detector [38]. Visual impressions of scene

appearance are analyzed and the method outputs a visual equivalence map.

Visual masking is taken into account in 3D graphical scenes with varying texture, orientation and luminance values [18]. Several approaches with color emphasis is introduced by Albin et al. [1], which predict differences in LLAB color space. Dong et al. [15] exploit entropy masking, which accounts for the lower sensitivity of the HVS to distortions in unstructured signals, for guiding adaptive rendering of 3D scenes to accelerate rendering.

An important question that arises is whether model-based metrics are superior over image-based solutions. Although there are several studies on this issue, it is not possible to clearly state that one group of metrics is superior to the other. Rogowitz et al. conclude that image quality metrics are not adequate for measuring the quality of 3D meshes since lighting and animation affect the results significantly [40]. On the other hand, Cleju and Saube claim that image-based metrics predict perceptual quality better than metrics working on 3D geometry, and discuss ways to improve the geometric distances [10]. A recent study [27] investigates the best set of parameters for the image-based metrics when evaluating the quality of 3D models and compares them to several model-based methods. The implications from this study show that image-based metrics perform well for simple use cases such as determining the best parameters of a compression algorithm or in the cases when model-based metrics are not applicable.

The distinction of our work from the current metrics can be listed as follows: Firstly, our metric can handle dynamic meshes in addition to the static meshes. Secondly, we produce a per-vertex error map instead of a global quality value per-mesh, which allows to guide perceptual geometry processing applications. Furthermore, our method can handle meshes with different connectivity. Lastly, the proposed metric is not application specific.

3 Background

In this section, we summarize and discuss several mechanisms of the human visual system that construct our model.

3.1 Luminance adaptation

The luminance that falls on the retina may vary in significant amount from a sunny day to moonless night. The photoreceptor response to luminance forms a nonlinear S-shaped curve, which is centered at the current adaptation luminance and exhibits a compressive behavior while moving away from the center [2].

Daly [14] has developed a simplified local amplitude nonlinearity model in which the adaptation level of a pixel

is merely determined from that pixel. Equation 1 provides this model.

$$\frac{R(i,j)}{R_{max}} = \frac{L(i,j)}{L(i,j) + c_1 L(i,j)^b} \quad (1)$$

where $R(i,j)/R_{max}$ is the normalized retinal response, $L(i,j)$ is the luminance of the current pixel, and c_1 and b are constants.

3.2 Channel decomposition

The receptive fields in the primary visual cortex are selective to certain spatial frequencies and orientations [2]. There are several alternatives to account for modeling the visual selectivity of the HVS such as Laplacian Pyramid, Discrete Cosine Transform (DCT), and Cortex Transform. Most of the studies in the literature tend to choose Cortex Transform [14] among these alternatives, since it offers a balanced solution for the tradeoff between physiological plausibility and practicality [2].

2D Cortex Transform combines both frequency selectivity and orientation selectivity of the HVS. Frequency selectivity component is modeled by the band-pass filters given in Eq. 2.

$$dom_k = \begin{cases} mesa_{k-1} - mesa_k & \text{for } k = 1 \dots K - 2 \\ mesa_{k-1} - baseband & \text{for } k = K - 1 \end{cases} \quad (2)$$

where K is the total number of spatial bands [2]. Low-pass filters $mesa_k$ and baseband are calculated using Eq. 3.

$$mesa_k = \begin{cases} 1 & , \rho \leq r - \frac{tw}{2} \\ \frac{1}{2} \left(1 + \cos \left(\frac{\pi(\rho - r + \frac{tw}{2})}{tw} \right) \right) & , r - \frac{tw}{2} < \rho \leq r + \frac{tw}{2} \\ e^{-\frac{\rho^2}{2\sigma^2}} & , \rho < r_{K-1} + \frac{tw}{2} \\ 0 & , \text{otherwise} \end{cases} \quad (3)$$

where $r = 2^{-k}$, $\sigma = \frac{1}{3}(r_{K-1} + \frac{tw}{2})$ and $tw = \frac{2}{3}r$. For the orientation selectivity, *fan filters* are used (Eq. 4 and 5).

$$fan_l = \begin{cases} \frac{1}{2} \left(1 + \cos \left(\frac{\pi|\theta - \theta_c(l)|}{\theta_{tw}} \right) \right) & \text{for } |\theta - \theta_c(l)| \leq \theta_{tw} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$\theta_c(l) = (l - 1) \cdot \theta_{tw} - 90 \quad (5)$$

where $\theta_c(l)$ is the orientation of the center and $\theta_{tw} = 180/L$ is the transitional width. Then, the cortex filter (Eq. 6) is obtained by multiplying the *dom* and *fan* filters.

$$B^{k,l} = \begin{cases} dom_k \cdot fan_l & \text{for } k = 1 \dots K - 1 \text{ and } l = 1 \dots L \\ baseband & \text{for } k = K \end{cases} \quad (6)$$

3.3 Contrast sensitivity

Spatial contrast sensitivity The contrast sensitivity function (CSF) measures the sensitivity to luminance gratings as a function of spatial frequency, where sensitivity is defined as the inverse of the threshold contrast. Mostly used spatial CSF models are Daly [14] and Barten's [3] models. Figure 2a shows Blakemore et al.'s experimental results without adaptation effects [4].

Temporal contrast sensitivity Intensity change across time constructs the temporal features of an image. In a user study conducted by Kelly [23], the sensitivity with respect to *temporal frequency* is estimated by displaying a simple shape with alternating luminance as a stimuli. The results of the experiment are used to plot the temporal CSF shown in Fig. 2b.

Another issue to consider is the eye's tracking ability, known as *smooth pursuit*, which compensates for the loss of sensitivity due to motion by reducing the retinal speed of the object of interest to a certain degree. Daly [13] draws a heuristic for smooth pursuit according to the experimental measurements.

It is also important to note the distinction between the spatiotemporal and spatiovelocity CSF [13]. Spatiotemporal CSF (Fig. 3a) takes spatial and temporal frequencies as input, while spatiovelocity CSF (Fig. 3b) takes directly the retinal velocity instead of the temporal frequency. Spatiovelocity CSF is more suitable for our application since it is more straightforward to estimate the retinal velocity than temporal frequency and it allows the integration of the smooth pursuit effect.

4 Approach

Our work shares some features of the VDP method [14] and recent related work. These methods have

shown the ability to estimate the perceptual quality of static images [14] and 2D video sequences for animated walkthroughs [35].

Figure 4 shows the overview of the method. Our method has a full reference approach in which a reference and a test mesh sequence are provided to the system. Both the reference and test sequences undergo the same perceptual quality evaluation process and the difference of these outputs is used to generate a per-vertex probability map for the animated mesh. The probability value at a vertex estimates the visible difference of the distortions in the test animation, when compared to the reference animation. In our method, we construct a 4D space-time (3D+time) volume and extend several HVS correlated processes used for 2D images, to operate on this volume. Below, the steps of the algorithm are explained in detail.

4.1 Preprocessing

Calculation of the illumination, construction of the spatiotemporal volume, and estimation of vertex velocities are performed in the preprocessing step.

Illumination calculation First we calculate the vertex colors assuming a Lambertian surface with diffuse and ambient components (Eq. 7).

$$\mathbf{I} = k_a \mathbf{I}_a + k_d \mathbf{I}_d (\mathbf{N} \cdot \mathbf{L}) \quad (7)$$

where \mathbf{I}_a is the intensity of the ambient light, \mathbf{I}_d is the intensity of the diffuse light, \mathbf{N} is the vertex normal, \mathbf{L} is the direction to the light source, and k_a and k_d are ambient and diffuse reflection coefficients, respectively.

