Front. Comput. Sci.
DOI
Xiang FENG: A perceptual quality metric for 3D triangle meshes based on spatial pooling
RESEARCH ARTICLE

A perceptual quality metric for 3D triangle meshes based on spatial pooling

Xiang FENG (⌧) [1,2,3], Wanggen WAN[1,3], Richard Yi Da Xu[2], Haoyu CHEN[1,3], Pengfei LI[1,3], J. Alfredo Sánchez[4]

1 School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China
2 Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW 2007, Australia
3 Institute of Smart City, Shanghai University, Shanghai 200444, China
4 Department of Computing, Electronics and Mechatronics, Universidad de las Américas Puebla (UDLAP), Mexico

© Higher Education Press and Springer-Verlag Berlin Heidelberg 2016

Abstract In computer graphics, various processing operations are applied on 3D triangle meshes and these processes often involve distortions, which affect the visual quality of surface geometry. In such context, perceptual quality assessment of 3D triangle meshes becomes a crucial issue. In this paper, we propose a new objective quality metric for assessing the visual difference between a reference mesh and a corresponding distorted mesh. Our analysis indicates that the overall quality of a distorted mesh is sensitive to the distortion distribution. The proposed metric is based on a spatial pooling strategy and statistical descriptors of distortion distribution. We generate a perceptual distortion map for vertices in reference mesh while taking into account the visual masking effect of human visual system. The proposed metric extracts statistical descriptors from distortion map as the feature vector to represent the overall mesh quality. With the feature vector as input, we adopt Support Vector Regression model to predict the mesh quality score. We validate the performance of our method with three publicly available databases, and the comparison with state-of-the-art metrics demonstrates the superiority of our method. Experimental results show that our proposed method achieves a high correlation between objective assessment and subjective scores.

Keywords Mesh Quality Assessment, Spatial Pooling, Statistical Descriptors, Support Vector Regression, Visual Masking

1 Introduction

With advances in telecommunication, multimedia and computer hardware, 3D digital data has been widely used in various industrial applications. As the most commonly used representation of 3D content, 3D triangle mesh is subjected to a wide variety of processes such as compression and watermarking. These processes are likely to involve distortions that degrade the visual quality of 3D mesh. It is important to evaluate the influence of visual distortions on the perceived mesh quality. Since subjective scoring is impractical in most applications, it is necessary to develop objective metrics that produce a score to accurately reflect the visual quality of a distorted triangle mesh.

Most existing Mesh Visual Quality (MVQ) metrics construct a local quality map (distortion map) that provides vertex-wise or edge-wise quality (distortion), such as MSDM [1], MSDM2 [2], DAME [3], FMPD [4], TPDM [5], Lu [6], and then pool the local quantities into a final score that reflects the overall quality of the distorted mesh. The commonly used pooling strategies are average weighting [3]-[4], [6] and weighted Minkowski sum [1]-[2], [5]. However, the pooling strategy of objective scores in mesh quality assessment is less investigated. Existing mesh quality metrics choose a specific pooling strategy in their respective algorithm design. To our knowledge, there is still not a clear consensus on which pooling strategy is suitable for various MVQ assessment scenarios. The average weighting scheme assumes that the contributions of local distortions to the final quality score are independent of the locations of a mesh. But this method has an obvious problem. When only part of the mesh is extremely distorted while the remaining part keeps uncorrupted, the average weighting scheme would overestimate the mesh quality. The Minkowski sum strategy [1] [2] is employed for distortion pooling mainly because when the summation exponent increases, low quality region will be emphasized, which is consistent with human visual system (HVS). However, it is non-trial to determine the proper summation exponent which is generally chosen in an ad-hoc way. In addition, the Minkowski sum assumes that distortions at different locations of a mesh are statistically independent, which lacks sufficient theoretical support particularly in complex HVS.

In the image visual quality (IVQ) assessment community, a similar two-step quality pooling procedure was employed [7]-[9]. Recently, Li et al. [10] proposed a spatial pooling strategy for image quality assessment based on the statistical analysis of a quality map. They extracted statistical descriptors from quality map to represent the overall image quality, and showed that the overall image quality is sensitive to the quality map distribution. We believe that there are some similarities between image quality assessment and mesh quality assessment since both methods are based on human observation. The studies in [1] and [7] at some degree illustrated the similarity of HVS in image quality assessment and mesh quality assessment. The image quality assessment community has conducted significant studies on using machine learning techniques to support IVQ assessment [11]-[17], while most existing mesh quality assessment approaches [1]-[6] attempt to model the HVS explicitly. Lavoue et al. [18] proposed a perceptual evaluation metric based on multiple attributes using machine learning techniques. This metric did not take the characteristics of HVS into account, and cross-database performance was not presented to justify its generalization capability. We believe that machine learning techniques can play a significant role in mesh quality assessment especially when more subjective ground-truth data are available in the future.

In this paper, we propose an objective quality metric for assessing the visual quality of distorted meshes. Our proposed method extends the spatial pooling strategy [10] in image quality assessment to mesh quality assessment. We analyze the relationship between the distortion distribution and the overall mesh quality, and extract statistical descriptors of distortion distribution as the feature vector of a distorted mesh to characterize the quality degradation. The Support Vector Regression (SVR) model is used to learn the complex nonlinear relationship between distortion distribution and overall quality score. In this paper, we employ TPDM [5] metric to generate the vertex-level local distortion. However, our method can be generalized to other local quality or distortion based MVQ metrics [1]-[4], [6].

The remainder of this paper is organized as follows: We present related work on MVQ assessments in Section 2. Section 3 gives an overview of the proposed method. In Section 4, we give a brief introduction to TPDM metric, and analyze the relationship between distortion distribution and overall mesh quality. In Section 5, we introduce the involved SVR model, which is used to learn the nonlinear relationship between distortion distribution and overall quality score. In Section 6, we evaluate the performance of the proposed method on three publicly available databases. We draw conclusions and suggest future work in Section 7.

2 Related work

In the last decade, considerable research works on objective assessment of 2D IVQ have been conducted [7], [19]-[20]. However, research on objective assessment of 3D MVQ is still in its infancy and only few objective metrics have been proposed. Detailed reviews of 3D MVQ metrics can be found in [21]-[22]. Existing studies reveal that classical geometric distances, root mean squared error (RMS) and Hausdorff Distance (HD) have poor correlation with human visual perception [22], and image-based metrics are not suitable for evaluating perceptual quality of 3D mesh [22]-[23]. Thus, the research community has analyzed the model geometry to evaluate perceptual quality of 3D meshes and developed model-based metrics. In contrast to classical geometric metrics, the model-based metrics are perception-aware and have significantly higher correlation with human visual perception. The perceptual quality refers to the quality that is produced by a perception-aware metric.

A first attempt on model-based metric, to our knowledge, was proposed by Karni and Gotsman [24] to properly evaluate their spectral mesh compression algorithm. They considered Geometric Laplacian of vertex coordinates together with vertex coordinates in order to evaluate the distance between two 3D models. This metric was subsequently enhanced by Sorkine et al. [25] by increasing weight associated to Geometric Laplacian. These two metrics pioneered the studies on perceptual quality assessment of 3D meshes. Corsini et al. [26] proposed two perceptual metrics, 3DWPM1 and 3DWPM2, for visual quality assessment in the mesh watermarking application.

Lavoue et al. proposed a perceptual metric called Mesh Structural Distortion Measure (MSDM) [1], which extends the well-known image quality metric, structural similarity (SSIM) [7] index to 3D mesh quality evaluation. They computed differences of curvature statistics over local windows between two compared meshes, and used Minkowski summation of distances over local windows to generate a final evaluation. Subsequently, Lavoue proposed an improved multiscale version MSDM2 [2] that allows the comparison of two meshes with arbitrary connectivities. Bian et al. [27] developed a MVQ metric based on strain energy that causes the deformation between the original mesh and a deformed mesh.

