Research on Perceptual Quality Metrics of 3D Meshes

Xiang Feng

Faculty of Engineering and Information Technology

University of Technology Sydney

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

December 2018

Certificate of Original Authorship

I certify that this thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been fully acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis. This research is supported by the Australian Government Research Training Program.

This thesis is the result of a research candidature conducted jointly with Shanghai University as part of a collaborative doctoral degree.

Production Note:
Signature removed prior to publication.

Signature of Candidate:__________

Date: January 23, 2019

Acknowledgements

I would like to express my sincere thanks to those who have supported me and contributed to this thesis during my PhD study.

I would like to thank my principal supervisor, Prof. Richard Xu, for his valuable guidance and insightful discussions during my PhD study. He provided me a great opportunity to broaden my research view and gave me lots of suggestions on my research topic. His passion for academic knowledge always motivates me to move forward towards my research goals. This thesis could not have been completed without his professional advice and great help. Besides, I wish to show great gratitude to my co-supervisor Prof. Stuart Perry for his valuable suggestions on my research works and continuous encouragement on my research progress. Particularly, he gave me lots of advice on the revision of our submitted papers. I also would like to thank Prof. Wanggen Wan, who is my PhD supervisor in Shanghai University, for his valuable guidance and great support.

I am grateful to my labmates in Faculty of Engineering and Information Technology, University of Technology Sydney. Great thanks also go to my friends and collaborators. We worked together and helped each other during the PhD study. They are: Junyu Xuan, Minqi Li, Jiatong Li, Tian Ding, Zhichao Sheng, Shan Xue, Sheng Wang, Maoying Qiao, Wenjie Zha, Bin Yang, Peibo Duan, Haodong Chang, Kai Yang.

I would like to express my appreciation for the financial support that I received during my PhD study. Special thanks go to China Scholarship Council and University of Technology Sydney.

Finally, I would like to thank my family members: my parents and brother. It is their constant love and great support that made all these possible ultimately.

I would like to dedicate this thesis to my parents for their love and support

Abstract

In mesh processing operations, the 3D mesh is always subject to geometric distortions which may degrade the visual quality of the 3D mesh. As the ultimate receptor of 3D meshes is typically human eyes, it is of great significance to develop perceptual metrics for mesh quality assessment. While most existing research works in mesh quality assessment focused on the evaluation of local distortions on the mesh, we mainly study the perceptual importance of local regions to the mesh quality in this thesis. We propose five perceptual mesh quality metrics and validate the performance superiority of these metrics over state-of-the-art metrics in the experiments.

Firstly, we investigate the relationship between the statistical descriptors of local distortions and the mesh quality. We propose the Tensor-based Perceptual Distance Measure-Spatial Pooling metric by exploiting the Support Vector Regression method. The experimental results demonstrate that the severely distorted regions on the mesh have a great impact on the mesh quality.

Secondly, we propose the percentile weighting method to emphasize the impact of the severely distorted regions on the mesh quality. A local distortion-based visual importance model is built by assigning a great weight to a small portion of vertices that suffer from severe distortions. We propose the Tensor-based Perceptual Distance Measure-Percentile Weighting metric by applying the percentile weighting method at the stage of spatial pooling.

Thirdly, we investigate the incorporation of mesh saliency into mesh quality assessment and build a visual importance model to emphasize the impact of salient regions on the mesh quality. We propose the Tensor-based Perceptual Distance Measure-Visual

Saliency metric by taking the saliency value as the weighting coefficient of the local distortion. A saliency map synthesis method is proposed to assemble the salient regions from different individual saliency maps. The experimental results reveal that the performance gain of the synthetic saliency map over individual saliency maps depends on the similarity between individual saliency maps.

Lastly, we propose the saliency weighting strategy to emphasize the impact of both severely distorted regions and salient regions on the mesh quality. We propose the Tensor-based Perceptual Distance Measure-Saliency Weighting metric and the Tensor-based Perceptual Distance Measure-Percentile Weighting-Saliency Weighting metric by applying the Minkowski method and the percentile weighting method respectively to the saliency-aware distortions. The experimental results reveal that the emphasis on the impact of salient regions improves the performance of the metric, but an overemphasis may cause a performance degradation.

Keywords — mesh quality assessment, perceptual importance, statistical descriptors, Support Vector Regression, percentile weighting method, spatial pooling, mesh saliency, saliency weighting strategy

Table of Contents

List of Figures

List of Tables

Abbreviations

CNN	Convolutional neural network
DAME	Dihedral angle mesh error
ECD	Empirical cumulative distribution
FMPD	Fast mesh perceptual distance
GL	Geometric Laplacian
HVS	Human Visual System
IVQ	Image visual quality
JND	Just noticeable difference
MACM	Multi-attribute computational model
MOS	Mean opinion score
MS	Mesh saliency
MSDM	Mesh structural distortion measure
MSDM2	Multiscale mesh structural distortion measure
MSLCE	Mesh saliency via local curvature entropy
MSSP	Mesh saliency spectral processing
MVQ	Mesh visual quality
PLCC	Pearson linear correlation coefficient
QS	Quality score
RBF	Radius basis function
RMS	Root mean square
SF	Strain field
SMQI	Saliency-based mesh quality index
SROCC	Spearman rank-order correlation coefficient

SSIM	Structural similarity index
SVR	Support vector regression
TPDM	Tensor-based perceptual distance measure
TPDMPW	Tensor-based perceptual distance measure-percentile weighting
TPDMPWSW	Tensor-based perceptual distance measure-percentile weighting-saliency weighting
TPDMSP	Tensor-based perceptual distance measure-spatial pooling
TPDMSW	Tensor-based perceptual distance measure-saliency weighting
TPDMVS	Tensor-based perceptual distance measure-visual saliency
VQEG	Video Quality Experts Group

Chapter 1

Introduction

1.1 Background

With the rapid development of 3D modeling technologies, 3D models and their applications have received extensive attention. As the most common representation of 3D models, a large number of 3D meshes have been created and used widely in various applications such as computer-aided design, digital entertainment and medical imaging. In various mesh processing operations such as mesh compression, mesh smoothing and mesh watermarking, it is common that the original 3D meshes are corrupted with geometric distortions, which may degrade the visual quality of the original mesh. As the ultimate receptor of 3D meshes is typically human eyes, it is very important to assess the visual quality of 3D meshes. A straightforward way to assess the visual quality of 3D meshes is subjective scoring. However, subjective scoring is usually time-consuming and has a certain randomness, which makes it unsuitable for most practical applications. Thus, it is urgent to develop mesh visual quality metrics to evaluate the perceptual quality of 3D meshes efficiently. In literature, mesh visual quality (MVQ) metrics are typically classified into three categories: full-reference metrics, reduced-reference metrics and no-reference metrics. Full-reference metrics require the original mesh to be fully available for the assessment, reduced-reference metrics require partial information of the original mesh, and no-reference metrics require no information of the original mesh. In this thesis, we focus on the full-reference MVQ metrics.