In this study, we aim a general-purpose quality evaluation that is independent of shading and material properties. Therefore, information about the material properties, light sources, etc. are not available. A directional light

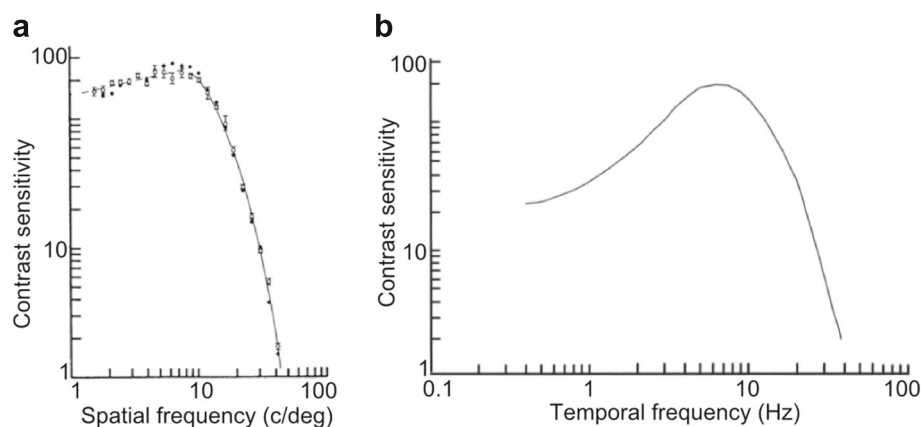


Fig. 2 Contrast sensitivity functions. **a** Spatial CSF (Image from [4], cc 1969 John Wiley and Sons, reprinted with permission.) **b** Temporal CSF (Constructed using Kelly's [23] temporal adaptation data)

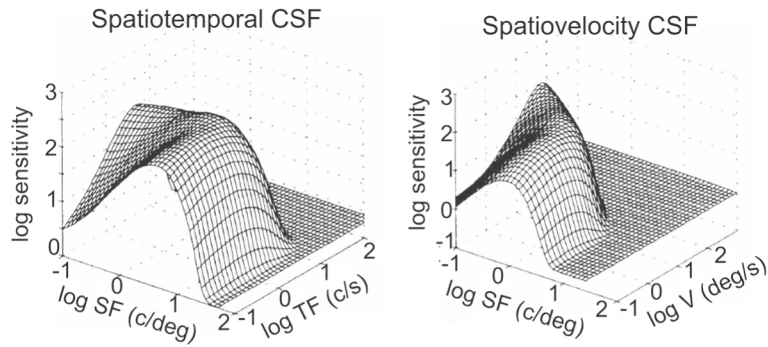


Fig. 3 Spatiotemporal vs. spatiovelocity CSF (Images from [13], cc 1998 SPIE, reprinted with permission.)

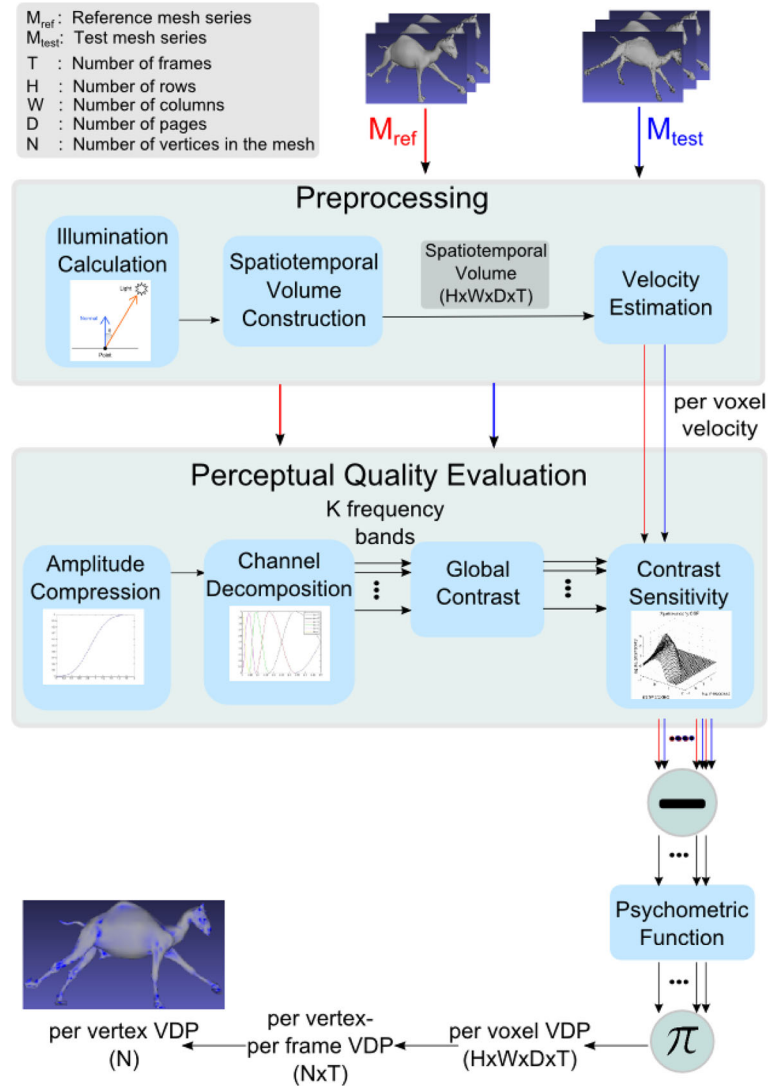


Fig. 4 Method overview

source from left-above of the scene is assumed in accordance with the human visual system's assumptions ([21], section 24.4.2).

The lighting model with the aforementioned assumptions can be generalized to incorporate multiple light sources, specular reflections, etc. using Eq. 8; if light sources and material properties are available.

$$\mathbf{I} = k_a \mathbf{I}_a + \sum_{i=1}^n \left[k_d \mathbf{I}_d^i (\mathbf{N} \cdot \mathbf{L}^i) + k_s \mathbf{I}_d^i (\mathbf{N} \cdot \mathbf{H}^i)^p \right] \quad (8)$$

where n is the number of light sources, k_s is the specular reflection coefficient, and \mathbf{H} is the halfway vector.

Construction of the spatiotemporal volume We convert the object-space mesh sequences into an intermediate volumetric representation, to be able to apply image-space operations. We construct a 3D volume for each frame, where we store the luminance values of the vertices at each voxel. The values of the empty voxels are determined by linear interpolation.

Using such a spatiotemporal volume representation provides an important flexibility as we get rid of the connectivity problems and it allows us to compare meshes with different number of vertices. Moreover, the input model is not restricted to be a triangle mesh; volumetric representation enables the algorithm to be applied on other representations such as point-based graphics. Another advantage is that the complexity of the algorithm is not much affected by the number of vertices.

To obtain the spatiotemporal volume, we first calculate the axis aligned bounding box (AABB) of the mesh. To prevent inter-frame voxel correspondence problems, we use the overall AABB of the mesh sequences. We use the same voxel resolution for both test and reference mesh sequences. Determining the suitable resolution for the voxels is critical since it highly affects the accuracy of the results and the time and memory complexity of the algorithm. At this point, we use a heuristic (Eq. 9) to calculate the resolution at each dimension, in proportion to the length of the bounding box in the corresponding dimension. We analyze the effect of the *minResolution* parameter in this equation on the performance, in Section 5.3.1.

$$\begin{aligned} \text{minLength} &= \min(\text{width}_{BB}, \text{height}_{BB}, \text{depth}_{BB}) \\ w &= \lfloor \text{width}_{BB} / \text{minLength} \rfloor \\ h &= \lfloor \text{height}_{BB} / \text{minLength} \rfloor \\ d &= \lfloor \text{depth}_{BB} / \text{minLength} \rfloor \\ W &= w * \text{minResolution} \\ H &= h * \text{minResolution} \\ D &= d * \text{minResolution} \end{aligned} \quad (9)$$

At the end of this step, we obtain a 3D spatial volume for each frame, which in turn constructs a 4D (3D+time) representation for both reference and test mesh sequences. We call this structure *spatiotemporal volume*. Also, an index structure is maintained to keep the voxel indices of each vertex. The rest of the method operates on this 4D spatiotemporal volume.

In the following steps, we do not use the full spatiotemporal volume for performance related concerns. We define a time window as suggested by Myszkowski et al. [35, p. 362]. According to this heuristic, we only consider a limited number of consecutive frames to compute the visible difference prediction map of a specific frame. In other words, to calculate the probability map for the i^{th} frame, we process the frames between $i - \lfloor tw/2 \rfloor$ and $i + \lfloor tw/2 \rfloor$, where tw is the length of the time window. We empirically set it as $tw = 3$.

Velocity estimation Since our method also has a time dimension, we need the vertex velocities in each frame. Using an index structure, we compute the voxel displacement of each vertex (D_i) between consecutive frames ($\Delta D_i = \|p_{it} - p_{i(t-1)}\|$ where p_{it} denotes the voxel position of vertex i at frame t). The remaining empty voxels inside the bounding box are assumed to be static.

Then, we calculate the velocity of each voxel at each frame (v in *deg/sec*), using the pixel resolution (*ppd* in *pixels/deg*) and frame rate (*FPS* in *frames/sec*) with Eq. 10. We assume default viewing parameters of 0.5 *m* viewing distance and 19-*inch* display with 1600X900 resolution, while calculating *ppd* in Eq. 10. This is then adapted with N_1 frames to reduce the erroneous computations (Eq. 11).

$$v_{it} = \frac{\Delta D_i}{ppd} * FPS \quad (10)$$

$$v'_{it} = \frac{v_{i(t-1)} + v_{it} + v_{i(t+1)}}{3} \quad (11)$$

Lastly, it is crucial to compensate for smooth pursuit eye movements to be used in spatiotemporal sensitivity calculations. This will allow us to handle temporal masking effect where high-speed motion hides the visibility of distortions. The following equation (Eq. 12) describes a motion compensation heuristic proposed by Daly [13].

$$v_R = v_I - \min(0.82v_I + v_{min}, v_{max}) \quad (12)$$

where v_R is the compensated velocity, v_I is the physical velocity, v_{min} is the drift velocity of the eye (0.15 *deg/sec*), v_{max} is the maximum velocity that the eye can track efficiently (80 *deg/sec*). According to Daly [13], the eye tracks

all objects in the visual field with an efficiency of 82%. We adopt the same efficiency value for our spatiotemporal volume. However, if the visual attention map is available, it is also possible to substitute this map as the tracking efficiency [51].