Vasa and Rus introduced a MVQ metric, Dihedral Angle Mesh Error (DAME) [3], by computing the differences in oriented surface dihedral angles between triangles in 3D meshes. This metric requires that the two triangle meshes should share the same connectivity. Wang et al. proposed a Fast Mesh Perceptual Distance (FMPD) [4] metric based on mesh local roughness measure derived from Gaussian curvature. Fakhri et al. proposed a MVQ metric, Tensor-based Perceptual Distance Measure (TPDM) [5], based on the distance between curvature tensors of two meshes under comparison. The metric adopts a roughness-based weighting of local tensor distance to account for the visual masking characteristic of HVS. This metric generates a final score via Minkowski pooling of area-weighted local tensor distances. This method does not require the meshes under comparison to share the same connectivities. Lu et al. [6] proposed a quality assessment metric based on curvature information. This metric requires the distorted models should have the same connectivity as the reference model. Tian and Alregib [28] and Pan et al. [29] investigated quality evaluation in the application of transmission of textured meshes, and concluded that image texture generally makes a greater contribution than model geometry in terms of perceptual quality.

Our contributions can be summarized as follows: First, we analyze the relationship between mesh distortion distribution and overall quality score. Our in-depth analysis indicates that overall mesh quality is sensitive to the distortion distribution. Second, we propose a spatial pooling strategy for mesh quality assessment, which extends the image quality assessment metric presented in [10]. We consider the distortion distribution instead of quality distribution because we find that distortions directly influence the overall quality degradation. Our method can be extended to other mesh quality metrics based on local distortion pooling [1]-[4], [6]. Third, we use cross-database validation strategy to demonstrate the generality of our method. This is, to our knowledge, the first attempt in mesh visual quality assessment. The experimental results confirm the robustness of our proposed method.

3 Overview of the proposed method

An overview of the proposed 3D mesh quality assessment method is illustrated in Fig.1. Given a reference mesh and a distorted mesh, the proposed method generates the distortion distribution, and extracts the statistics descriptors from the distortion distribution as the feature vector of the distorted mesh in terms of quality degradation. In this paper, we employ the TPDM [5] metric to generate perceptual distortion for each vertex in the reference mesh. Since TPDM metric takes into account the visual masking effect of HVS, our proposed method inherits this characteristic.

Our method follows the machine learning paradigm. We pair the statistical descriptors and Mean Opinion Score of each distorted mesh into a data sample. In the training stage, we feed the pairwise data sample into the training model to learn the relationship between the statistical descriptors and subjective score. In this paper, we use SVR model as the training model since SVR has been used in the IVQ assessment and it proves that SVR has a good regression and prediction performance [10, 13, 14]. The parameters of SVR are determined through cross validation on the training sample set. After the SVR model is learned, it will be used to predict the quality of distorted mesh in the testing stage.

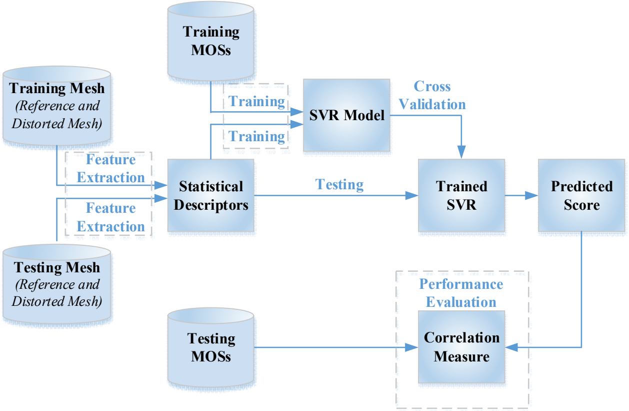


Fig. 1 Schematic diagram of the proposed 3D mesh quality assessment method

4 Spatial pooling strategy

In image quality assessment, Li et al. [10] indicated that overall image quality is sensitive to quality distribution. We heuristically construct a local distortion map for the distorted mesh, and then extract statistical descriptors from the distortion map distribution to characterize the overall quality degradation. We employ the TPDM [5] metric to generate the local distortion. The procedure of spatial pooling mainly consists of three steps:

- ●Step 1: Firstly, we use TPDM metric to generate a spatially local perceptual distortion for each vertex of reference mesh, and produce a distortion sequence which consists of all the spatially local perceptual distortions.
- ●Step 2: Then, we extract a feature vector from the distortion sequence to represent the overall mesh quality. We propose to use statistical descriptors as the feature vector of local distortions for each distorted mesh. The feature vector of each distorted mesh will be paired with the subjective score to form a data sample.
- ●Step 3: Lastly, we propose to use machine learning techniques to learn the nonlinear complex relationship between the feature vector and the subjective score. Specifically, we adopt Support Vector Regression model as the learning model, which will be introduced in detail in Section 5.

4.1 TPDM metric

The TPDM [5] metric evaluates the perceptual distance between the reference mesh and a distorted mesh based on the differences of curvature tensors. The metric accounts for the visual masking effect of HVS by roughness-based weighting of local tensor distance. The overall mesh quality measure is computed as the weighted Minkowski sum of local distances:

$$TPDM = \left(\sum_{i=1}^{N} \omega_i \left| LTPDM_{v_i} \right|^p \right)^{\frac{1}{p}} \tag{1}$$

where $LTPDM_{v_i}$ represents the local distance between vertex v_i in reference mesh and corresponding vertices in distorted mesh, and $\omega_i = s_i \big/ \sum_{i=1}^{N} s_i$ is the area weighting coefficient with s_i one third of total areas of all incident facets of v_i. The area weighting coefficient ω_i is used to improve the stability of the metric to variation of surface vertex sampling density. The summation exponent p is empirically chosen as $p = 2.5$. An AABB tree data structure is used to match each vertex v_i of reference mesh to corresponding facet $\overline{F}_i$ of distorted mesh. The local distance $LTPDM_{v_i}$ associated to v_i is computed as the barycentric interpolation of local perceptual distances between v_i and three vertices $v'_{i,1}$, $v'_{i,2}$ and $v'_{i,3}$ of facet $\overline{F}_i$.

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

where $LPD_{v_i, v'_{i,k}}$ is the local perceptual distance between vertex v_i in the reference mesh and vertex $v'_{i,k}$ of facet $\overline{F}_i$ in the distorted mesh. The local perceptual distance $LPD_{v_i, v'_{i,k}}$ accounts for the visual masking effect through incorporating roughness-based weights that are derived from principal surface directions and curvature amplitudes [5]. The local perceptual distance $LPD_{v_i, v'_{i,k}}$ between v_i and $v'_{i,k}$ is computed as:

$$LPD_{v_i, v'_{i,k}} = RW_i^{(\gamma)} \cdot RW_i^{(\kappa)} \cdot \left(\frac{\theta_{min}}{(\pi/2)} \delta_{\kappa_{min}} + \frac{\theta_{max}}{(\pi/2)} \delta_{\kappa_{max}} \right) \tag{3}$$

where $RW_i^{(\gamma)}$ and $RW_i^{(\kappa)}$ are the roughness-based weights derived from surface principal directions and curvature amplitudes in the 1-ring neighborhood of v_i respectively. θ_{min} is the angle between the curvature lines of γ_{min} and γ'_1 while θ_{max} is the angle between the lines of γ_{max} and γ'_2. $\delta_{\kappa_{min}}$ is the Michelson-like contrast of the curvature amplitudes κ_{min} and κ'_1 while $\delta_{\kappa_{max}}$ is the Michelson-like contrast of the curvature amplitudes κ_{max} and κ'_2.

In this paper, we use $LTPDM_{v_i}$ to represent the local distortion for each vertex of reference mesh, and construct a distortion map to provide vertex-level distortion evaluation. We choose TPDM metric due to the following reasons: Firstly, TPDM does not require the meshes under comparison to share the same connectivities and thus has a broad application prospect; secondly, TPDM metric integrates the features of HVS such as visual masking effect; thirdly, TPDM metric achieves comparably high correlation between objective assessment and subjective scores among state-of-the-art methods [6]; lastly, the source code of TPDM is freely available online [5], so it facilitates the implementation of our work.