Until now, some well-performing full-reference MVQ metrics have been proposed, such as Mesh Structural Distortion Measure (MSDM) [1], Multiscale Mesh Structural Distortion Measure (MSDM2) [2], Dihedral Angle Mesh Error (DAME) [3], Fast Mesh Perceptual Distance(FPDM) [4], Tensor-based Perceptual Distance Measure (TPDM) [5], Dong [6]. These metrics basically follow a similar two-step computation process: firstly constructing a local quality map between the reference mesh and the distorted mesh by calculating the vertex-wise or edge-wise quality values, and then aggregating the local quality values into a single score that reflects the overall quality of the distorted mesh with a spatial pooling method. Most research works [1–6] focused on the evaluation of the local quality map, and adopted either the average method [3, 4, 6] or the Minkowski method [1, 2, 5] at the stage of spatial pooling. However, investigations into the spatial pooling method for the MVQ metric are fewer. To our knowledge, there is not a clear consensus yet on which method best suits the pooling stage in various MVQ metrics. The pooling method was usually selected in an empirical manner in respective algorithm design. Even the parameters and weighting coefficients of each pooling method were also determined empirically. The average pooling method assumes that all the local distortions on the mesh have the same contributions to the overall quality score of the mesh. There is an obvious problem with this method. When a part of the mesh is corrupted with high perceptual distortions while the rest is uncorrupted or slightly corrupted, the average pooling method would overestimate the quality of the distorted mesh. The underlying reason is that human eyes typically pay more attention to the regions with severe distortions than the areas with slight or no distortions, especially in the task of mesh quality assessment. The Minkowski pooling method was adopted mainly because it puts more emphasis on the severely distorted regions as the value of the Minkowski exponent increases. The

Minkowski exponent was typically determined in an ad-hoc way. The Minkowski pooling method is intuitively closer to the subjective judgment than the average pooling method. The Minkowski pooling method follows the assumption that the local distortions at different locations are statistically independent, which lacks sufficient theoretical basis in the human visual system.

In the research field of image quality assessment, full-reference image visual quality (IVQ) metrics evaluate the overall quality of the images by following a similar two-step process as the full-reference MVQ metrics. However, compared with the research works on the image quality, the research works on the mesh quality are still in their infancy. Some research works on the MVQ metrics [1] were inspired by the research works on IVQ metrics [7]. Li et al. [8] proposed a spatial pooling method for IVQ metrics by exploiting machine learning methods. They extracted statistical descriptors from the local quality map to represent the overall quality of the image, and demonstrated that the overall quality of the image is sensitive to the distribution of the local quality map. We believe that there are some similarities between MVQ metrics and IVQ metrics since both of them are expected to align with visual perception. The studies in literature [1, 7] demonstrated the similarity in the perceptual characteristics to some degree between image quality assessment and mesh quality assessment. The scholars in the community of image quality assessment have conducted many studies on IVQ metrics [9–15] by exploiting machine learning methods while most scholars in the community of mesh quality assessment investigated MVQ metrics [1–6] by constructing the visual perception model explicitly for the mesh quality. Lavoué et al. [16] proposed the Multi-Attribute Computational Model (MACM) metric by using the machine learning techniques. The MACM metric does not take the perceptual characteristics of the human visual system into account, and the cross-database generalization capability of the MACM metric was not evaluated in literature [16]. We believe that machine learning techniques can play an increasingly important role in MVQ metrics, particularly when a large subjective database for mesh quality assessment is available in the future. Several spatial pooling strategies have been proposed for the IVQ metrics by putting more emphasis on the low quality regions of the image [17]. Wang et al. [18] proposed three spatial pooling strategies for the IVQ metrics, and demonstrated that the local information content-weighted pooling strategy achieves the best performance for the metric among three spatial pooling strategies. Later, Wang et al. [17] proposed an advanced statistical model to estimate the local information content that is used to weight the quality map. Moorthy et al. [19] proposed the percentile weighting method for the spatial pooling by assigning a greater weight to the low quality regions of the image. In this thesis, we investigate the relationship between the statistical characteristics of local distortions and the overall quality of the mesh, and propose a mesh visual quality metric [20] by exploiting machine learning methods. We also investigate the percentile weighting strategy for spatial pooling in the mesh visual quality metric, and propose a mesh visual quality metric [21] with the percentile weighting method.

As another research topic related to mesh quality, mesh saliency detection [22] has also attracted much attention in the field of visual perception. Many computational saliency methods [23–27] have been proposed to detect the perceptually important regions where human visual attention is focused on the mesh. Since the receptor of both mesh visual quality and mesh saliency is the human visual system, we believe that it is possible to improve the performance of MVQ metrics by incorporating mesh saliency into the metric. Actually, in the research field of image quality assessment, there are already some research works [19, 28–31] that investigated the incorporation of either visual attention or computational image saliency into IVQ metrics. Zhang et al. [32] presented a statistical evaluation to analyze the added value of integrating the image saliency map into IVQ metrics. They concluded that, when integrating the image saliency map into IVQ metrics, the computational saliency models yield a performance gain for the IVQ metrics statistically though the specific amount of the performance gain depends on the combination of the computational saliency model and the IVQ metric [32]. In literature [19, 28–32], either the visual fixation map or the image saliency map was incorporated in the IVQ metric to improve the performance of the metric based on the assumption that the distortions occurring in more salient regions of the image are more visible and thus have a greater impact on the image quality, which was finally verified by the experimental results. Compared with the works in the image quality assessment, there are much fewer works that investigated the relationship between mesh saliency and mesh visual quality, not to mention the incorporation of the mesh saliency map into MVQ metrics. Since the ultimate assessor of both the mesh visual quality and the image visual quality is the human visual system, we similarly assume that the distortions occurring in more salient regions of the mesh have a greater impact on the mesh quality. Based on this assumption, in this thesis, we investigate the incorporation of the mesh saliency map into the MVQ metric and propose three mesh visual quality metrics, one of which was introduced in literature [33].

As mentioned in literature [22], many computational methods have been proposed to detect mesh saliency. However, the question is which computational method should be selected to generate the mesh saliency map that will be incorporated into the MVQ metric. Kim et al. [34] carried out a user study with an eye-tracking experiment and evaluated the correlation between the mesh saliency map generated by the method [23] and the fixation map acquired from the eye-tracking experiment. Chen et al. [35] built a benchmark with pseudo-ground truth mesh saliency maps based on the Schelling points [36], and employed a regression model to predict the saliency map of 3D meshes with the benchmark. Tasse et al. [37] proposed three metrics to evaluate the performance of three 3D computational saliency methods [24, 38, 39] based on the benchmark [35]. However, there is a lack of comprehensive quantitative analysis to reveal the accuracy and reliability of state-of-the-art mesh saliency detection methods. In literature [23–27], the effectiveness of the mesh saliency detection method was justified mostly through either the application-guided evaluation [23–25] or the subjective visual analysis [26, 27]. As the research works in literature [23–25] demonstrated that the three mesh saliency detection methods proposed in literature [23–25] were capable of enhancing the results of graphics applications, such as mesh simplification and viewpoint selection, we employ them [23–25] to generate the saliency maps for the meshes in this thesis.