4.2 Perceptual quality evaluation

In this section, the main steps of the perceptual quality evaluation system are explained in detail.

Amplitude compression Daly [14] proposes a simplified local amplitude nonlinearity model as a function of pixel location, which assumes perfect local adaptation (Section 3.1). We have adapted this nonlinearity to our spatiotemporal volume representation (Eq. 13).

$$\frac{R(x, y, z, t)}{R_{max}} = \frac{L(x, y, z, t)}{L(x, y, z, t) + c_1 L(x, y, z, t)^b} \quad (13)$$

where x, y, z , and t are voxel indices, $R(x, y, z, t)/R_{max}$ is the normalized response, $L(x, y, z, t)$ is the value of the voxel, $b = 0.63$ and $c_1 = 12.6$ are constants. In this step, voxel values are compressed by this amplitude nonlinearity.

Channel decomposition We adapt the cortex transform [14] which is described in Section 3.2, on our spatiotemporal volume with a small exception. A 3D model is not assumed to have a specific orientation at a given time, in our method. For this purpose, we exclude fan filters that are used for orientation selectivity from the cortex transform adaptation. Therefore, in our cortex filter implementation, we use Eq. 14 instead of Eq. 6 with only *dom* filters (Eq. 2). These band-pass filters are portrayed in Fig. 5.

$$B^k = \begin{cases} \text{dom}_k & \text{for } k = 1..K - 1 \\ \text{baseband} & \text{for } k = K \end{cases} \quad (14)$$

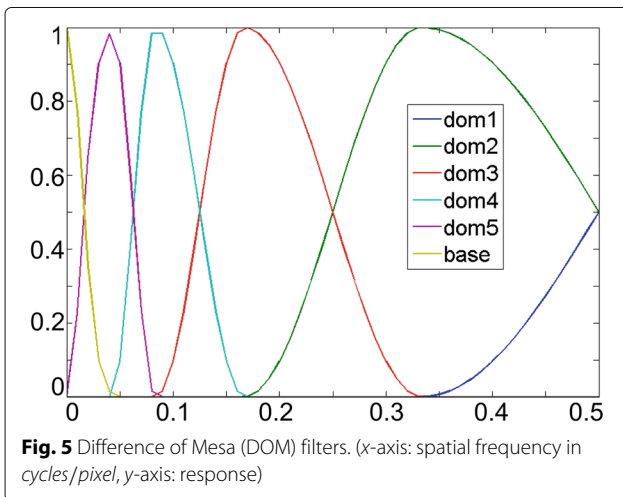


Fig. 5 Difference of Mesa (DOM) filters. (x-axis: spatial frequency in cycles/pixel, y-axis: response)

We perform cortex filtering in the frequency domain by applying Fast Fourier Transform (FFT) on the spatiotemporal volume and multiplying this with the cortex filters that are constructed in the frequency domain. We obtain K frequency bands at the end of this step. Each frequency band is then transformed back to the spatial domain. This process is illustrated in Fig. 6.

Global contrast The sensitivity to a pattern is determined by its contrast rather than its intensity [17]. Contrast in every frequency channel is computed according to the global contrast definition with respect to the mean value of the whole channel, given in Eq. 15 [35], [17].

$$C^k = \frac{I^k - \text{mean}(I^k)}{\text{mean}(I^k)} \quad (15)$$

where C^k is the spatiotemporal volume of contrast values and I^k is the spatiotemporal volume of luminance values in frequency channel k .

Contrast sensitivity Filtering the input image with the contrast sensitivity function (CSF) constructs the core part of the VDP-based models (Section 3.3). Since our model is for dynamic meshes, we use the spatiovelocity CSF (Fig. 3b) which describes the variations in visual sensitivity as a function of both spatial frequency and velocity, instead of the static CSF used in the original VDP.

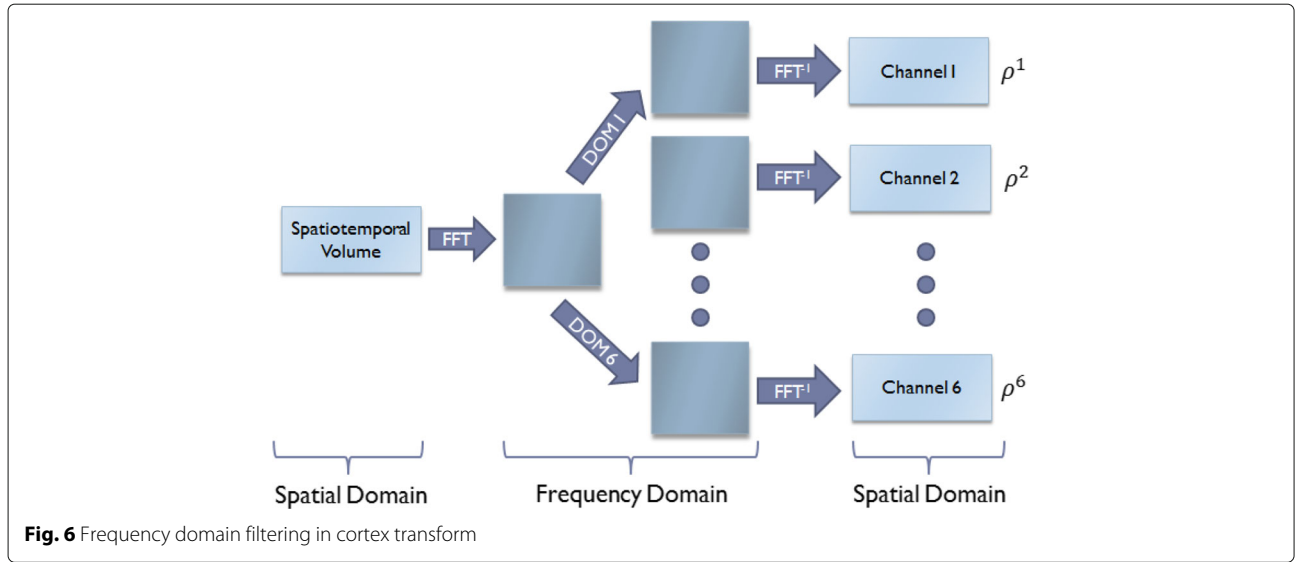
Our method handles temporal distortions in two ways. First, smooth pursuit compensation handles temporal masking effect which refers to the loss of sensitivity due to high speed. Secondly, we use spatiovelocity CSF in which contrast sensitivity is measured according to the velocity, instead of static CSF.

Each frequency band is weighted with the spatiovelocity CSF which is given in Eq. 16 [13], [23]. One input to the CSF is per voxel velocities in each frame, estimated in preprocessing; and the other input is the center spatial frequency of each frequency band.

$$CSF(\rho, v) = c_0 \left(6.1 + 7.3 \left| \log \left(\frac{c_2 v}{3} \right) \right|^3 \right) * c_2 v (2\pi c_1 \rho)^2 * \exp \left(-\frac{4\pi c_1 \rho (c_2 v + 2)}{45.9} \right) \quad (16)$$

where ρ is the spatial frequency in *cycles/degree*, v is the velocity in *degrees/second*, and $c_0 = 1.14$, $c_1 = 0.67$, $c_2 = 1.7$ are empirically set coefficients. A more principled way would be to obtain these parameters through a parameter learning method.

Error pooling All the previous steps are applied on the reference and test animations. At the end of these steps, we obtain K channels for each mesh sequence. We take the difference of test and reference pairs for each channel and the outputs go through a psychometric function that maps the perceived contrast (C') to detection probability



using Eq. 17 [2]. After applying the psychometric function, we combine each band using the probability summation formula (Eq. 18) [2].

$$P(C') = 1 - \exp(-|C'|^3) \quad (17)$$

$$\hat{P} = 1 - \prod_{k=1}^K (1 - P^k) \quad (18)$$

The resulting \hat{P} is a 4D volume that contains the detection probabilities per voxel. It is then straightforward to convert this 4D volume to per vertex probability map for each frame, using the index structure (Section 4.1). Lastly, to combine the probability maps of each frame into a single map, we take the average of all frames per vertex. This gives us a per vertex visible difference prediction map for the animated mesh.