4.2 Statistical descriptors

Li et al. [10] proposed an image quality assessment metric based on statistical analysis of the quality map. Their analysis shows the overall image quality is sensitive to quality distribution. They provided two feature vectors to characterize the quality map distribution: histogram and statistical descriptors. Their experimental results indicate that the method based on statistical descriptors achieves slightly better performance than the histogram based method. In this paper, we extract statistical descriptors from the distortion map to characterize the distortion distribution. Statistical descriptors together with the SVR model is referred to as the spatial pooling strategy. The statistical descriptors [10] include mean, standard deviation, minimum, maximum, first quartile, second quartile, and third quartile of the distortion distribution.

Let $M(i)$ denote local distortions which are sorted in an ascending order. Then the seven descriptors can be defined successively as follows:

$$Mean = \frac{1}{N}\sum_{i=1}^{N} M(i) \tag{4}$$

$$Std = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(M(i)-\overline{M(i)})^2} \tag{5}$$

$$Min = \min(M(i)) = M(1) \tag{6}$$

$$Max = \max(M(i)) = M(\mathrm{N}) \tag{7}$$

$$Q_1 = M((\mathrm{N}+1)/4) \tag{8}$$

$$Q_2 = M((\mathrm{N}+1)/2) \tag{9}$$

$$Q_3 = M(3(\mathrm{N}+1)/4) \tag{10}$$

The local distortion of each vertex is evaluated by the aforementioned TPDM metric. Here we refer to our method as TPDMSP since we apply the spatial pooling strategy to the TPDM metric.

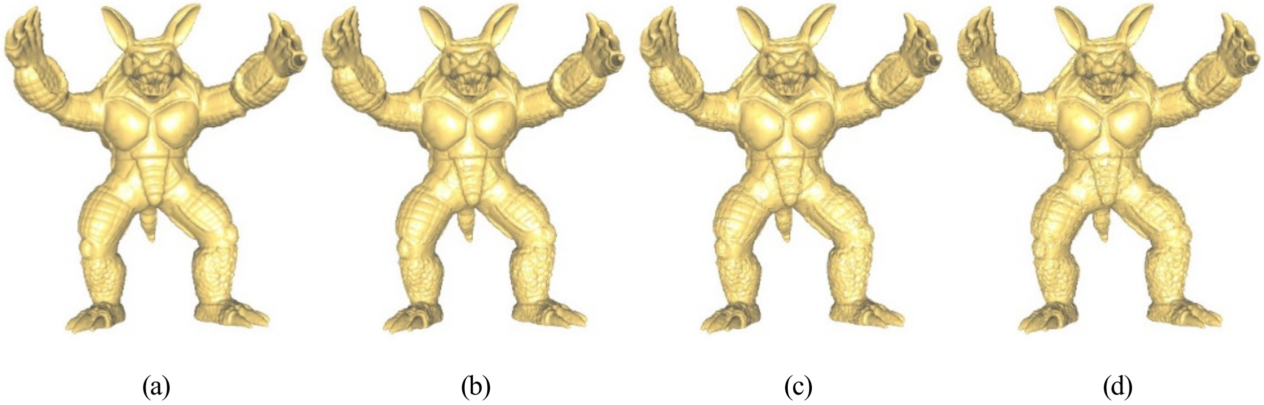
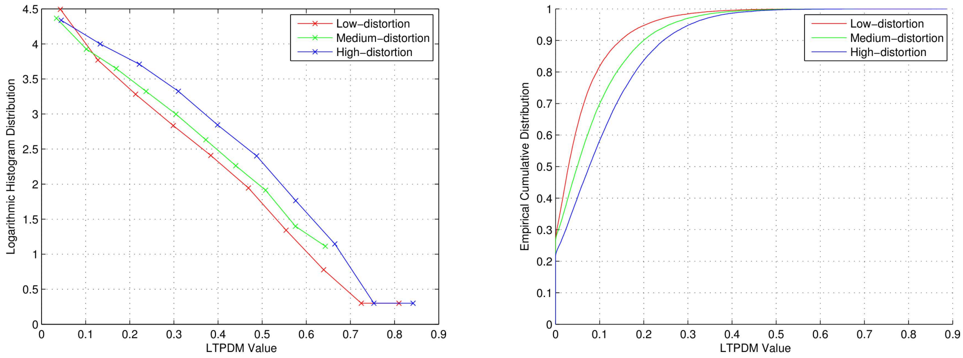


Fig. 2 Reference mesh and noise distorted meshes of the Armadillo Model. Their MOSs are successively 3.082, 3.393, 6.412 and 8.142. (a) Original Mesh. (b) Low-Distortion Mesh. (c) Medium-Distortion Mesh. (d) High-Distortion Mesh.

In order to demonstrate the relationship between the mesh distortion distribution and the overall quality score, we analyze the distortion distributions of three distorted meshes that suffer distortions of different strengths. Fig. 2 illustrates four meshes of the Armadillo Model from the LIRIS/EPFL general-purpose database [1]. The first mesh in Fig. 2(a) is the reference mesh while the other three meshes in Fig. 2(b), Fig. 2(c) and Fig. 2(d) are the distorted meshes with different distortion strengths. In the database, a subjective score between 0 (best) and 10 (worst) is given to each distorted mesh by each observer. A normalized Mean Opinion Score (MOS) is generated for each mesh by averaging the scores. The three distorted meshes are resulted by applying noise addition of different strengths to rough areas of the original mesh. Higher distortions correspond to more intense noise.

We compare each distorted mesh with the original mesh to construct the distortion map. *LTPDM* is computed as local distortion for each vertex in the original mesh. As illustrated in Fig. 3, we generate the logarithmic histogram distribution and empirical cumulative distribution (ECD) for each distortion map of three distorted Armadillo Models. In Fig. 3(a), we partition the distortion range into 10 equal ranges and build the histogram with 10 bins, then we use logarithm to represent the sample number in each cluster. In visual quality assessment, it is commonly recognized that a severe local distortion generally makes a greater contribution to the overall quality degradation than a slight local distortion. We also find that when only a very small portion of mesh is severely distorted, these distortions could not be discovered with the existence of a large amount of moderate distortions, especially when the topology of the mesh is complicated and the surface is highly sampled. Based on our analysis above, we focus on LTPDM value range [0.15, 0.65] in Fig. 3(a). We find that in this range, at a given LTPDM value bin, high-distortion always has the largest sample size while low-distortion has the smallest sample size. This keeps consistent with subjective scores since a large amount of high-distortions will generally lead to a low-quality score.

Figure 3(b) illustrates the empirical cumulative distribution for each distortion map of distorted meshes. This figure gives an intuitive comparison of three distortions in terms of statistical descriptors. For example, on three curves, at the second quartile point Q_2 where ECD is equal to 0.5, the corresponding LTPDM values for three distortions are respectively 0.029, 0.049 and 0.076, and sequentially increase. Similar comparative results are obtained for the third quartile Q_3. The values of statistical descriptors together with MOS of each distorted mesh can be found in Table 1. These analyses indicate that there is a strong relationship between the statistical descriptors and the overall mesh quality.



(a) (b)

Fig. 3 Distributions for each distortion map of three distorted Armadillo Models. (a) Logarithmic Histogram Distribution. (b) Empirical Cumulative Distribution.