1.2 Research Questions and Objectives

This research aims to investigate the perceptual importance of local regions to the mesh quality and will answer the following research questions:

Question 1. *Do the local distortions on the mesh contribute equally to the overall quality of the mesh ?*

Before choosing an appropriate method to aggregate the local distortions into the quality score of the mesh at the stage of spatial pooling, one typically needs to consider whether the local distortions contribute equally to the overall quality of the mesh or not.

Question 2. *Do the severely distorted regions have a greater impact on the mesh quality than the moderately or slightly distorted regions?*

If the contributions of the local distortions to the mesh quality are not equal, do the severely distorted regions have a greater impact on the mesh quality than the moderately or slightly distorted regions ? If so, how could the severely distorted regions be distinguished from the other regions ?

Question 3. *Do the salient regions have a greater impact on the mesh quality than the non-salient regions ?*

Do the local distortions occurring in the salient regions have a greater impact on the mesh quality than the local distortions with the same strength occurring in the non-salient regions ? If so, how could the impact of the salient regions on the mesh quality be emphasized in the mesh visual quality metric ?

Question 4. How do the severely distorted regions and the salient regions jointly influence the mesh quality ?

If both the severely distorted regions and the salient regions have a great impact on the mesh quality, how do they jointly influence the mesh quality ? How could the contributions of both the severely distorted regions and the contributions of the salient regions to the mesh quality be combined in the metric ?

This research will achieve the following objectives, which are expected to answer the aforementioned research questions:

Objective 1. To explore the relationship between the statistical characteristics of the local distortions and the overall quality of the mesh.

This thesis will investigate how local distortions are linked to the overall quality of the mesh from a statistical point of view. Statistical characteristics will be extracted to characterize the statistical distribution of the local distortions. The machine learning methodology will be exploited to find the relationship between the statistical characteristics of the local distortions and the overall quality of the mesh. This objective corresponds to *Question 1*.

Objective 2. To develop a visual importance model based on the local distortions.

This thesis will develop a visual importance model based on the local distortions to emphasize the impact of the severely distorted regions on the mesh quality. The vertices with large local distortions will be assigned a greater weight at the stage of spatial pooling to highlight their contributions to the quality score. The effectiveness of the visual importance model will be evaluated through the experiment. This objective corresponds to *Question 2*.

Objective 3. *To develop a visual importance model based on the mesh saliency.*

This thesis will develop a visual importance model based on the mesh saliency to emphasize the impact of salient regions on the mesh quality. The contributions of two vertices with the same intensity of local distortions to the quality score will be distinguished based on their saliency values. The effectiveness of the visual importance model will be evaluated through the experiment. This objective corresponds to *Question 3*.

Objective 4. *To develop a method to combine the local distortion and saliency value of the vertex.*

This thesis will develop a method to combine the local distortion and saliency value of the vertex, and investigate the joint impact of the severely distorted regions and the salient regions on the mesh quality. The contributions of the vertices to the quality score will be determined based on both the local distortions and the saliency values of the vertices. This objective corresponds to *Question 4*.

1.3 Research Contributions

The main contributions of this thesis are summarized as follows:

- We have proposed a spatial pooling method based on machine learning and proposed the TPDMSP metric by applying this method to generate the quality score. The support vector regression model is used to learn a function that maps from the statistical descriptors of the local distortions to the quality score of the mesh. The experiments demonstrate that the severely distorted regions have a significant impact on the mesh quality.

- We have proposed the percentile weighting method for the mesh visual quality metric by emphasizing the impact of the severely distorted regions on the mesh quality. The TPDMPW metric is proposed by applying the percentile weighting method to aggregate the local distortions at the stage of spatial pooling.
- We have investigated the incorporation of the mesh saliency map into the mesh visual quality metric, and built a saliency-based visual importance model to emphasize the impact of the salient regions on the mesh quality. The TPDMVS metric is proposed by applying the visual importance model in the Minkowski method.
- We have proposed the TPDMSW metric and the TPDMPWSW metric by applying the Minkowski method and the percentile weighting method respectively to the saliency-aware distortions. These two metrics emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality. The experiments demonstrate that an emphasis on the impact of the salient regions contributes to improving the performance of the metric, but an overemphasis on the impact of salient regions may cause a performance degradation to the metric.
- We have investigated the influence of the local surface area of the vertex on the mesh quality with the TPDMPW metric and the TPDMVS metric. The experiments demonstrate that it is inappropriate to include the local surface area in the metric on the LIRIS/EPFL general-purpose database. This finding may facilitate the investigation on how the local surface area influences the overall quality of distorted meshes with various types of distortions.

1.4 Research Significance

When the 3D mesh goes through processing operations such as mesh compression, mesh smoothing and mesh watermarking, the 3D mesh is subject to geometric distortions which may degrade the visual quality of the original 3D mesh. Since subjective scoring is time-consuming and unsuitable for practical applications, it is critical to design perceptual metrics to evaluate the visual quality of the 3D mesh. This thesis investigates the perceptual importance of the local regions to the mesh quality and proposes five mesh visual quality metrics. The research outcomes and the mesh visual quality metrics proposed in this thesis would play an important role in a variety of applications related to 3D mesh processing. Typical applications include:

- **Evaluation and optimization of mesh processing algorithms**: The proposed metrics, such as TPDMVS and TPDMPWSW, can be used as the benchmark to compare the performance of different mesh processing algorithms. While geometric measures, such as the Root Mean Square (RMS) error and Hausdorff Distance (HD), have been widely used to evaluate the performance of mesh processing algorithms, our proposed metrics can be employed to evaluate the performance of mesh processing algorithms in a perceptual way and would lead to a more accurate comparison result. The research outcomes in the thesis can be used to optimize the algorithm design for mesh processing operations, such as mesh smoothing, mesh simplification and mesh watermarking, to achieve the optimal performance with the least perceptual distortions. For example, in mesh processing operations, more attention should be paid to the visually important regions such as the salient regions. In this way, the introduced distortions on the visually important regions would be minimized and the visual degradation of mesh quality would also be minimized.