Summary of the method The overall process is summarized in Eq.19 in which \mathcal{F} denotes the Fourier Transform, \mathcal{F}^{-1} denotes the inverse Fourier Transform, and L_T and L_R are spatiotemporal volumes for test and reference mesh sequences, respectively. ρ^k is the center spatial frequency of channel k and V_T and V_R contain the voxel velocities for L_T and L_R , respectively.

$$\begin{aligned} C_{TR}^k &= \text{Contrast} \left(\text{Channel}_{TR}^k \right) * \text{CSF} \left(\rho^k, V_{TR} \right) \\ \text{Channel}_{TR}^k &= \mathcal{F}^{-1} \left[\mathcal{F} \left(A_{CTR} \right) * \text{DOM}^k \right] \\ A_{CTR} &= \text{AmplitudeCompression} \left(L_{TR} \right) \\ P^k &= P \left(C_T^k - C_R^k \right) \\ P &= 1 - \prod_{k=1}^K (1 - P^k) \end{aligned} \quad (19)$$

5 Validation of the metric

In this section, we provide a two-fold validation of our metric: through a psychophysical user study designed for dynamic meshes and comparison to several standard objective metrics. We also give measurements on the computational time of the proposed method.

5.1 User evaluation

We conducted subjective user experiments to evaluate the fidelity of our quality metric. In this section, we explain the experimental design and analyze the results. The subjective evaluation results in this study are publicly available as supplementary material.

5.1.1 Data

We used four different mesh sequences in the experiments. The original versions of these animated meshes (Fig. 7) are obtained from public datasets [42] and [47]; and information about these meshes are given in Table 1. The animations are continuously repeated and the playback frame rate is 60 frames/second for the sequences. For the modified versions of the animated meshes, we apply random vertex displacement filter on each frame of the reference meshes, using MeshLab tool [8]. The only parameter of this filter is the maximum displacement which we set as 0.1. The vertices are randomly displaced with a vector whose normal is bounded by this value. This corresponds to adding random noise on the mesh vertices.

5.1.2 Experimental design

In this experiment, our aim is to measure the correlation between the subjective evaluation and the proposed metric results. The subjects in the experiment evaluated the perceived quality of the animated meshes

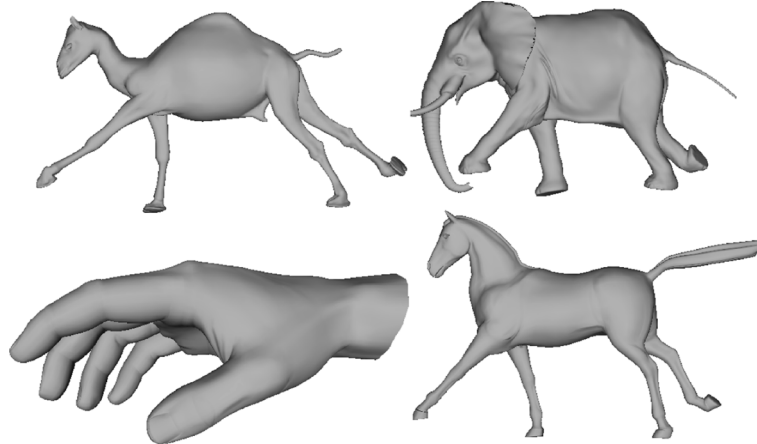


Fig. 7 Sample frames from the reference animations

by marking the perceived distortions on the mesh. For the experiment setup, we used simultaneous double stimulus for continuous evaluation (SDSCE) methodology among the standards listed in [6]. According to this design, presenting both stimuli simultaneously eliminates the need for memorization.

Task In the experiments, we used two displays; one for viewing the animations and the other for evaluation. In the viewing screen (Fig. 8a), both the reference and test meshes were shown in animation and the interaction (rotating and zooming) was simultaneous.

In the evaluation screen (Fig. 8b), a marking tool with tip intensity was supplied to the user. The user's task was to mark the visible distortions. The task of annotation would be very difficult if it was performed on dynamic state. Therefore, the users marked the visible distortions on a single static frame, selected manually (frames in Fig. 7). One may argue that marking the distortions on static state may introduce bias. We try to minimize this effect in two ways. First of all, the annotation was done on a sample frame of the reference animation instead of the modified animation. In this way, the distortions were never seen statically by the observers. Secondly, the user was still able to view both of the animations and manipulate the view-point simultaneously in the viewing screen, during the evaluation. This eliminates the necessity for memorization.

Table 1 Information about the meshes

	Camel	Elephant	Hand	Horse
# vertices	21,885	42,321	7997	8431
# frames	42	48	45	48

At the beginning of the experiments, subjects were given the following instruction: "A distortion on the mesh is defined as the spatial artifacts, compared to the reference mesh. Consider the relative scale of distortions and mark the visible distortions accordingly, using the intensity tool."

Setup The environment setup in the experiments has a significant impact on the results. Therefore, the parameters such as lighting, materials, and stimuli order should be carefully designed [6]. We explain each parameter below.

- *Viewing Parameters:* The observers viewed the stimuli on a 19-inch display from 0.5 m away the display.
- *Lighting:* We use a stationary left-above, center directed lighting [40].
- *Materials and Shading:* To prevent highlighting effects and accentuate distortions unpredictably, we used Gouraud shading in the experiments. Moreover, we used meshes without texture.
- *Animation and Interaction:* Free-viewpoint was enabled to the viewers for interaction. Furthermore, since inspection of the mesh during paused state was contradictory to the purpose of the experiment, two different displays were used and the evaluation of the mesh was conducted on one of the screens while the animation is ongoing on the other screen.
- *Stimuli order:* Each modified and reference mesh combination was presented in a random order allowing for more accurate comparisons. In other words, there was not a specific ordering of the meshes and subjects were also able to pause their evaluation and continue whenever they want.

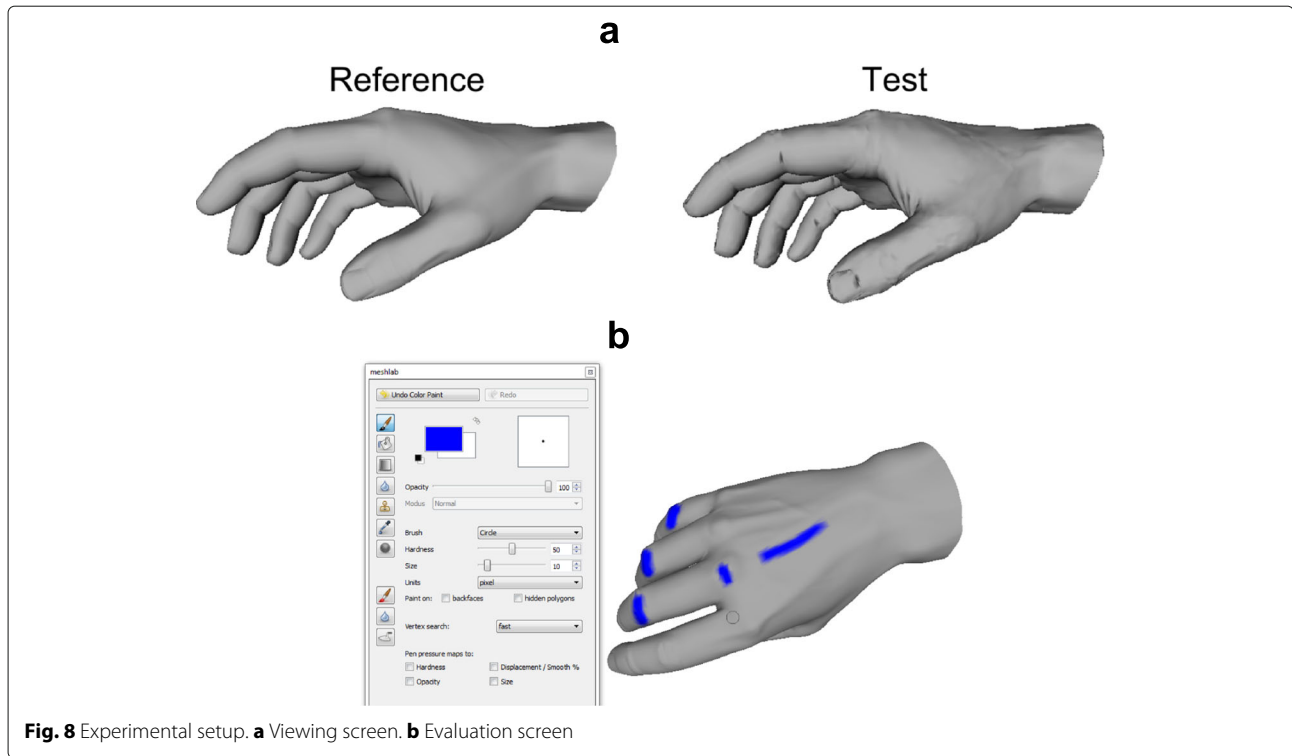


Fig. 8 Experimental setup. **a** Viewing screen. **b** Evaluation screen

Subjects Twelve subjects with various levels of computer experience participated in the experiment. All of the subjects evaluated every animated mesh in the experiment.

5.1.3 Results and discussion

The mesh frames that were marked by the subjects were stored as vertex color maps. To unify the responses of each subject for each mesh, we calculate a mean subjective response using Eq. 20.

$$\mu(v_{i,M'}) = \frac{\sum_{s=1}^N R_s(v_{i,M'})}{N} \quad (20)$$

where N is the number of subjects who evaluated the mesh M' , $R_s(v_{i,M'})$ represents the given response to a single vertex v_i , mesh M' and subject s combination. Figure 9a, b shows sample results from the experiment along with the reference and modified mesh pair and the output of our algorithm.