Table 1 Statistical Descriptors and MOS of Each Distorted Mesh

Distorted Mesh	Statistical Descriptors							MOS
	Mean	Std	Min	Max	Q_1	Q_2	Q_3	
Low-Distortion Mesh	0.055	0.073	0	0.853	0	0.029	0.076	3.393
Medium-Distortion Mesh	0.077	0.090	0	0.677	0	0.049	0.118	6.412
High-Distortion Mesh	0.101	0.103	0	0.886	0.008	0.076	0.158	8.142

5 SVR-Based Fusion of Distortion Distribution

We assemble the statistical descriptors of distortion distribution as a feature vector x of the distorted mesh, and represent the overall mesh quality score Q as a function of the feature vector.

$$Q = f(x) \tag{11}$$

where f is the function mapping the vector x to the overall mesh quality score. Due to the complexity and limited knowledge of HVS, it is difficult to properly define such a function *a priori*. We use a machine learning technique to learn the nonlinear complex relationship between the feature vector and the perceptual quality score. In this paper, we use SVR to map the feature vector into a quality score by learning the underlying complex function. We exploit the advantage of machine learning, which has a strong capability for learning a complex relationship from ground-truth data. SVR has been well established and is widely used in visual quality assessments [10], [13]-[14]. A famous implementation of SVR is provided in LIBSVM [30]. SVR is used to find the optimal f based on training samples. As a popular regression method in the SVR family, given the training sample set of sample-target pairs $\langle x_i, \mathrm{y}_i \rangle$, $\varepsilon - SVR$ tries to find a function f subject to that the allowed max deviation between the target y_i and $f(\mathrm{x}_i)$ is ε for all training data [30]. The function to be learned is

$$f(x) = w^T \varphi(x) + \mathrm{b} \tag{12}$$

where w is the weight vector, $\varphi(x)$ is a nonlinear function of x, and b is the bias. We use the $\varepsilon - SVR$ method in our training process. Let x_i denote the feature vector of the ith mesh in the training mesh set, then the weight vector w and the bias are determined via training data subject to

$$| f(x_i) - y_i | \leq \varepsilon \tag{13}$$

for each ith training sample. The training vectors x_i are mapped into higher dimensional space by the function φ, and a linear separating hyperplane will be found with the maximal margin in the higher dimensional space. $K(x_i, x_j) \equiv \varphi(x_i)^T \varphi(x_j)$ is called the kernel function in the SVR training process. From several kernels available, in this paper, we chose the radius basis function (RBF) kernel as our kernel function since RBF kernel performs well in many applications [30]. The RBF kernel is expressed as follows:

$$K(x_i, x_j) = \exp(-\gamma \| x_i - x_j \|^2), \gamma > 0 \tag{14}$$

where γ is a positive parameter controlling the radius. We use a cross-validation strategy on the training data set to select the parameters $\{ C, \gamma, \varepsilon \}$.

6 Experimental results and analysis

6.1 Experimental protocol

In our experiments, we use three public databases to evaluate the performance of our proposed method. The LIRIS/EPFL general-purpose database [1] contains 4 reference meshes and 84 distorted meshes (21 distorted meshes for each reference mesh). For each reference mesh, two types of distortion including random noise addition and smoothing are applied with different strengths and at four locations (uniformly, smooth areas, rough areas, intermediate areas). Subjective quality scores are provided for each distorted mesh in the form of MOS ranging from 0 (best quality) to 10(worst quality). The LIRIS masking database [31] contains 4 reference meshes and 24 distorted meshes (6 distorted meshes for each reference mesh). Random noise with different strength is added in rough or smooth regions of the reference mesh to test visual masking effect in mesh quality assessment. MOS is provided for each distorted mesh in the range from 0 (worst quality) to 4 (best quality). The UWB compression database [3] contains 5 reference meshes and 63 distorted meshes (12 or 13 distorted meshes for each reference mesh). Thirteen types of compression distortions are applied on the reference meshes. MOS is provided for each distorted mesh in the range from 0 (best quality) to 1 (worst quality).

Compared with image quality assessment [10], 3D mesh quality assessment has a relatively small number of models in each subjective database. However, Support Vector Machine (SVM) has excellent performance even with a small training sample size [30]. The size of a proper training sample set depends on the length and the representational ability of feature vector. In this paper, the feature vector consists of seven scalars. We find that it is not appropriate to directly train SVR on the LIRIS masking database due to the small number of samples as there are only 24 distorted meshes for training and testing. The UWB compression database is not suitable for training either because the scores of distorted meshes for different reference meshes are uncorrelated, though the total number of distorted meshes is relatively high compared with the length of feature vector. In this paper, first, we train SVR model and test it within the LIRIS/EPFL general-purpose database to validate the overall performance of our proposed method. We compare our method with state-of-the-art methods in terms of prediction performance. We also train the SVR model on the whole LIRIS/EPFL general-purpose database, and then test it on both UWB compression database and LIRIS masking database to evaluate its generality and robustness.

The performances of MVQ metrics are evaluated by two measures: Pearson linear correlation coefficient and Spearman rank-order correlation coefficient. In this paper, these two measures are used to measure the correlation between prediction scores and Mean Opinion Scores of testing meshes. The Pearson linear correlation coefficient (PLCC or r_p) measures the prediction accuracy of objective metrics, while the Spearman rank-order correlation coefficient (SROCC or r_s) measures the prediction monotonicity [20]. Both values of PLCC and SROCC range from -1 to 1, where 1 indicates total positive correlation, -1 indicates total negative correlation, and 0 indicates no correlation. Suppose we have two datasets $x = \{x_1, x_2, \dots, x_n\}$ and $y = \{y_1, y_2, \dots, y_n\}$, both containing n values. The Pearson linear correlation coefficient r_p between dataset x and y is calculated as follows.

$$r_p = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2}} \tag{15}$$

where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ and $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$. After the datasets x and y are sorted in the same order, in either ascending or descending order, we generate two new datasets $X = \{X_1, X_2, \dots, X_n\}$ and $Y = \{Y_1, Y_2, \dots, Y_n\}$. The value of X_i is the rank of x_i in dataset x while the value of Y_i is the rank of y_i in dataset y. Let $d_i = X_i - Y_i$, the Spearman rank-order correlation coefficient r_s between datasets x and y is calculated as follows.

$$r_s = 1 - \frac{6\sum_{i=1}^{N} d_i^2}{n(n^2 - 1)} \tag{16}$$

The Spearman rank-order correlation coefficient is also regarded as the Pearson linear correlation coefficient between two ranked variables.

Existing metrics, such as [1], [5] use a psychometric function to conduct the fitting between the prediction scores and MOSs. Since our method directly learns the nonlinear function between the feature vector and the perceptual quality score, the prediction output of a testing distorted mesh will be a quality score, thus we do not have to map the prediction into a quality score in practical applications. However, for illustration purposes, we also apply a cumulative Gaussian psychometric function [32] to fit prediction scores and MOSs of testing meshes:

$$g(a,b,S) = \frac{1}{\sqrt{2\pi}}\int_{a+bS}^{\infty} e^{-(t^2/2)}dt \tag{17}$$

where S is the prediction score. Parameters a and b are obtained through fitting the prediction scores and MOSs of testing meshes.

6.2 Overall performance

We compare our method TPDMSP with state-of-the-art mesh quality assessment metrics, including Lu [6], TPDM [5], DAME [3], FMPD [4], MSDM2 [2], MSDM [1], 3DWPM [26], and so on. We separate the LIRIS/EPFL general-purpose database randomly into two non-overlapping mesh sets, in which 80% of data is used as training set and the other 20% is used as testing set. The PLCC and SROCC are evaluated between the prediction scores and MOSs of testing meshes. In order to get statistical significance, we repeat the procedure 50 times. We denote the median performance and optimal performance respectively as TPDMSP (m) and TPDMSP (o). We report the PLCC and SROCC performances on LIRIS/EPFL general-purpose database in Table 2.

Table 2 Correlation Coefficients of Different Objective Metrics on LIRIS/EPFL General-Purpose Database

Metrics	Performance Indicator (%)	
	PLCC	SROCC
HD	11.4	13.8
RMS	28.1	26.8
GL1	35.5	33.1
GL2	42.4	39.3
SF	7	15.7
3DWPM1	61.8	69.3
3DWPM2	49.6	49
MSDM	75	73.9
MSDM2	81.4	80.4
FMPD	83.5	81.9
DAME	75.2	76.6
TPDM	86.2	89.6
Lu	87.7	86.6
TPDMSP (m)	89.0	87.5
TPDMSP (o)	95.9	96.1

From Table 2, we can see that our TPDMSP (o) has much higher values for PLCC and SROCC than all state-of-the-art metrics. Our TPDMSP (m) has higher values for PLCC than all state-of-the-art metrics. Our spatial pooling strategy is based on local distortions which are computed by the TPDM metric, but our method can achieve a significant performance gain over TPDM. Since PLCC measures the prediction accuracy of objective metrics, this indicates that our method achieves better prediction accuracy. A possible reason may be that we use SVR to directly learn the nonlinear function between statistical descriptors and the overall mesh quality score, thus the prediction scores of testing meshes reflect the nonlinear relationship between local distortions and overall mesh quality score. The scatter plot of a set of testing meshes at a training/testing experiment is illustrated in Fig. 4. From the scatter plot, we can also see the proposed method performs well on LIRIS/EPFL general-purpose database.