- **Medical simulation and training**: In the applications of medical simulation and training, soft tissues and organs are typically modeled with 3D simulation technologies. A realistic simulation system facilitates doctors to learn and train for the surgery procedure repeatedly in order to develop a reasonable and feasible surgical plan. The simulation system will provide a good basis for actual medical diagnosis and improve the effectiveness and quality of surgeries dramatically. The study on mesh quality metrics in this thesis will lay the foundation for the design of a realistic medical simulation system. Taking the facial plastic surgery as an example, it is critical to consider the visual difference between the ideal face shape and the achieved face shape after the surgery in the design of the surgical plan. The mesh visual quality metric can be used to evaluate the visual difference between the ideal face mesh and the achieved face mesh. By including the visual perception in the design of the medical simulation system, the medical simulation system can not only reduce the cost and loss in the surgery, but also contributes to improving the surgical effect.

- **Remote visualization and interaction**: In the applications of remote visualization and interaction, such as virtual museum and digital cultural heritage, high-resolution 3D meshes are typically stored on remote servers that have a high storage capacity and powerful computing capabilities. When the user browses the 3D meshes online at the receiver ends or client terminals, the 3D meshes stored in the remote servers are typically compressed and then transmitted over the network to mobile terminals. The network condition will determine the compression rate, and the visual quality of the received meshes at the receiver ends or client terminals may decline due to the compression operation and possible noise induced during the transmission process. The mesh visual quality metrics proposed in this thesis can be used to evaluate the perceptual quality of 3D meshes efficiently. The quality scores of 3D meshes can be used as feedback for the content and service providers to evaluate the performance of network transmission in order to optimize the quality of user experience.

1.5 Thesis Structure

While most existing mesh quality metrics are full-reference metrics, this thesis also investigates full-reference mesh quality metrics. The investigation on full-reference MVQ metrics will not only facilitate the understanding of perceptual characteristics of the human visual system, but also lay the foundation for the investigation of no-reference MVQ metrics in the future. Most full-reference MVQ metrics consist of two steps: firstly, computing the local quality map (either local distortion map or local similarity map) between the reference mesh and the distorted mesh; then, accumulating the local quality values with a spatial pooling method to generate the quality score for the distorted mesh. Though many scholars have conducted some excellent research works on the evaluation of the local quality map, the research works on the spatial pooling method are much fewer. In this thesis, we focus on the spatial pooling methods for the metric and investigate the perceptual importance of local regions on the mesh to the mesh quality. We generated the local distortions between the reference mesh and the distorted mesh with the Tensor-based Perceptual Distance Measure (TPDM) metric. The main research contents of this thesis include: mesh quality prediction based on statistical descriptors, spatial pooling with the percentile weighting strategy, mesh quality assessment by incorporating mesh saliency, and mesh visual quality metric based on the saliency-aware distortion.

The organizational structure of the thesis is illustrated in Fig. 1.1. As illustrated in Fig. 1.1, the thesis consists of seven chapters, and the main research contents are described in Chapter 3, Chapter 4, Chapter 5, and Chapter 6. In the community of mesh quality assessment, several spatial pooling methods have been used to aggregate the local distortions into the overall quality score. In order to explore the relationship between the local distortions and the mesh quality, in Chapter 3, we apply the learning-based strategy to learn the function that maps from the statistical descriptors of local distortions to the overall quality score with the mesh samples in the subjective databases. The exper-

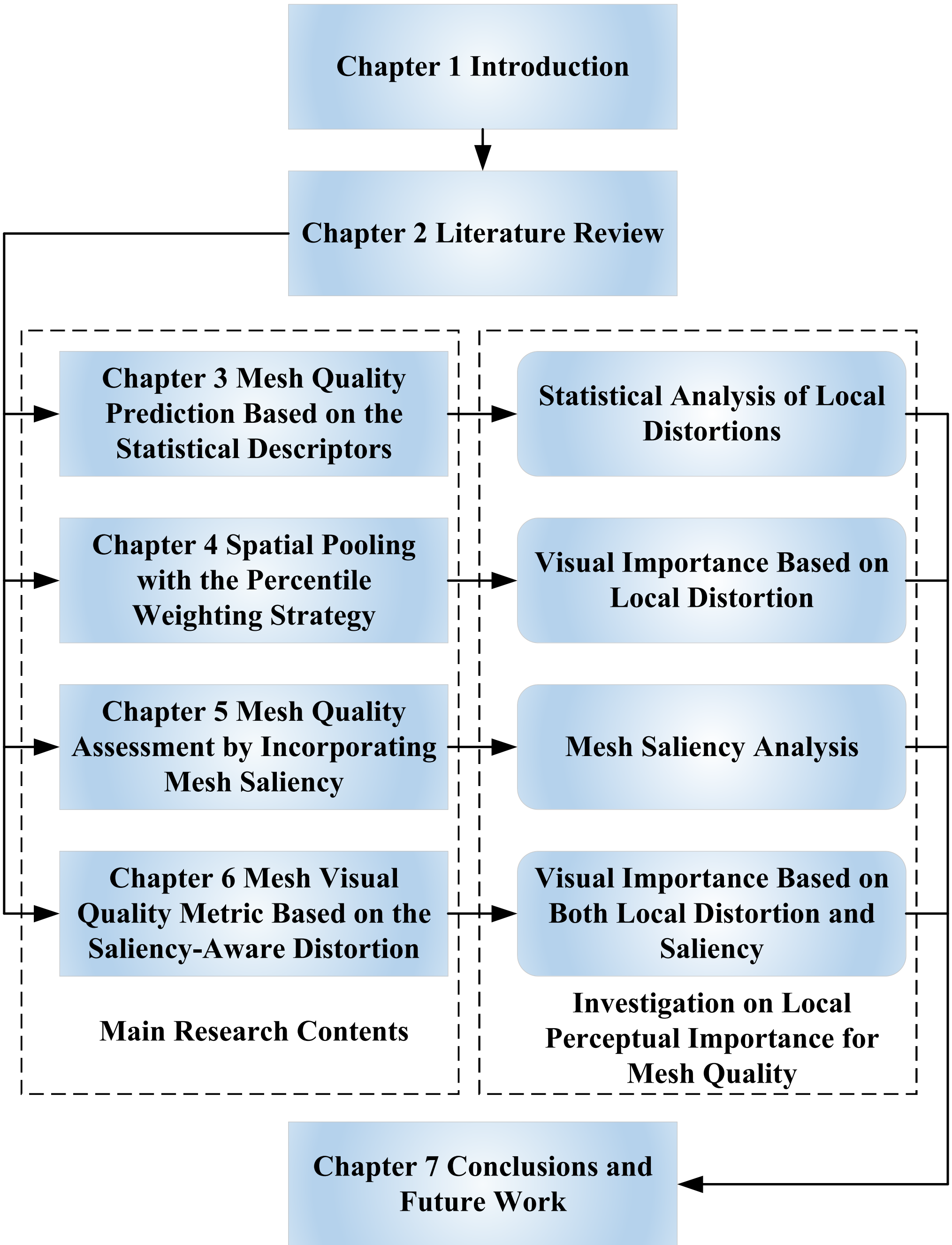


Figure 1.1 : The organizational structure of this thesis.

iments in Chapter 3 demonstrate that the severely distorted regions on the mesh have a great impact on the mesh quality. Based on the conclusions in Chapter 3, we investigate emphasizing the impact of the severely distorted regions on the mesh quality and apply the explicit design-based strategy to develop the mesh visual quality metric explicitly in Chapter 4. We also apply the explicit design-based strategy to investigate the incorporation of mesh saliency map into mesh visual quality metrics in Chapter 5 and Chapter 6. The explicit formulas facilitate the understanding of how the local distortion vector and the mesh saliency map are related to the mesh quality. The major difference between two aforementioned strategies is that the learning-based strategy follows the machine learning paradigm while the explicit design-based strategy allows us to express the mesh visual quality metrics explicitly.