Next, we compare the mean subjective responses with our proposed method's predictions. For this purpose, we use two common methods for correlation: Pearson linear correlation coefficient (r) for prediction accuracy, and Spearman rank order correlation coefficient (ρ) for monotonicity between the mean subjective response and estimated response [31].

Notice that correlation coefficients vary in the range of $[-1,1]$ and a negative coefficient indicates a negative

correlation while positive coefficient means a positive correlation. While interpreting the correlation analysis, we used the categorization in [43], where correlation coefficients (in absolute value) which are ≤ 0.35 are considered as low or weak correlations, $0.36 \leq r, \rho \leq 0.67$ modest or moderate correlations, and $0.68 \leq r, \rho \leq 1$ strong or high correlations.

While measuring the correlation, we considered the limitations of the paint tool, in which subjects may unintentionally mark some region nearby the region they actually target. To reduce the effect of this problem, we followed the approach used in image/video quality assessment validations where image or video frame is divided into a regular grid and the comparison is done tile by tile [2]. Based on this idea, we grouped the nearby vertices and find the correlation based on the average intensity of these regions. We asked a designer to segment the mesh manually using a paint-based interface, although any available mesh segmentation technique could also be used for this purpose [7]. The designer was instructed to create about 50 segments for each model.

Table 2 includes the correlation coefficients for each mesh and when all the samples are combined (overall). Both Pearson and Spearman correlation analysis give consistent results. However, Spearman's correlation could be more reliable in our case, because a darker mark in user responses indicates a higher distortion; yet, it is a subjective issue to decide on which intensity

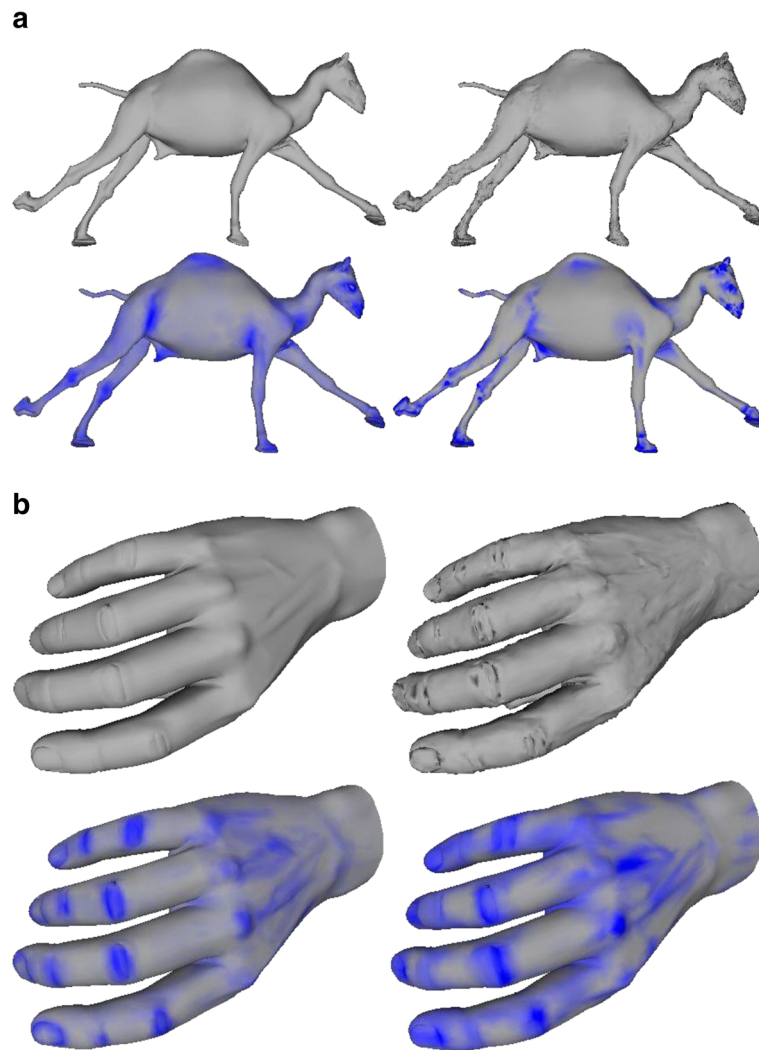


Fig. 9 **a** Camel mesh. **b** Hand mesh. *Top-left*: reference mesh, *top-right*: modified mesh, *bottom-left*: mean subjective response, *bottom-right*: estimated visual response. *Blue* regions in the mean subjective response and estimated response maps demonstrate the high perceptual differences

corresponds to which distortion amount. Hence, finding a correlation between the rank orders of the vertices rather than the absolute color values is more appropriate.

As the table indicates, the average correlation is about 70%, which can be considered as a promising result for

Table 2 Pearson (r) and Spearman (ρ) correlation coefficients for each mesh

	Pearson $_r$	Spearman $_\rho$	Strength
Camel	0.835	0.829	High
Elephant	0.585	0.654	Modest
Hand	0.715	0.707	High
Horse	0.713	0.700	High
Overall	0.712	0.723	High

the field of local dynamic mesh quality assessment. Correlation coefficients for *Camel*, *Hand*, and *Horse* meshes are high, while *Elephant* mesh exhibits a moderate correlation.

One important issue that affects the results negatively is that the subjects tend to evaluate only certain views of the meshes. Eight of the subjects reported that they had generally marked the meshes from the side views. In addition, since the meshes are known objects, visual attention principles may have come into play and our metric does not reflect this mechanism.

5.2 Comparison to STAR techniques

It is required to compare the performance of our method with the current state-of-the-art techniques. We first compared our metric to the static metrics using the public LIRIS/EPFL general purpose dataset [26].

In this dataset, there are 88 models, between 40 K and 50 K vertices, which were generated from four reference objects: *Armadillo*, *Venus*, *Dinosaur*, and *RockerArm*. Two types of distortion, noise addition and smoothing, were applied with different strengths at four locations: on the whole model, on smooth areas, on rough areas, and on intermediate areas. The dataset also includes mean opinion scores (MOS) from 12 observers and 7 static metric results for these models.

Since our method is also applicable for static meshes, we ran our algorithm on these models by setting velocities to 0. Although our aim is to produce a 3D map as output, to be able to compare our metric to the other techniques, we used the average of the vertex probabilities in the output map as the overall score of the mesh quality. These scores are in the range of 0–1 and a high score indicates that the distortions on this mesh are highly visible.

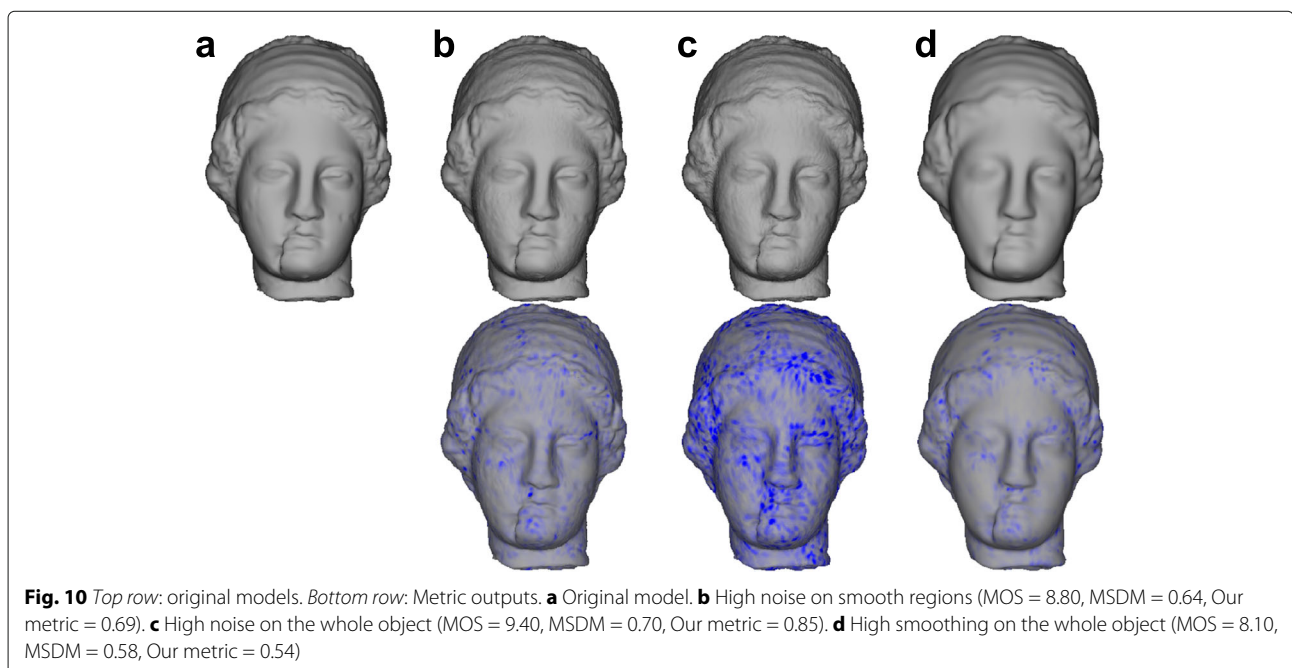
Figure 10 includes several examples from the *Venus* model. MOS values of the highly noisy objects in (b) and (c) are higher than the smoothed object in (d). This is intuitive as the smooth model seems less distorted than the noisy object. Our metric conforms to this situation since the metric outputs for (b) and (c) are higher than the output for (d). According to the subjective evaluations, model in (c) exhibits the highest distortion as our model also reflects. Our results show similarity between the results of the MSDM metric as well.