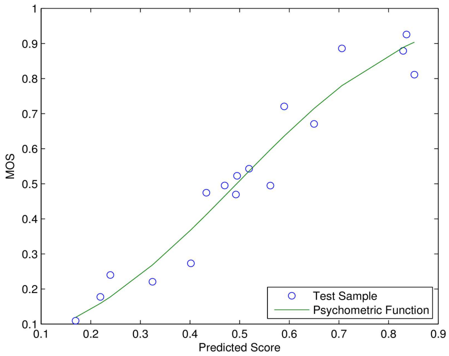


Fig. 4 Scatter plots of MOSs versus predicted scores for a set of testing meshes at a training/testing experiment in the LIRIS/EPFL general-purpose database. Both MOSs and predicted scores are normalized.

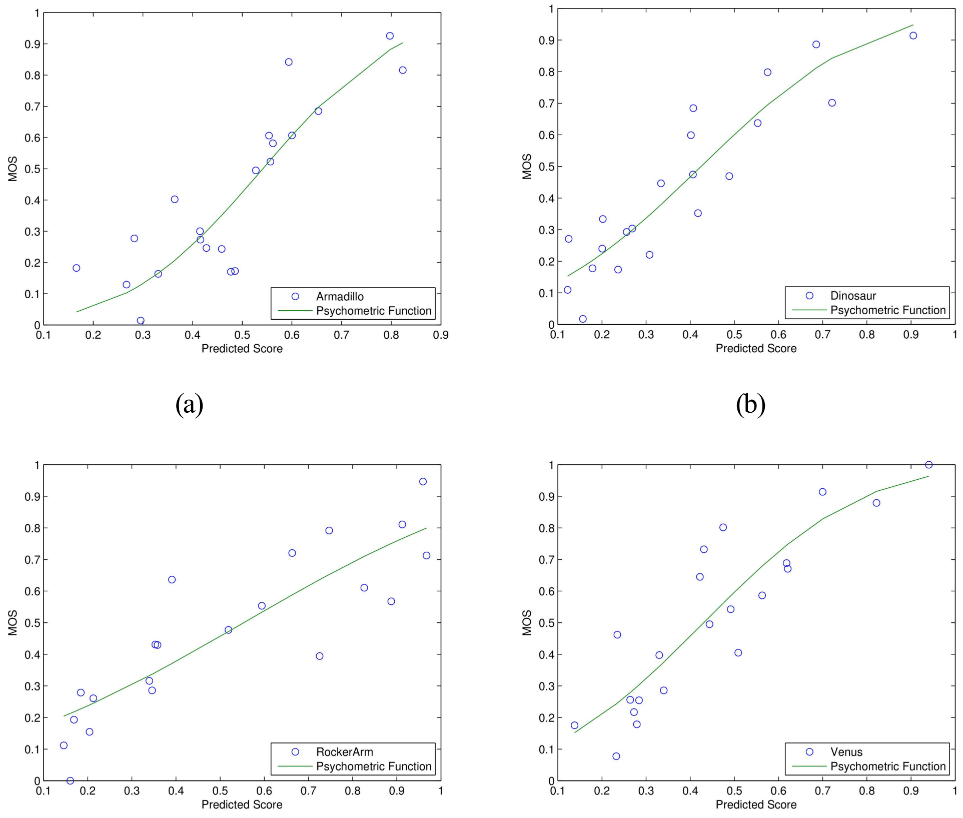
6.3 Inter-model generalization performance

In order to evaluate the performance of the proposed method on different geometry models, we test the inter-model generalization capability of our method. There are four models (Armadillo Model, Dinosaur Model, RockerArm Model, and Venus Model) in the LIRIS/EPFL general-purpose database. We use distorted meshes of three models for training and use distorted meshes of the fourth model for testing. In order to demonstrate the inter-model generalization capability of our method, we give a comparison between the inter-model generalization performances of our method and the direct evaluation performances of state-of-the-art metrics in Table 3. *Note that state-of-the-art metrics in Table 3 are not suitable for inter-model prediction because they are not based on machine learning paradigm*. The performances of these metrics for each geometry model in Table 3 are obtained by defining an explicit pooling function and determining parameters properly. But it is non-trivial to determine the explicit pooling function and the parameters in different experiment settings. Most state-of-the-art methods explicitly model the relationship between the local distortion and overall quality score via average weighting [3], [4] or weighted Minkowski sum [1], [2], and choose parameters of the explicit model in an ad-hoc way. Moreover, their performances are achieved by considering all meshes in the whole database while our method does not involve testing mesh in the training stage. The PLCC and SROCC performances of state-of-the-art metrics are extracted from published paper [4]. The inter-model generalization performances of our method are listed in the last row of Table 3. The *All models* performances of our method are obtained by averaging the four performances for each testing geometry model.

Table 3 Comparison Between Inter-Model Generalization Performances of Our Method and Direct Evaluation Performances of State-of-The-Art Metrics

Metric	Armadillo		Dinosaur		RockerArm		Venus		All models		Suitable for Training/Testing
	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	
HD	54.9	69.5	47.5	30.9	23.4	18.1	8.9	1.6	11.4	13.8	No
RMS	56.7	62.7	0.0	0.3	17.3	7.3	87.9	90.1	28.1	26.8	No
3DWPM1	59.7	65.8	59.7	62.7	72.9	87.5	68.3	71.6	61.9	69.3	No
3DWPM2	65.6	74.1	44.6	52.4	54.7	37.8	40.5	34.8	49.6	49.0	No
MSDM2	85.3	81.6	85.7	85.9	87.2	89.6	87.5	89.3	81.4	80.4	No
DAME	76.3	60.3	88.9	92.8	80.1	85.0	83.9	91.0	75.2	76.6	No
FMPD	83.2	75.4	88.9	89.6	84.7	88.8	83.9	87.5	83.5	81.9	No
Inter-Model Generalization	86.3	82.5	90.9	90.9	87.2	90.8	88.0	86.2	88.1	87.6	Yes

Table 3 indicates that our method achieves consistently high prediction for different testing models and our average performances for both PLCC and SROCC are highest in the *All models* column. These comparisons indicate our method has a good inter-model generalization capability with high prediction accuracy.



(a)

(b)

(c)

(d)

Fig. 5 Scatter plots of MOSs versus predicted scores for each geometry model in the inter-model generalization experiment. Both MOSs and predicted scores are normalized. (a) Armadillo model. (b) Dinosaur model. (c) RockerArm model. (d) Venus model.

Figure 5 (a)-(d) illustrate the scatter plots for distorted meshes corresponding to each geometry model that are used as testing meshes. For example, in Fig. 5 (a), all distorted meshes of the Armadillo model constitute the testing set, while all distorted meshes of the other three models constitute the training set. The psychometric fitting curve and the scatter points that represent the *Prediction-MOS* pairs are plotted. From the figures, we can see the *Prediction-MOS* pairs are fitted well by the Gaussian psychometric curve.

6.4 Performance analysis of training data

As the proposed method requires MOSs to learn the SVR model, we show how performance varies with different amounts of training data. We report the performance under six different partition proportions: 10%, 20%, 30%, 40%, 60%, 80% of distorted meshes are used for training and the remaining 90%, 80%, 70%, 60%, 40%, 20% of distorted meshes are used for testing. We denote PLCC (m) and SROCC (m) respectively as the median performance for PLCC and SROCC, while we denote PLCC (o) and SROCC (o) respectively as the optimal performance.