In Chapter 1, we introduce the background, the research questions and objectives, the research contributions, and the research significance of this thesis.

In Chapter 2, we review the related work and literature on the mesh quality assessment. Firstly, we introduce the subjective databases for mesh quality assessment. Then, we review state-of-the-art works on mesh visual quality metrics, image visual quality metrics and mesh saliency detection methods. Finally, we introduce the performance evaluation methods for mesh visual quality metrics.

In Chapter 3, we investigate the relationship between the statistical distribution of the local distortions and the overall quality of the distorted mesh, and propose the Tensor-based Perceptual Distance Measure-Spatial Pooling (TPDMSP) metric. The TPDMSP metric firstly computes the local distances between the corresponding vertices of the reference mesh and the distorted mesh as the local distortions, and then extracts the statistical descriptors from the local distortions as a feature vector which represents the overall quality of the distorted mesh. The pair of the feature vector and the mean opinion score (MOS) of the distorted mesh constitutes a mesh sample. We build the training dataset and the testing dataset with the mesh samples in the subjective databases. The training dataset is fed to the support vector regression (SVR) model to learn a function that maps from the feature vector to the quality score of the distorted mesh. The trained SVR model is used to predict the quality scores of the distorted meshes in the testing dataset. We demonstrate the performance superiority of the TPDMSP metric over state-of-the-art metrics, and also the inter-model generalization capability and cross-database generalization capability of the TPDMSP metric on three databases. The TPDMSP metric shows good performance for mesh quality prediction even with a small amount of training samples. The experimental results demonstrate that the statistical descriptors of the local distortions reflect the mesh quality well, and the severely distorted regions have a great impact on the mesh quality.

In Chapter 4, we propose a spatial pooling method with the percentile weighting strategy to emphasize the impact of the severely distorted regions on the mesh quality. We build a visual importance model based on local distortion and assign a greater weight to the vertices with large local distortions at the stage of spatial pooling. We weight the local distortion by the local surface area and investigate the influence of the local surface area on the mesh quality. We propose the Tensor-based Perceptual Distance Measure-Percentile Weighting (TPDMPW) metric by applying the percentile weighting method to accumulate the local distortions. We investigate the influence of the parameters of the percentile weighting method on the performance of the TPDMPW metric, and determine the optimal values and unified values for the parameters through empirical tests on three publicly available databases. The experimental results demonstrate the effectiveness of the percentile weighting method and the superiority of the TPDMPW metric over state-of-the-art metrics. The research content in this chapter is an extension and development of the research content in Chapter 3.

In Chapter 5, we investigate the incorporation of the mesh saliency map into the mesh visual quality metric. We assign the saliency value as the weighting coefficient of the local distortion for each vertex at the stage of spatial pooling to emphasize the impact of the salient regions on the mesh quality. We compute the local distances between the vertices of the reference mesh and the distorted mesh as the local distortions with the TPDM metric, and generate the saliency maps for the reference meshes with three well-known mesh saliency detection methods. We build a visual importance model based on mesh saliency and propose the Tensor-based Perceptual Distance Measure-Visual Saliency (TPDMVS) metric by applying the visual importance model in the Minkowski method. A saliency map synthesis method is proposed by assembling the salient regions of different individual saliency maps. We evaluate the performance gain of the synthetic saliency map over the individual saliency maps for the TPDMVS metric, and investigate the dependence of the performance gain on the similarity between the individual saliency maps. The experimental results demonstrate the effectiveness of this visual importance model and the superiority of the TPDMVS metric over state-of-the-art metrics. The saliency-based visual importance model achieves better performance for the metric than the local distortion-based visual importance model which is proposed in Chapter 4.

In Chapter 6, we propose the saliency weighting strategy to combine the contributions of the severely distorted regions and the salient regions to the mesh quality. We generate a saliency-aware distortion for each vertex through weighting the local distortion by the exponential result of the saliency value. We propose the Tensor-based Perceptual Distance Measure-Saliency Weighting (TPDMSW) metric and the Tensor-based Perceptual Distance Measure-Percentile Weighting-Saliency Weighting (TPDMPWSW) metric by applying the Minkowski method and the percentile weighting method respectively to the saliency-aware distortions. These two metrics emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality. The TPDMSW metric is the generalized version of the TPDMVS metric proposed in Chapter 5. The experimental results demonstrate that an emphasis on the impact of the salient regions improves the performance of the metric, but an overemphasis on the impact of salient regions may cause a performance degradation to the metric. We suggest the appropriate range of value for the exponent in the saliency weighting strategy through the experimental tests. The experimental results demonstrate the effectiveness of the saliency weighting strategy and the performance superiority of the TPDMPWSW metric. We evaluate the performance gain of the synthetic saliency map over individual saliency maps for the TPDMPWSW metric, and demonstrate the dependency of the performance gain on the similarity between individual saliency maps.

In Chapter 7, we conclude the thesis and summarize the findings of this thesis. We point out our future work on the research topic and look to the prospects for the development of mesh visual quality metrics.

1.6 Publications Related to This Thesis

Below is a list of the refereed papers during my PhD research that have been published:

1. **Xiang Feng**, Wanggen Wan, Richard Yi Da Xu, Stuart Perry, Song Zhu, Zexin Liu. A new mesh visual quality metric using saliency weighting-based pooling strategy. Graphical Models, September 2018, vol. 99, pp. 1-12.

2. **Xiang Feng**, Wanggen Wan, Richard Yi Da Xu, Stuart Perry, Pengfei Li, Song Zhu. A novel spatial pooling method for 3D mesh quality assessment based on percentile weighting strategy. Computers & Graphics, August 2018, vol. 74, pp. 12-22.

3. **Xiang Feng**, Wanggen Wan, Richard Yi Da Xu, Haoyu Chen, Pengfei Li, J. Alfredo Sánchez. A perceptual quality metric for 3D triangle meshes based on spatial pooling. Frontiers of Computer Scicence, August 2018, vol. 12, no. 4, pp. 798–812.

4. A. A. M Muzahid, Wanggen Wan, **Xiang Feng**. Perceptual quality evaluation of 3D triangle mesh: a technical review. The 2018 International Conference on Audio, Language and Image Processing (ICALIP), IEEE, July 2018, Shanghai, China, pp.266-272.