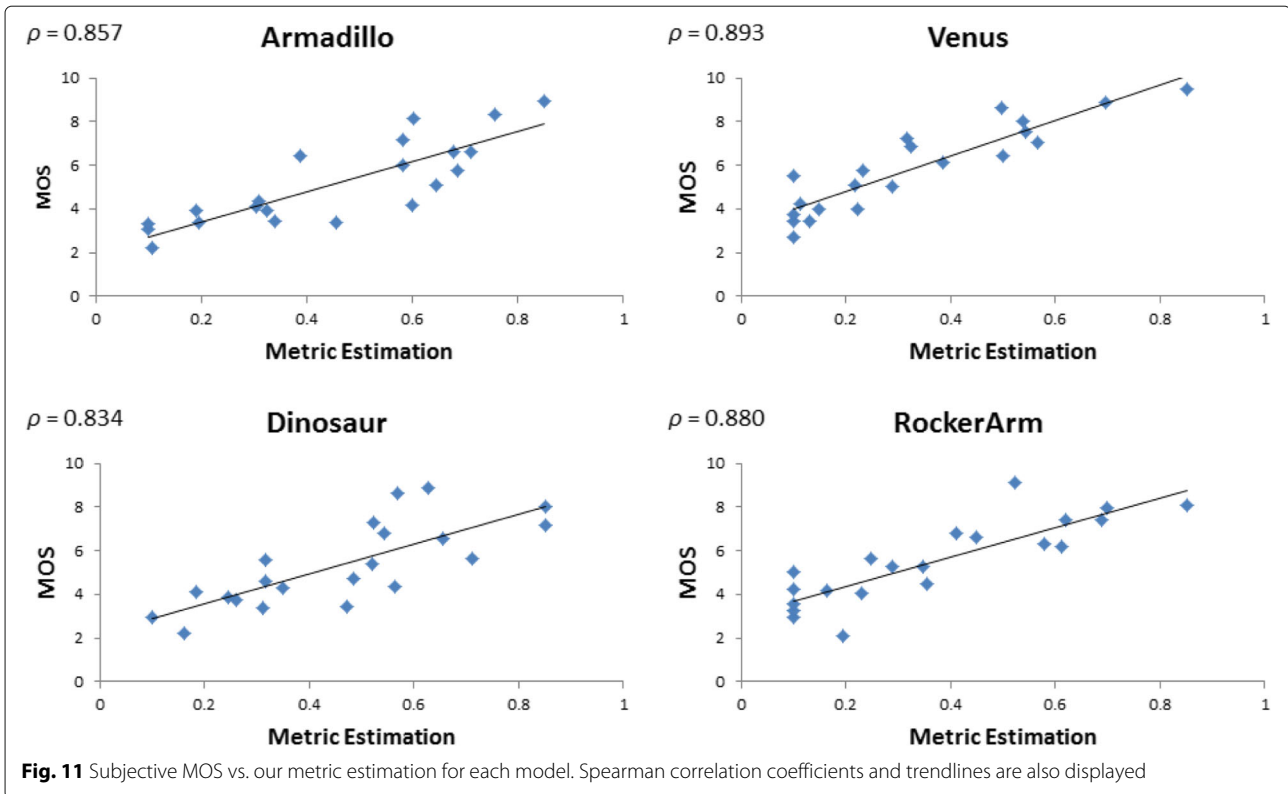
Figure 11 provides MOS vs. our metric estimation plots for each object in the dataset. Spearman correlation coefficients between MOS values and each of the provided

metric results were also calculated as listed in Table 3. We have not included the results for pure geometric metrics RMS and Hausdorff Distance since they are quite low. According to these results, our metric well correlates with the subjective responses and it is superior to most of the static metrics.

Perceptual error metrics designed for dynamic meshes to date that we are aware of are [46] and [45]. However, dynamic mesh datasets of [46] and [45] provide only one frame per animation and this is not sufficient for our metric to be applied on these datasets. Our metric also differs from these metrics in two ways. First, we do not require the test and reference meshes to be the same connectivity; for example, the test mesh could be a simplified version of the reference mesh, with a different number of vertices. Moreover, they are not directly comparable to our method since we produce a 3D map of local visible distortions as output, while they give a global error per dynamic mesh. Even though they also generate a 3D map in the interim steps and accumulate it to a single value, we do not have access to those interim steps. Hence, although developing a single error value per dynamic mesh is out of our purpose, to be able to compare our metric, we unified our 3D map into a single score by averaging the error values of each vertex. Then, we performed a second user experiment, following a similar design in [46].

In this experiment, we produced three modification levels per dynamic mesh given in Table 1, resulting in 12 animations. Using the MeshLab [8] tool, we applied random vertex displacement filter by varying the





maximum displacement parameter (The parameter was set as 0.1, 0.2, and 0.3 for modification levels 1, 2, and 3, respectively).

During the experiments, given the non-modified animation as reference, the subjects were asked to assign a score of 0, 1, 2, or 3 to the modified animation. In this evaluation scheme, 0 means that there is no perceptible difference between the reference and test animations. Evaluations of ten subjects were combined by calculating the mean opinion score (MOS) per modified mesh. Then, the correlation between the metric outputs and MOS values was calculated.

MOS vs. metric estimation plot in Fig. 12 reveals an almost linear relationship. Pearson and Spearman correlation coefficients for each mesh are also listed in Table 4.

Table 3 Spearman correlation coefficients for each model and metric

	Armadillo	Venus	Dinosaur	Rocker Arm
Our metric	0.86	0.89	0.79	0.88
MSDM [26]	0.84	0.86	0.70	0.88
3DWPM2 [11]	0.71	0.26	0.47	0.29
3DWPM1 [11]	0.64	0.68	0.59	0.85
GL1 [22]	0.68	0.91	0.05	0.02
GL2 [41]	0.76	0.89	0.22	0.18

Although the meshes used in the experiments are different; considering that the correlation coefficients in [46] varies between 0.92 and 0.98, our results are comparable to the state-of-the-art. We see that the correlation is very high (> 0.9) in this second experiment. This is because assigning an overall score to the given dynamic mesh is an easier task than marking the locations that are perceived different. The main purpose of this study is to produce a 3D map of visible distortions rather than generating an overall quality estimation per mesh.

5.3 Performance evaluation

5.3.1 Resolution of the spatiotemporal volume

The resolution of the spatiotemporal volume at each dimension affects the success of our method. In order to investigate this effect, we also performed several runs of our algorithm with varying voxel resolutions and calculated correlation coefficients for each run. We changed the *minResolution* parameter in Eq. 9, which determines the length of the spatiotemporal volume at each dimension, in proportion to the length of the bounding box of the mesh.

Figure 13 plots the correlation coefficients with respect to the *minResolution* parameter in Eq. 9. The plot includes the mean results of all the meshes. We see that the correlation is very low when *minResolution* is 10. Then, it starts to increase rapidly with the increasing resolution to

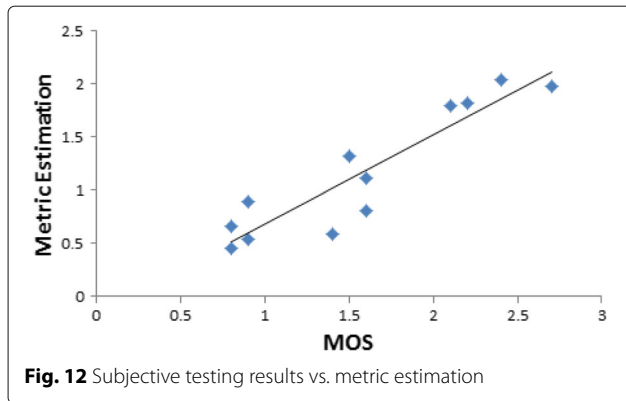


Fig. 12 Subjective testing results vs. metric estimation

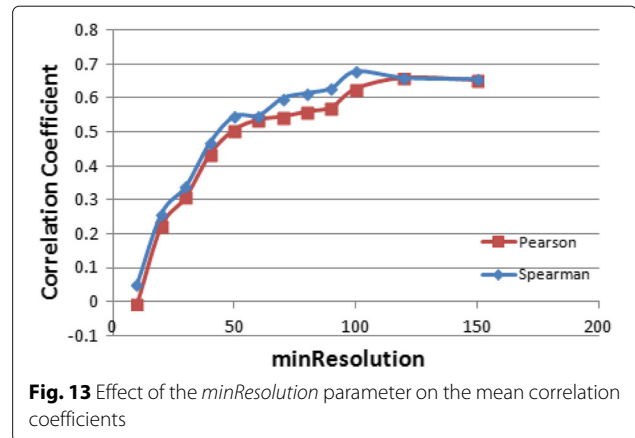


Fig. 13 Effect of the *minResolution* parameter on the mean correlation coefficients

a certain extent. After a while, for about $minResolution > 50$, the increase rate drops. For $minResolution > 100$, mean correlation settles to the band of 0.6 – 0.7 and increasing the resolution no further improves the accuracy.