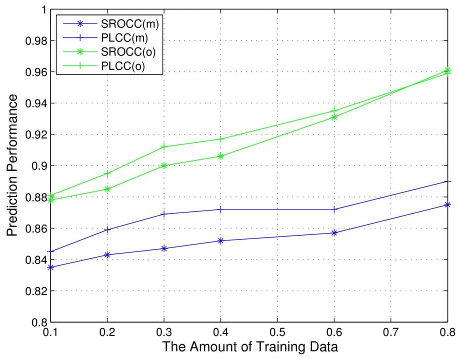


Fig. 6 Performances with different amounts of training data on LIRIS/EPFL General-Purpose Database

From Fig. 6, we can see both PLCC and SROCC generally improve when increasing the training data, and our method achieves acceptable performance even with a small training set.

6.5 Performance analysis of statistical descriptors

In the proposed spatial pooling method, we consider seven features in the statistical descriptors: *mean, standard deviation, minimum, maximum, the first quartile, the second quartile, and the third quartile* [10]. We investigate the contribution of each feature to the prediction performance. We use six out of seven features as input to learn the SVR model, and compare the results. We follow the assumption that leaving out a key feature will lead to a significant performance decrease and the resultant performances will be relatively low. As illustrated in Fig. 7, the *max* feature makes the most contribution among all features since both resultant PLCC (m) and SROCC (m) values are smallest in their respective group when leaving *max* out. This conclusion is consistent with subjective judgment of human evaluators because an intense local distortion always attracts more attention and would lead to severe degradation of overall quality. We can also see that the *min* feature makes the least contribution, the reason is that slight local distortions are usually unnoticeable. Among three quartiles, *the third quartile* makes the most contribution, this is because *the third quartile* represents more severe local distortions than the other two quartiles. The *standard deviation* feature makes more contributions than *mean* feature.

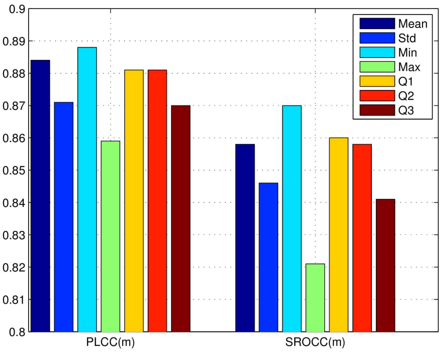


Fig. 7 Leave-one-out performance on LIRIS/EPFL General-Purpose Database

6.6 Cross-database validation performance

In this subsection, we evaluate the cross-database performance of our method to validate its generality and robustness. This is, to our knowledge, the first attempt to use one 3D distortion database to train a quality evaluation model and test the model with another 3D distortion database. It is expected that the SVR model trained on meshes of one database should be applicable to other databases. We train the SVR model on LIRIS/EPFL General-Purpose Database and test it on UWB compression database and LIRIS masking database. We find that it is not appropriate to apply the local distortion without considering surface area since there is much difference on the surface area size and vertex sample density between three different databases. We use the surface area-weighted local distortion $WLTPDM_{v_i}$ and follow the spatial pooling strategy by extracting statistical descriptors from the distortion distribution as the feature vector.

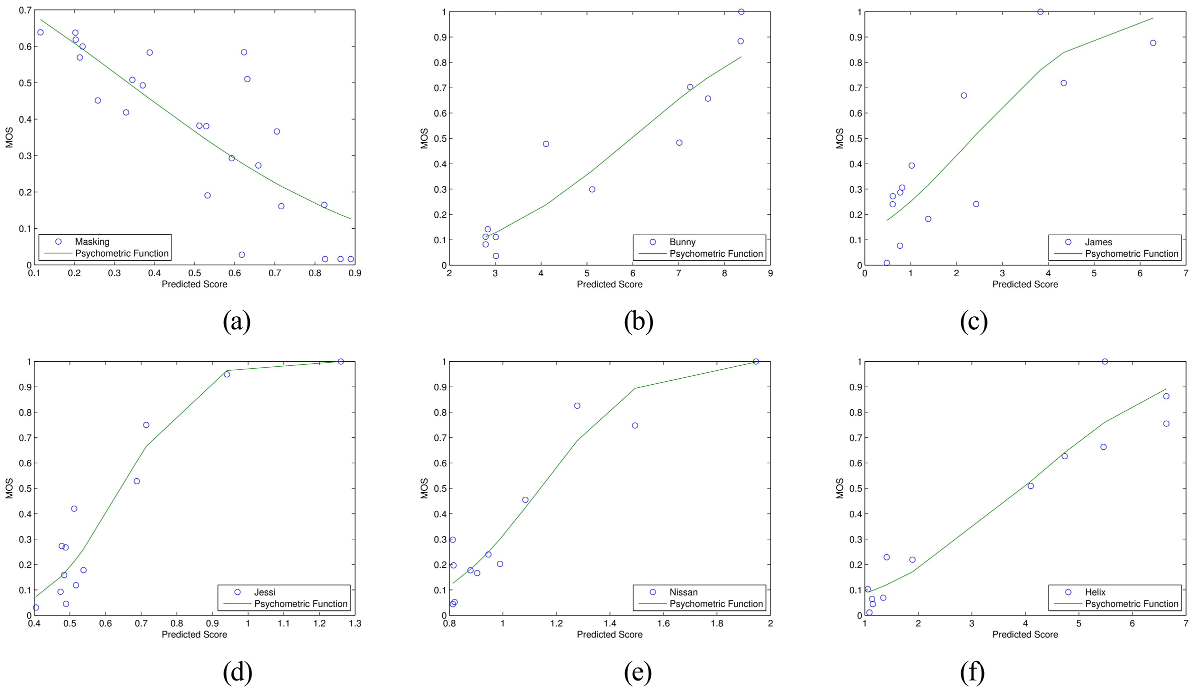
$$WLTPDM_{v_i} = w_i^p \cdot LTPDM_{v_i} \qquad (18)$$

where $w_i = s_i / \sum_{i=1}^{N} s_i$ with s_i one third of the total surface areas of all incident faces of v_i. p is the parameter for regulating the weight of the surface area and is empirically set as $p = -5$.

The testing performances for UWB compression database and LIRIS masking database are reported in Table 4. For the UWB compression database, since the scores of distorted meshes for different reference meshes are uncorrelated, we evaluate the generalization performance on distorted meshes of each reference mesh and calculate the average performance. Our method generally achieves high PLCC and SROCC performances on the UWB compression database and the LIRIS masking database. Note that our method does not involve MOSs of testing meshes in the training stage. The cross-database comparison result in Table 4 is encouraging and it indicates our method has good generality and robustness. The differences between various subjective databases lie in multiple factors, including the coverage of different distortion types, the coverage of models with different geometry complexity, different scoring protocols, and so on. We believe that the construction of richer databases with a wide coverage of various factors will alleviate such differences in the future, and our machine learning based method will show its strong learning capability with a large number of ground-truth data. Fig. 8 illustrates the scatter plots of MOSs versus predicted scores for meshes in the cross-database experiments. The MOSs of all testing meshes are normalized. Fig. 8 (a) illustrates the scatter plots for testing on the LIRIS masking database while Fig. 8 (b)-(f) illustrate the scatter plots for testing each model on the UWB compression database. From Fig. 8, we can see the psychometric curves generally fit the Prediction-MOS pairs well.

Table 4 Cross-Database Validation Performance

Performance	UWB Compression Database (%)						LIRIS Masking Database (%)
	Bunny	James	Jessi	Nissan	Helix	Average	
PLCC	94.6	83.8	90.6	92.7	95.9	91.5	83.5
SROCC	88.1	76.4	79.1	74.1	91.8	81.9	83.6



(a) (b) (c)

(d) (e) (f)

Fig. 8 Scatter plots of MOSs versus predicted scores for meshes in LIRIS masking database and UWB compression database. (a) LIRIS masking database. (b) Bunny model. (c) James model. (d) Jessi model. (e) Nissan model. (f) Helix model in the UWB compression database.