5. **Xiang Feng**, Wanggen Wan, Xiaoqiang Zhu, Yanlu Yin, Jing Wang. A method for physics-based dynamic deformation with St. Venant Kirchhoff elasticity and

implicit Newmark integrator. Journal of Information and Computational Science, June 2015, vol. 12, no. 9, pp. 3333- 3343.

Chapter 2

Literature Review

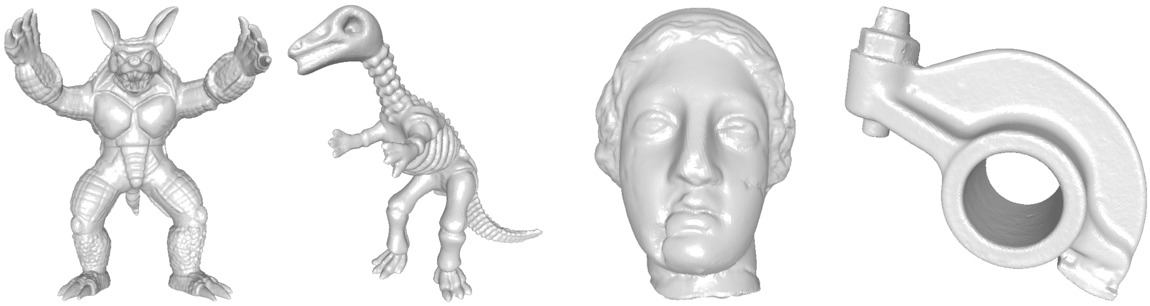
2.1 Subjective database for mesh quality assessment

The straightforward way to mesh quality assessment is subjective scoring, i.e, inviting some observers to rate the quality for the meshes according to the perceptual visual distortions of the meshes. This way has a certain randomness. Different people may have different ratings for the same mesh, and even the same person may have slightly different ratings for the same mesh when the mesh is viewed at different times. In order to arrive at an accurate and effective subjective score for the mesh, sufficient observers should be invited to participate in the subjective experiment and complete the rating in an appropriate environment. The process of subjective scoring is both tedious and time-consuming, which makes the subjective scoring impractical in most applications. Thus, it is necessary to develop objective metrics to produce a quality score for the mesh that reflect the subjective judgement of the mesh quality accurately. In order to evaluate the performance of mesh quality metrics, several publicly available subjective databases have been constructed as the benchmark for mesh quality metrics, such as, LIRIS/EPFL general-purpose database [1], LIRIS masking database [40] and UWB compression database [3]. The databases consist of the reference meshes and distorted meshes of several 3D models, and the subjective opinion scores that reflect the visual quality of the distorted meshes.

The distorted meshes were generated by applying artificial distortion operations to the reference meshes. The distortions in the distorted meshes reflect the visual degradation that occur frequently during the geometric processing of 3D meshes. The level of distortions of the distorted meshes varies in the range where the human eyes can recognize and distinguish the distortions. After generating the distorted meshes, several subjects participated in the subjective experiments by observing the distorted meshes projected on the computer screen. They rated the visual quality of the meshes according to the degree of perceptual distortions in the meshes. After collecting the subjective ratings for the meshes from all the observers, a statistical analysis was conducted on the ratings in order to evaluate the validity and reliability of the ratings. Finally, a mean opinion score was calculated as the subjective quality score for each distorted mesh.

2.1.1 LIRIS/EPFL general-purpose database

Lavoué et al. constructed the LIRIS/EPFL general-purpose database [1] in 2006. They used four 3D models: Armadillo, Dinosaur, Venus and RockerArm in the subjective experiment. The LIRIS/EPFL general-purpose database contains 4 reference meshes and 84 distorted meshes. There are 21 distorted meshes for each reference mesh. Two types of distortion including noise addition and smoothing were applied with different strengths at four locations (uniformly, smooth areas, rough areas, and intermediate areas) on the reference mesh. The noise addition refers to adding a random offset to the geometric coordinates of the vertex in the reference mesh to change the spatial position of the vertex. The intensity of the noise addition distortion was controlled by adjusting the value of the maximum offset. The smoothing refers to applying the Taubin smoothing filter to the vertices of the reference mesh to smooth the local regions on the surface. The intensity of the smoothing distortion was controlled by adjusting the number of iterations of the smoothing filter. These distortions reflect the geometric distortions that occur commonly in the mesh processing operations, such as, mesh simplification, mesh compression, and mesh watermarking. The reference meshes of four 3D models in the LIRIS/EPFL general-purpose database are illustrated in Fig. 2.1 and the geometric information of the reference meshes is listed in Table 2.1.



(a) Armadillo (b) Dinosaur (c) Venus (d) RockerArm

Figure 2.1 : The reference meshes in the LIRIS/EPFL general-purpose database.

Table 2.1 : The geometric information of the reference meshes in the LIRIS/EPFL general-purpose database.

Reference mesh	Number of vertices	Number of facets	Number of edges
Armadillo	40002	80000	120000
Dinosaur	42146	84288	126432
Venus	49666	99328	148992
RockerArm	40177	80354	120531

The process of generating the distorted meshes is as follows: The curvature tensor was estimated at each vertex of the reference mesh based on the normal cycle [41], and then the local roughness of each vertex was defined by the variance of the curvature. The surface of the reference mesh was divided into the smooth regions, the rough regions, and the intermediate regions according to the local roughness of each vertex. For the noise addition distortion, the random noise was added in the smooth regions, rough regions, intermediate regions and the whole surface respectively. There were three levels of added noise. Twelve distorted meshes with noise addition were generated for each reference mesh. For the smoothing distortion, the smoothing filter was applied to the rough regions, the intermediate regions, and the whole surface respectively. There were three levels of smoothing distortion. Nine distorted meshes with noise addition were generated for each reference mesh.

After the distorted meshes were generated, twelve students from Swiss Federal Institute of Technology and University Claude Bernard of Lyon participated in the subjective experiment as the subjects. They viewed the 3D meshes on the computer screen while sitting at a comfortable distance from the screen. Interaction (rotation, scaling, translation) was allowed so that the subjects could perceive the distortions in the mesh from different perspectives. Firstly, the subjects viewed the reference meshes, some distorted meshes and the distorted meshes with worst quality in the case of noise addition and smoothing in order to establish the referential range for each model. Then, the subject chose the distorted mesh with worse quality between the distorted mesh with worst quality for the noise addition and the distorted mesh with worst quality for the smoothing distortion. Finally, the subject viewed all the meshes iteratively and provided a score reflecting the degree of perceived distortion to each mesh. The score ranges from 0 (best quality) to 10 (worst quality), where 0 indicates that the mesh has the identical quality to the reference mesh while 10 indicates that the mesh has the worst quality. In order to avoid the influence of sequencing on the subjective scoring, the sequence of the meshes was randomly generated for each subject by the experimental platform.