Table 5 lists the strength of the correlation with respect to the *minResolution* parameter, for each mesh. One can observe that the correlation coefficients generally increase with the increasing resolution. When the resolution is too small, too many vertices fall in a single voxel, thus the result is not accurate. As the resolution gets higher, estimation is more accurate but the computational cost also increases. Moreover, incrementing the resolution does not improve the performance radically after a certain value.

According to our experiments, we drew a new heuristic to calculate the *minResolution* parameter. It is not desired to have too small resolution that allows many vertices to fall into the same voxel. So, we aim to distribute the vertices to different voxels as much as possible. We start with the assumption that vertices are distributed homogeneously. We also know that a mesh is generally represented with the vertices located on the surface and inside of the mesh is empty. Hence, we can assume that vertices are located on the facets of the bounding box. More conservatively, we take the facet of the AABB with the minimum area and obtain a resolution that allows distributing all the N vertices of the mesh to this facet

homogeneously. For this purpose, we first calculate the proportions of the facets of the AABB (w, h , and d in Eq. 9). Then, we can express each dimension as a function of some constant k (such that wk, hk, dk). If we select the minimum two of these dimensions as min_1 and min_2 , we can distribute N vertices to the facet of minimum area with $k = \sqrt{N / (min_1 * min_2)}$. We can then substitute this k value as the *minResolution* parameter.

This heuristic results in the following approximate *minResolution* values for *Camel, Elephant, Hand*, and *Horse* meshes, respectively: 100, 200, 90, and 60. According to Table 5, these values provide high correlations.

In summary, the resolution of the spatiotemporal volume has a significant impact on the estimation accuracy and computational cost of our method. Our heuristic to calculate the resolution of the volume works well. Alternatively, a more intelligent algorithm that considers the distribution and density of the vertices along the mesh bounding box could produce better estimations.

5.3.2 Processing time

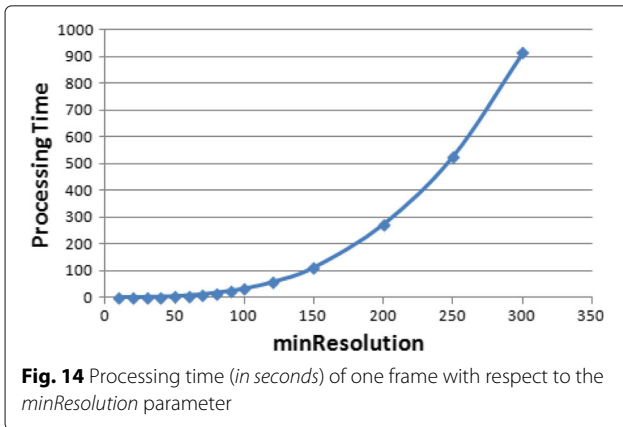
We monitored the processing time of our algorithm on a 3.3 GHz PC. As mentioned before, the resolution of the spatiotemporal volume, namely *minResolution* parameter in Eq. 9, determines the running time of our method. Figure 14 displays the change in the running time of our metric (without preprocessing) per frame, with respect to

Table 4 Pearson (r) and Spearman (ρ) correlation coefficients for each mesh

	Pearson $_r$	Spearman $_\rho$
Camel	0.926	0.937
Elephant	0.939	0.972
Hand	0.949	0.941
Horse	0.988	0.948
Overall	0.921	0.883

Table 5 Effect of the *minResolution* parameter on the correlation strengths of each mesh

	30	60	90	120	150
Camel	Weak	Modest	High	High	High
Elephant	Weak	Weak	Weak	Modest	Modest
Hand	Weak	Modest	High	High	High
Horse	Modest	High	High	High	High



the *minResolution* parameter. Note that in our method, frames of the animation can be processed in parallel. Hence, processing time of the animation is determined by the processing time of one frame. The figure implies that processing time changes in proportion to the cube of the *minResolution* parameter, expectedly.

Table 6 includes the approximate processing times for several meshes, along with their vertex count and *minResolution* parameter calculated according to our heuristic described in Section 5.3.1. As the table indicates, our metric cannot be used in real-time applications in its current form. However, it is possible to improve the performance by processing the spatiotemporal volume on GPU or employing more efficient data structures which process only the non-empty voxels. Another improvement possibility is to use lookup tables for CSF and Difference of Mesa (dom) filters, instead of calculating them on-the-fly.

6 Conclusions

In this paper, our aim is to provide a general-purpose visual quality metric for dynamic triangle meshes since it is a costly process to accomplish subjective user evaluations. For this purpose, we propose a full-reference perceptual quality estimation method based on the well-known VDP approach by Daly [14]. Our approach accounts for both spatial and temporal sensitivity of the HVS. As the output of our algorithm, we obtain a 3D probability map of visible distortions. According to our formal

Table 6 Processing times (seconds) for several meshes

	# Vertices	minResolution	Time
Horse	8 K	60	8
Camel	21 K	100	33
Elephant	42 K	200	274
Venus	100 K	300	915

experimental study, our perceptually-aware quality metric produces promising results.

The most significant distinction of our method is that it handles animated 3D meshes; since most of the studies in the literature omit the effect of temporal variations. Our method is independent of connectivity, shading, and material properties; which offers a general-purpose quality estimation method that is not application-specific. It is possible to measure the quality of 3D meshes that are distorted by a modification method which changes the connectivity or number of vertices of the mesh. Moreover, the number of vertices in the mesh does not have a significant impact on the performance of the algorithm. The algorithm can also account for static meshes. The proposed method is even applicable to the scenes containing multiple dynamic or static meshes. More importantly, the representation of the input mesh is not limited to triangle meshes and it is possible to apply the method on point-based surface representation. Lastly, we provide an open dataset including subjective user evaluation results for 3D dynamic meshes.

The main drawback of our method is the computational complexity due to 4D nature of the spatiotemporal volume. However, we overcome this problem to some extent by using a time window approach which processes a limited number of consecutive frames. Furthermore, a significant amount of speed-up may be obtained by processing the spatiotemporal volume in GPU.

As a future work, we aim to perform a more comprehensive user study, investigating the effects of several parameters. Another possible research direction is to integrate visual attention and saliency mechanism to the system.

Appendix

Subjective user evaluation dataset

Supplementary material consisting of the subjective user evaluation results can be downloaded from the following link: <http://cs.bilkent.edu.tr/~zeynep/DynamicMeshVQA.zip>.

The supplemental material includes the mesh files in *off* format and has the following directories:

- *Metric output* directory includes the results of our algorithm for each mesh used in the experiments.
- *Reference* directory includes the original mesh animations.
- *Test* directory includes the modified mesh animations.
- *User responses* directory includes the user evaluations of twelve subjects and the mean subjective responses.

Acknowledgements

We would like to thank all those who participated in the experiments for this study.

Authors' contributions

ZCY and TC developed the methodology together. ZCY conducted the experimental analysis and drafted the manuscript. TC composed the Related Work section and performed the proofreading and editing of the overall manuscript. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Author details

¹Faculty of Engineering, Celal Bayar University, Muradiye/Manisa, Turkey.

²Computer Engineering Department, TED University, 06420 Kolej/Ankara, Turkey.