6.7 Time performance analysis

6.7.1 Time complexity

Our method mainly involves the following computational tasks: computing local perceptual distortions, extracting statistical descriptors, SVR training and testing. Our method firstly computes a local perceptual distortion for each vertex pair of two meshes under comparison, the time complexity of computing local perceptual distortions is linearly dependent on the number of vertices in the reference mesh. The time complexity of computing local distortions is $O(n)$, where n is the number of vertices in the reference mesh. After generating local perceptual distortions, our method extracts statistical descriptors from the local distortion sequence. We adopt the sort routine of Matlab, which achieves sequence sorting with Quick Sort algorithm, to sort the local distortion sequence. The time complexity of sorting the local distortion sequence is $O(n{\cdot}\log_2 n)$. We then compute seven elements of statistical descriptors from the sorted local distortion sequence. The time complexity of computing seven elements is $O(n)$. Thus, the overall time complexity of extracting statistical descriptors is $O(n{\cdot}\log_2 n)$. After extracting statistical descriptors, we pair the statistical descriptors and MOS of each distorted mesh into a data sample. At the training stage, the SVR model is trained with the training dataset. We use the Radius Basis Function kernel as the kernel function of SVR model. Let n_{sample} denote the number of training samples and d denote the dimension of feature vector, the time complexity of SVR training is $O(n_{sample}^2 \cdot d)$. At the testing stage, the quality of distorted meshes in the testing dataset are predicted with the trained SVR model. Let n_{sv} denote the number of support vectors that are generated during the training procedure, the time complexity of predicting the quality of a distorted mesh is $O(n_{sv} \cdot d)$.

6.7.2 Running time performance

We use the LIRIS/EPFL general-purpose database to demonstrate the running time performance of our method. All experiments are conducted on a desktop computer with 3.1 GHz i5-2400 CPU and 4 GB RAM. In the LIRIS/EPFL general-purpose database, there are four reference meshes: Armadillo, Dinosaur, RockerArm and Venus. The vertex numbers of these four reference meshes are in the same order of magnitude, and they are respectively 40002, 42146, 40177, and 49666.

Table 5 Average Running Time Performance of Main Computational Tasks

Computational Tasks	Running Time (ms)
Computing local perceptual distortions	865.31
Extracting statistical descriptors	1.93
SVR training	1.16
SVR testing	0.14

In Table 5, we report the average running time performance of main computational tasks of our method, which include computing local perceptual distortions, extracting statistical descriptors, SVR training and SVR testing. The procedure of computing local perceptual distortions mainly involves two steps: vertex pair matching between two meshes, and computing the local perceptual distortion for each vertex pair based on curvature tensor. We record the processing time for each distorted mesh and report the average processing time for all distorted meshes in the database. The average time of computing local perceptual distortions for all distorted meshes in the database is 865.31 millisecond. For each distorted mesh in the database, we extract statistical descriptors from the local distortion sequence. The average time of extracting statistical descriptors for all meshes in the database is 1.93 millisecond. We randomly choose 80% data in the database as the training dataset and use the remaining 20% data as the testing dataset. We repeat the procedure 50 times and report the average time performance respectively for training and testing. The average time for training the SVR model is 1.16 millisecond while the average time for testing with the trained SVR model is 0.14 millisecond. Since both the sample size of training dataset and the dimension of feature vector are small, as expected, the time for SVR training is little. The time for SVR testing is much less than the time for SVR training.

From the reported running time performance, we observe that, in terms of time consumption, the dominant part of our method is computing local perceptual distortions. Compared with computing local perceptual distortions, the time consumed with extracting statistical descriptors, SVR training and SVR testing is almost negligible. Overall, our method has good running time performance and the objective quality of a distorted mesh is evaluated in a short time.

6.8 Analysis of suitable distortions

In this paper, we use the LIRIS/EPFL general-purpose database as the training dataset. In this database, two types of distortions are included in the distorted meshes: noise addition and smoothing. As shown in Subsection 6.2, our method achieves good prediction performance when 80% of data is used as training set and the other 20% is used as testing set. This indicates our method works well for both noise addition and smoothing. Since editing (e.g., watermarking) and compression are both noise-like distortions, and simplification is a smoothing-like distortion which has a low-pass effect on the mesh, our method will work well for distortions that come from editing, compression and simplification of meshes. In literature [1], Lavoue et al. gave an analysis on the distortions that the LIRIS/EPFL general-purpose database includes. They concluded that noise addition and smoothing basically reflect a lot of possible distortions that occur in common mesh processing operations, including compression, watermarking, and simplification. In summary, our method can work with distortions of 3D mesh which come from editing, compression and simplification.

Our method is a full-reference metric, which means the reference mesh should be completely available. But in the case of 3D mesh reconstruction, the original underlying mesh (the reference mesh) is usually unknown and only the reconstructed mesh is available. So our method is not applicable to distortions that result from 3D mesh reconstruction. But one of our future works is to develop a no-reference metric for mesh visual quality assessment. Since no-reference metrics need no information about the reference mesh, no-reference metrics are expected to work for distortions that come from 3D mesh reconstruction.

7 Conclusions

In this paper, we proposed an objective MVQ metric using statistical descriptors of distortion distribution based on spatial pooling strategy. Our analysis indicates the overall mesh quality is sensitive to the distortion distribution. The statistical descriptors are consistent with subjective judgement of HVS, but how individual feature in the statistical descriptors interacts with each other remains an open question. The experiments on the LIRIS/EPFL General-Purpose Database indicate our method achieves better performance than state-of-the-art metrics. Our method achieves acceptable performance even with a small amount of training data. The inter-model prediction experiments show that our method has good inter-model generalization capability, and the cross-database experimental results are encouraging. Our method will show its strong learning capability when richer databases are available in the near future. One of our future works is to build a subjective database with a wide coverage of various factors, including different distortion types and models with different geometry complexity.

Acknowledgements This work is supported by the National Natural Science Foundation of China (No. 61373084) and Innovation Program of Shanghai Municipal Education Commission (No. 14YZ011).

References

1. Lavoue G., Drelie Gelasca E., Dupont F., Baskurt A., Ebrahimi T, “Perceptually driven 3D distance metrics with application to watermarking,” In Proceedings of SPIE Electronic Imaging, 2006，pp. 63120L.1-63120L.12.
2. Lavoue G, “A multiscale metric for 3D mesh visual quality assessment,” Computer Graphics Forum, vol. 30, no. 5, 2011, pp. 1427-1437.
3. L. Vasa and J. Rus, “Dihedral angle mesh error: a fast perception correlated distortion measure for fixed connectivity triangle meshes,” Computer Graphics Forum, vol. 31, no. 5, 2012, pp. 1715-1724.
4. K. Wang, F. Torkhani, and A. Montanvert, “A fast roughness-based approach to the assessment of 3D mesh visual quality,” Computer & Graphics, vol. 36, no. 7, 2012, pp. 808-818.
5. F. Torkhani, K. Wang, and J.-M. Chassery, “A curvature-tensor-based perceptual quality metric for 3D triangular meshes,” Machine Graphics Vision, vol. 23, no.1/2, 2014, pp. 59-82.
6. Lu Dong, Yuming Fang, Weisi Lin, and Hock Soon Seah, “Perceptual Quality Assessment for 3D Triangle Mesh Based on Curvature,” IEEE Trans. Multimedia, vol. 17, no. 12, 2015, pp. 2171-2184.
7. Wang Z., Bovik A. C., Sheikh H. R., Simoncelli E. P, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no.4, 2004, pp. 1-14.
8. Zhang L., Zhang D., Xuanqin M, “FSIM: A feature similarity index for image quality assessment,” IEEE Trans. Image Process., vol. 20, no. 8, 2011, pp. 2378-2386.
9. Xue W, Zhang L, Mou X, Bovik A C, “Gradient magnitude similarity deviation: A highly efficient perceptual image quality index,” IEEE Trans. Image Process., vol. 23, no. 2, 2014, pp. 684-695.
10. Li Q, Fang Y-M, Xu J-T, “A novel spatial pooling strategy for image quality assessment,” Journal of Computer Science and Technology, vol. 31, no. 2, 2016, pp. 225-234.
11. Guillaume Lavoué and Rafał Mantiuk, “Quality assessment in computer graphics, Visual signal quality assessment”, Springer International Publishing, 2015, pp.243-286.
12. Gastaldo P, Zunino R, Redi J, “Supporting visual quality assessment with machine learning,” EURASIP Journal on

Image and Video Processing, vol. 2013, no. 54, 2013, pp. 1-15.