Since the perception acuity and rating scale may vary among the subjects, the subjective scores of each subject were normalized after the subjective scores of all the subjects were collected. The mean and variance of the subjective scores for each subject were computed and then used to normalize the subjective score of each mesh. Finally, the mean opinion score of each mesh was calculated from the normalized subjective score:

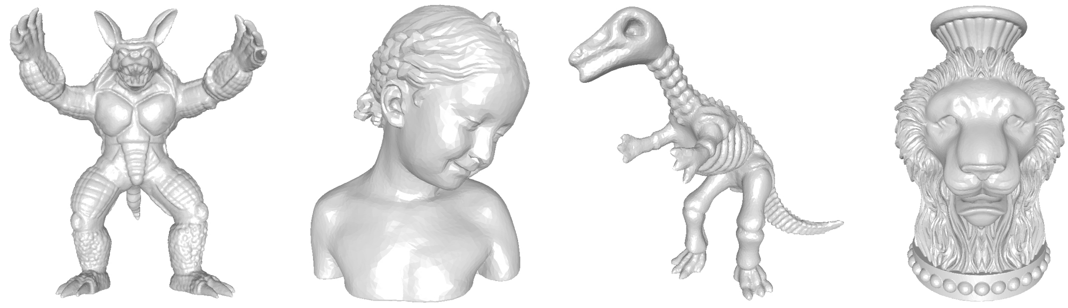
$$MOS_i = \frac{1}{n}\sum_{j=1}^{n} m_{ij}, \tag{2.1}$$

where MOS_i is the mean opinion score of the i^{th} mesh, n is the number of the subjects, and m_{ij} is the normalized score of the i^{th} mesh given by the j^{th} subject.

2.1.2 LIRIS masking database

Lavoué constructed the LIRIS masking database [40] in 2009. The database contains four 3D models: Armadillo, Bimba, Dinosaur, and Lion. The surface of the mesh was divided into rough regions and smooth regions according to the local roughness of the vertices in the mesh. The random noise was added to the vertices on the rough regions and smooth regions respectively for each reference mesh. There were three noise strengths: high, medium, and low. In literature [40], the database was used to demonstrate the visual masking effect of human visual system, that is, it is difficult for the human eyes to perceive the local distortions caused by the additive noise in the rough regions. The database consists of 28 meshes, i.e, 4 reference meshes and 24 distorted meshes. Each 3D model corresponds to one reference mesh and six distorted meshes. Among the distorted meshes of each 3D model, three distorted meshes were generated by adding noise with three different strengths to the rough regions of the reference mesh while the other three distorted meshes were generated by adding noise with three different strengths to the smooth regions of the reference mesh. The noises with the same strength applied to the rough regions and smooth regions have the same root mean square error. The reference meshes of four 3D models in the LIRIS masking database are illustrated in Fig. 2.2 and the geometric information of the reference meshes is listed in Table 2.2.

Eleven researchers from the LIRIS lab of INSA-Lyon participated in the subjective experimnent and rated the visual quality of the distorted meshes. During the subjective experiment, the subject viewed the reference mesh and six distorted meshes together for



(a) Armadillo (b) Bimba (c) Dinosaur (d) Lion

Figure 2.2 : The reference meshes of four 3D models in the LIRIS masking database.

Table 2.2 : The geometric information of the reference meshes in the LIRIS masking database.

Reference mesh	Number of vertices	Number of facets	Number of edges
Armadillo	40002	80000	120000
Bimba	8857	17710	26565
Dinosaur	20218	40432	60648
Lion	38728	77452	116178

each 3D model. The subject provided an opinion score to each distorted mesh that reflects the degree of the perceived similarity between the reference mesh and the distorted mesh. The score ranges from 0 to 4, where 4 indicates that the distorted mesh has the identical quality to the reference mesh while 0 indicates that the distorted mesh has the worst quality compared with the reference mesh. The interactions (rotation, scaling, translation) were allowed when the subject was observing the meshes. All the six distorted meshes of each 3D model were displayed simultaneously on the screen so that the subject could observe the six meshes at the same time. Thus, it is not necessary to establish the referential range for the meshes any more. In order to avoid the impact of the sequencing factor on the subjective experiment, the sequences of four models and six distorted meshes were generated randomly for each subject.

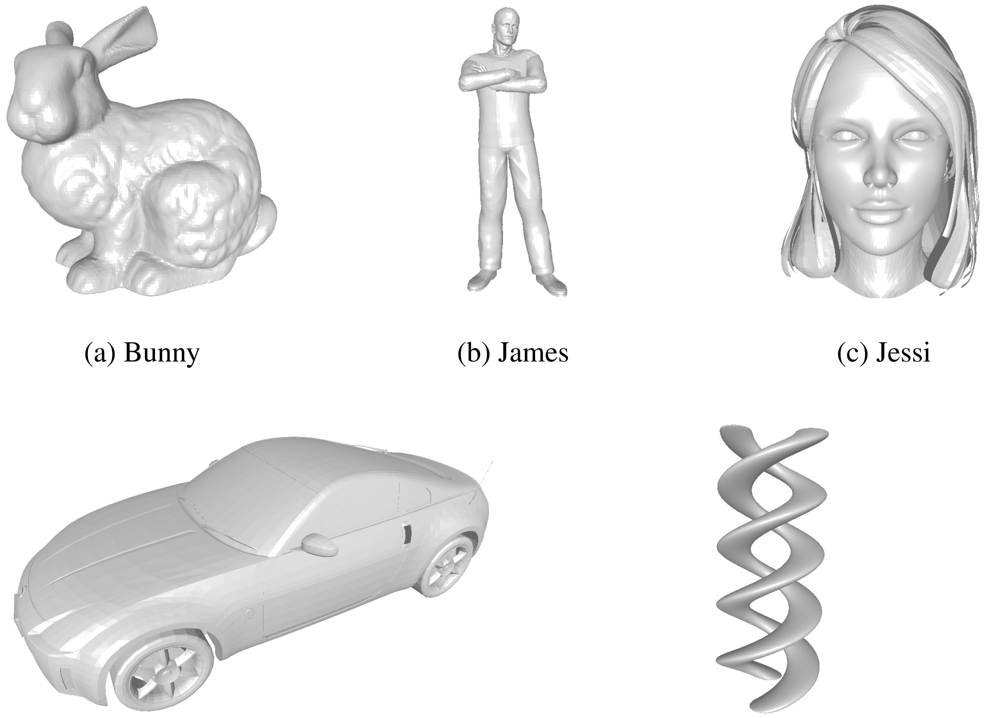
After collecting the opinion scores for the meshes, Lavoué conducted a one-way Analysis of Variance for the opinion scores in order to evaluate the statistical correlation between the scores of different subjects. The results of the statistical analysis in literature [40] showed that there is no significant variation between the subjective scores of different subjects. Finally, the mean value of all the subjective scores for each distorted mesh was calculated as the mean opinion score of the distorted mesh.