Received: 29 October 2015 Accepted: 19 December 2016

Published online: 25 January 2017

References

1. S Albin, G Rougeron, B Peroche, A Tremeau, Quality image metrics for synthetic images based on perceptual color differences. *IEEE Trans. Image Process.* **11**(9), 961–971 (2002)
2. TO Aydin, M Čadík, K Myszkowski, HP Seidel, *ACM Transactions on Graphics (TOG)*, vol. 29, *Video quality assessment for computer graphics applications*. (ACM, New York, 2010), p. 161
3. PG Barten, *Contrast sensitivity of the human eye and its effects on image quality*, vol. 21. (SPIE Optical Engineering Press, Washington, 1999)
4. C Blakemore, FW Campbell, On the existence of neurons in the human visual system selectively sensitive to the orientation and size of retinal images. *J. Physiol.* **203**(1), 237–260 (1969)
5. MR Bolin, GW Meyer, in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques, SIGGRAPH '98*. A perceptually based adaptive sampling algorithm (ACM, New York, 1998), pp. 299–309
6. A Bulbul, TK Çapın, G Lavoué, M Preda, Assessing visual quality of 3-D polygonal models. *IEEE Signal Process. Mag.* **28**(6), 80–90 (2011)
7. X Chen, A Golovinskiy, R Funkhouser, A benchmark for 3D mesh segmentation. *ACM Trans. Graph. (Proc. SIGGRAPH)*. **28**(3), 73 (2009)
8. P Cignoni, M Corsini, G Ranzuglia, Meshlab: an open-source 3d mesh processing system. *ERCIM News.* **73**, 45–46
9. P Cignoni, C Rocchini, R Scopigno, Metro: measuring error on simplified surfaces. *Comput. Graph. Forum.* **17**(2), 167–174 (1998)
10. I Cleju, D Saupe, in *Proceedings of the 3rd symposium on Applied perception in graphics and visualization, APGV '06*. Evaluation of supra-threshold perceptual metrics for 3d models (ACM, New York, 2006), pp. 41–44
11. M Corsini, E Gelasca, T Ebrahimi, M Barni, Watermarked 3-D mesh quality assessment. *IEEE Trans. Multimed.* **9**(2), 247–256 (2007)
12. M Corsini, MC Larabi, G Lavoué, LVáša Petřík O, K Wang, *Computer Graphics Forum, Perceptual metrics for static and dynamic triangle meshes*. (Wiley Online Library, 2012)
13. S Daly, Engineering observations from spatiovelocity and spatiotemporal visual models. *Human Vision Electron Imaging III.* **3299**, 180–191 (1998)
14. SJ Daly, in *SPIE/IS&T 1992 Symposium on Electronic Imaging: Science and Technology*. Visible differences predictor: an algorithm for the assessment of image fidelity, (1992), pp. 2–15. International Society for Optics and Photonics
15. L Dong, Y Fang, W Lin, C Deng, C Zhu, HS Seah, Exploiting entropy masking in perceptual graphic rendering. *Signal Process Image Commun.* **33**, 1–13 (2015)
16. L Dong, Y Fang, W Lin, HS Seah, Perceptual quality assessment for 3d triangle mesh based on curvature. *IEEE Trans. Multimed.* **17**(12), 2174–2184 (2015)
17. R Eriksson, B Andren, KE Brunnstroem, in *Photonics West'98 Electronic Imaging, Modeling the perception of digital images: a performance study*. International Society for Optics and Photonics, (1998), pp. 88–97
18. JA Ferwerda, P Shirley, SN Pattanaik, DP Greenberg, in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques, SIGGRAPH '97*. A model of visual masking for computer graphics (ACM Press/Addison-Wesley Publishing Co., New York, 1997), pp. 143–152
19. E Gelasca, T Ebrahimi, M Corsini, M Barni, in *Image Processing, 2005. ICIP 2005. IEEE International Conference on Image Processing*. Objective evaluation of the perceptual quality of 3D watermarking, vol. 1 (IEEE, 2005), pp. 1–241
20. J Guo, V Vidal, A Baskurt, G Lavoué, in *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception*. Evaluating the local visibility of geometric artifacts (ACM, New York, 2015), pp. 91–98
21. I Howard, B Rogers, *Seeing in Depth*, (Oxford University Press, 2008)
22. Z Karni, C Gotsman, in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. Spectral compression of mesh geometry (ACM, New York, 2000), pp. 279–286
23. D Kelly, Motion and vision.ii. stabilized spatio-temporal threshold surface. *JOSA.* **69**(10), 1340–1349 (1979)
24. SJ Kim, SK Kim, CH Kim, in *Computer Graphics and Applications, 2002. Proceedings. 10th Pacific Conference on*. Discrete differential error metric for surface simplification (IEEE, 2002), pp. 276–283
25. G Lavoué, in *Computer Graphics Forum*. A multiscale metric for 3d mesh visual quality assessment, vol. 30 (Wiley Online Library, 2011), pp. 1427–1437
26. G Lavoué, ED Gelasca, F Dupont, A Baskurt, T Ebrahimi, in *Optics & Photonics*. Perceptually driven 3d distance metrics with application to watermarking (International Society for Optics and Photonics, 2006). 63,120L–63,120L
27. G Lavoué, MC Larabi, L Vasa, On the efficiency of image metrics for evaluating the visual quality of 3d models. *IEEE Trans. Vis. Comput. Graph.* **22**(8), 1987–1999 (2015)
28. G Lavoué, R Mantiuk, in *Visual Signal Quality Assessment*. Quality assessment in computer graphics (Springer, 2015), pp. 243–286
29. C Lee, A Varshney, D Jacobs, *Mesh saliency*. (ACM, New York, 2005), pp. 659–666
30. B Li, GW Meyer, RV Klassen, in *Photonics West'98 Electronic Imaging*. Comparison of two image quality models, (1998), pp. 98–109
31. W Lin, CC Jay Kuo, Perceptual visual quality metrics: a survey. *J. Vis. Commun. Image Represent.* **22**(4), 297–312 (2011)
32. P Longhurst, A Chalmers, in *Proceedings of the Theory and Practice of Computer Graphics 2004 (TPCG'04)*. User validation of image quality assessment algorithms (IEEE Computer Society, Washington, 2004), pp. 196–202
33. J Lubin, A visual discrimination model for imaging system design and evaluation. *Vision models for target detection and recognition.* **2**, 245–357 (1995)
34. BD Luebke, JD Watson, M Cohen, A Reddy, *Varshney, Level of Detail for 3D Graphics*. (Elsevier Science Inc., New York, 2002)
35. K Myszkowski, P Rokita, T Tawara, Perception-based fast rendering and antialiasing of walkthrough sequences. *IEEE Trans. Vis. Comput. Graph.* **6**(4), 360–379 (2000)
36. G Nader, K Wang, F Hetroy-Wheeler, F Dupont, Just noticeable distortion profile for flat-shaded 3d mesh surfaces. *IEEE Trans. Vis. Comput. Graph.* **22**(11), 2423–2436 (2015)
37. Y Pan, Li Cheng, A Basu, Quality metric for approximating subjective evaluation of 3-D objects. *IEEE Trans. Multimed.* **7**(2), 269–279 (2005)
38. G Ramnarayanan, J Ferwerda, B Walter, K Bala, in *ACM SIGGRAPH 2007 papers, SIGGRAPH '07*. Visual equivalence towards a new standard for image fidelity (ACM, New York, 2007)
39. M Ramasubramanian, SN Pattanaik, DP Greenberg, in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. SIGGRAPH '99, A perceptually based physical error metric for realistic image synthesis (ACM Press/Addison-Wesley Publishing Co., New York, 1999), pp. 73–82
40. BE Rogowitz, HE Rushmeier, in *Photonics West 2001-Electronic Imaging*. Are image quality metrics adequate to evaluate the quality of geometric objects?, (2001), pp. 340–348. International Society for Optics and Photonics
41. O Sorkine, D Cohen-Or, S Toledo, in *Symposium on Geometry Processing*. High-pass quantization for mesh encoding (Citeseer, 2003), pp. 42–51
42. RW Sumner, J Popović, in *ACM Transactions on Graphics (TOG)*. Deformation transfer for triangle meshes, vol. 23 (ACM, New York, 2004), pp. 399–405
43. R Taylor, Interpretation of the correlation coefficient: a basic review. *J. Diagn. Med. Sonography.* **6**(1), 35–39 (1990)
44. F Torkhani, K Wang, JM Chassery, A curvature-tensor-based perceptual quality metric for 3d triangular meshes. *Mach. Graph. Vis.* **23**(1-2), 59–82 (2014)

45. F Torkhani, K Wang, JM Chassery, Perceptual quality assessment of 3d dynamic meshes: subjective and objective studies. *Signal Process Image Commun.* **31**, 185–204 (2015)
46. L Vasa, V Skala, A perception correlated comparison method for dynamic meshes. *IEEE Trans. Vis. Comput. Graph.* **17**(2), 220–230 (2011)
47. I Wald, Utah 3d animation repository. <http://www.sci.utah.edu/~wald/animrep/>. Accessed 8 Jan 2017
48. K Wang, F Torkhani, A Montanvert, A fast roughness-based approach to the assessment of 3d mesh visual quality. *Comput. Graph.* **36**(7), 808–818 (2012)
49. Z Wang, HR Sheikh, AC Bovik, No-reference perceptual quality assessment of JPEG compressed images. *Proceedings of IEEE International Conference on Image Processing 2002.* **1**, 477–480 (2002)
50. B Watson, A Friedman, McA Gaffey, in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. SIGGRAPH '01, Measuring and predicting visual fidelity (ACM, New York, 2001), pp. 213–220
51. H Yee, S Pattanaik, DP Greenberg, Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. *ACM Trans. Graph. (TOG).* **20**(1), 39–65 (2001)

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Immediate publication on acceptance
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ springeropen.com