13. Narwaria M, Lin W, “Objective image quality assessment based on support vector regression,” IEEE Transactions on Neural Networks, vol. 21, no. 3, 2010, pp. 515-519.

14. Narwaria M, Lin W, “SVD-based quality metric for image and video using machine learning,” IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 42, no. 2, 2012, pp. 347-364.

15. Hines A, Kendrick P, Barri A, et al., “Robustness and prediction accuracy of Machine Learning for objective visual quality assessment,” European Signal Processing Conference (EUSIPCO), 2014, pp. 2130-2134.

16. Gastaldo P, Redi J A, “Machine learning solutions for objective visual quality assessment,” 6th International Workshop on Video Processing and Quality Metrics for Consumer Electronics, vol. 12, 2012, pp.1-6.

17. Xu L, Lin W, Kuo C C J, “Visual quality assessment by machine learning,” SpringerBriefs in Signal Processing, Springer Singapore, 2015.

18. Lavoué G, Cheng I, Basu A, “Perceptual quality metrics for 3d meshes: towards an optimal multi-attribute computational model,” 2013 IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 3271-3276.

19. Z. Wang and A. C. Bovik, “Modern image quality assessment,” Synthesis Lectures on Image, Video, and Multimedia Processing, vol. 2, no. 1, 2006, pp. 1-156.

20. Z. Wang and A. C. Bovik, “Reduced- and no-reference image quality assessment,” IEEE Signal Processing Magazine, vol. 28, no. 6, 2011, pp. 29-40.

21. G. Lavoué and M. Corsini, “A comparison of perceptually-based metrics for objective evaluation of geometry processing,” IEEE Trans. Multimedia, vol. 12, no. 7, 2010, pp. 636-649.

22. M. Corsini, M.C. Larabi, G. Lavoue, O. Petrik, L. Vasa, and K. Wang, “Perceptual metrics for static and dynamic triangle meshes,” Computer Graphics Forum, vol. 32, no. 1, 2013, pp.101-125.

23. B.-E. Rogowitz and H.-E. Rushmeier, “Are image quality metrics adequate to evaluate the quality of geometric objects?,” In Proc. of Human Vision and Electronic Imaging, 2001, pp. 340-348.

24. Z. Karni and C. Gotsman, “Spectral compression of mesh geometry,” 27th annual conference on Computer graphics and interactive techniques, 2000, pp. 279-286.

25. Sorkine O, Cohen-Or D and Toledo S, “High-pass quantization for mesh encoding,” Symposium on Geometry Processing, 2003, pp. 42-51.

26. Corsini M., Drelie Gelasca E., Ebrahimi T., Barni M, “Watermarked 3-D mesh quality assessment,” IEEE Trans. Multimedia, vol. 9, no. 2, 2007, pp. 247-256.

27. Bian Z., Hu S.-M., Martin R. R, “Evaluation for small visual difference between conforming meshes on strain field,” Journal of Computer Science and Technology, vol. 24, no. 1, 2009, pp. 65-75.

28. Tian D., Alregib G, “FQM: A fast quality measure for efficient transmission of textured 3D models,” 12th annual ACM international conference on Multimedia, 2004, pp. 684-691.

29. Pan Y., Cheng I., Basu A, “Quality metric for approximating subjective evaluation of 3-D objects,” IEEE Trans. Multimedia, vol. 7, no. 2, 2005, pp. 269-279.

30. C.-C. Chang and C.-J. Lin, “LIBSVM: a library for support vector machines,” ACM Transactions on Intelligent Systems and Technology, vol. 2, no. 3, 2011, Article no. 27, pp. 1-27.

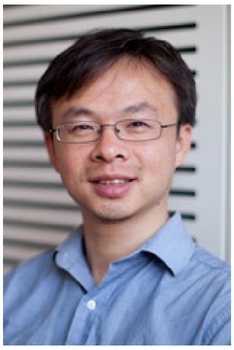
31. G. Lavoué, “A local roughness measure for 3D meshes and its application to visual masking,” ACM Transactions on Applied perception, vol. 5, no. 4, 2009, Article no.23, pp.1-21.

32. Peter G. Engeldrum, “Psychometric Scaling: A Toolkit for Imaging Systems Development,” Imcotek Press, April, 2000.

Xiang Feng is a PhD student of School of Communication and Information Engineering, Shanghai University. He is also a member of Institute of Smart City, Shanghai University. He received his BE Degree at School of Communication and Information Engineering, Shanghai University in 2011. Since 2011 he has been doing Master and PhD at School of Communication and Information Engineering, Shanghai University. He is a Dual Doctoral Degree student between Shanghai University and University of Technology, Sydney. He was awarded the CSC Scholarship (China Scholarship

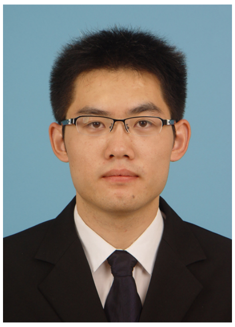
Council) to study at University of Technology, Sydney between August 2014 and September 2015. His research interests include 3D modelling and reconstruction, 3D deformation and animation, and perceptual quality assessment.

Wanggen Wan is a professor of School of Communication and Information Engineering, Shanghai University. He received his PhD Degree from Xidian University, China in 1992. He was Postdoctoral Research Fellow at Information and Control Engineering Dept. of Xi’an Jiao-Tong University in 1993-1995. From 1995 to 1998, he was working at Electronic and Information Engineering Dept. of Shanghai University. He was Visiting Scholar at Electrical and Electronic Engineering Dept. of HKUST in 1998-1999. He was Visiting Professor and Section Head at Multimedia Innovation Center of HKPU in 2000-2004. Since 2004, he has been a professor with School of Communication and Information Engineering, Shanghai University. His research interests include computer graphics, signal processing and data mining.



Richard Yi Da Xu is currently a Senior Lecturer with the School of Computing and Communications, University of Technology, Sydney (UTS), Australia. He received the B.Eng. degree in computer engineering from University of New South Wales, Sydney, NSW, Australia, in 2001, and the Ph.D. degree in computer sciences from University of Technology, Sydney, Chippendale, NSW, Australia, in 2006. His current research interests include machine learning, computer vision, and statistical data mining.

Haoyu Chen is a postdoc of School of Communication and Information Engineering, Shanghai University. He is also a member of Institute of Smart City, Shanghai University. He received his BE Degree at School of Communication and Information Engineering, Shanghai University in 2009. He received his PhD Degree School of Information System, University of Southern Queensland in 2016. He was awarded the CSC Scholarship (China Scholarship Council) to study as a PhD student at University of Southern Queensland between October 2009 and November 2015. His research interests include perceptual quality assessment, audio signal processing.



Pengfei Li is a master student of School of Communication and Information Engineering, Shanghai University. He received his BE Degree at School of Information and Electrical Engineering, Harbin Institute of Technology (Weihai) in 2014. His research interests include 3D data compression and transmission, perceptual quality assessment.

J. Alfredo Sánchez is a professor of computer science and director of the Laboratory of Interactive and Cooperative Technologies at Universidad de las Américas Puebla (UDLAP), Mexico. He received his MSc and PhD degrees in Computer Science from Texas A&M University, USA in 1993 and 1996, respectively. He chairs the Latin American Community of Human-Computer Interaction under ACM's SIGCHI, and is co-founder of the Latin American Conference Series on Human-Computer Interaction (CLIHC). He also has served as president of the Mexican Computer Science Society and is a member of the National

Researchers System. His research interests lie in the areas of human-computer interaction, natural user interfaces and information visualization.