2.1.3 UWB compression database

Váša and Rus constructed the UWB compression database [3] in 2012. The database contains five 3D models: Bunny, James, Jessi, Nissan, and Helix. Each model represents a different object. The Bunny model is a model with a complex surface generated by 3D surface reconstruction methods. It is familiar to the subjects and commonly used in mesh processing operations. The James and Jessi models are the artificial models created in a modeling software, and represent a male body and a female face respectively. The Nissan model is an artificial model of a car in the computer aided design. The Helix model is an abstract shape with smooth surface. The reference meshes of five 3D models in the UWB compression database are illustrated in Figure 2.3 and the geometric information of the reference meshes is listed in Table 2.3.

Table 2.3 : The geometric information of the reference meshes in the UWB compression database.

Reference mesh	Number of vertices	Number of facets	Number of edges
Bunny	35946	71888	107832
James	40214	78029	118164
Jessi	70290	127956	198108
Nissan	16923	27547	44300
Helix	30534	61060	91590



(a) Bunny (b) James (c) Jessi

(d) Nissan (e) Helix

Figure 2.3 : The reference meshes of five 3D models in the UWB compression database.

In order to include different types of distortion in the experiment, the following 13 distortions are applied to the reference mesh:

- Adding random noise to the coordinates of vertices uniformly in three axises
- Adding random noise to the coordinates of vertices uniformly in each axis, and then scaling the noise in each axis by the proportion of the edges of the bounding box for the mesh
- Adding sine noise to the coordinates of vertices in the x axis
- Adding random noise with Gaussian distribution to the coordinates of vertices in each axis
- Uniform quantization of the coordinates of vertices

- Quantization of the coordinates of vertices, and then scaling the quantization errors in three axes by the proportion of the edges of the bounding box for the mesh
- Random affine transformation of the vertices
- Smoothing the vertices
- Adding the angle-preserving noise to the vertices
- Adding the normal-preserving noise to the vertices
- Applying the Angle-Analyzer mesh compression [42] to the mesh
- Applying the spectral mesh compression [43] to the mesh
- Applying the high-pass encoding mesh compression [44] to the mesh

Among the distortions above, the angle-preserving noise is a special type of distortion that preserves the dihedral angles of the mesh. The distortion is induced by adding random noise to the randomly selected vertices of the mesh while limiting the caused distortion of the incident dihedral angles to a predefined threshold. The desired intensity of distortion is achieved by repeating the process. Similarly, the normal-preserving noise is introduced by adding random noise to the randomly selected vertices of the mesh while limiting the distortion of normal directions of triangles to a predefined threshold. Finally, the UWB compression database [3] contains 5 reference meshes and 63 distorted meshes (12 or 13 distorted meshes for each reference mesh).

The subjective experiment was performed using the MeshTest software platform [3] and involved 69 subjects (63 males and 6 females), most of whom were students of computer science and engineering. The subjects viewed the reference mesh and two distorted meshes on the same monitor simultaneously. The subjects were asked to select the distorted mesh that was considered to have less distortion of the two distorted meshes. The reference mesh was also included in the test set as a hidden anchor mesh in order to determine the level of visually imperceptible distortion. The subjects were allowed to manipulate (translate, rotate, scale) the meshes synchronously on the software platform. The number of comparisons would be too large if each subject selected the preference from all the pairs of distorted meshes for each model. In order to reduce the number of comparisons, each subject was asked to select the preference from 40 pairs of distorted meshes, which were chosen randomly from different datasets. Let $p(i, j)$ denote the number of times that the distorted mesh i was selected as the less distorted one than the distorted mesh j, N denote the number of distorted meshes for each model, then the perceived quality score of the distorted mesh i was calculated as follows:

$$q(i) = \sum_{j=0}^{N} \frac{p(i, j)}{p(i, j) + p(j, i)}. \tag{2.2}$$

After the quality scores of all the meshes for each model were calculated, the quality score of each mesh was normalized and the normalized quality score was taken as the mean opinion score. The mean opinion score lies in the range [0, 1], where 0 indicates that the mesh has the best quality while 1 indicates that the mesh has the worst quality. The MOS of each distorted mesh reflects the quality ranking among all the distorted meshes of each model, but there is no coherence between the MOSs of distorted meshes for different models. Therefore, as suggested in literature [3, 5, 6, 45], the overall performance on the UWB compression database is evaluated by averaging the performances on five models.

2.2 Mesh visual quality metrics

2.2.1 Overview of Mesh Visual Quality Metrics

Until now, the mesh visual quality (MVQ) metrics assess the visual quality of 3D meshes mainly in the geometric space. The MVQ metrics evaluate the degree of distortions in the mesh caused by changing the geometric coordinates or topological connections of the vertices in the mesh. Several well-performing MVQ metrics have been proposed, and detailed reviews of the MVQ metrics can be found in literature [45–47]. The study in literature [45] shows that the classical geometric distances, such as Hausdorff distance (HD) and root mean square error (RMS) [48, 49], have weak correlation with the human visual perception.

According to the data source used for the quality evaluation, the MVQ metrics can be classified into two categories: image-based MVQ metrics and model-based MVQ metrics. The 3D mesh is typically displayed on the computer monitor when the human eyes are evaluating the visual quality of the 3D mesh. Human eyes perceive the visual quality of the 3D mesh through observing the 2D images generated by projecting the 3D mesh on various planes. Thus, an intuitive way to evaluate the mesh quality is using a virtual camera to take the snapshots from multiple viewpoints in the 3D scene and generating a projected 2D image at each viewpoint. The quality of the projected 2D image is evaluated by using the image quality metric. Finally, the quality score of the 3D mesh is calculated by pooling the quality scores of all the images. This type of metric is called an image-based MVQ metric. Since many excellent image quality metrics have been proposed, the image-based MVQ metrics can make use of existing image quality metrics to evaluate the visual quality of meshes. An alternative type of metric is the model-based MVQ metric [1–6], which evaluates the visual quality of 3D mesh by utilizing the geometric information of 3D model. The commonly used geometric information includes the local curvature of vertex, the local roughness of vertex, the dihedral angle. Since the model-based MVQ metrics reflect the perceptual characteristics of human visual system well, the model-based MVQ metrics generally achieve higher accuracy for mesh quality assessment than the image-based MVQ metrics.

Lindstrom and Turk [50] proposed a mesh simplification method by evaluating the cost of edge collapse based on the difference between the images of the original mesh and the images of the simplified mesh. Rogowitz and Rushmeier [51] studied the applicability of image-based MVQ metrics in the perceptual evaluation of simplified 3D meshes, and concluded that the image-based MVQ metrics [9, 52] are inadequate to evaluate the perceptual quality of 3D meshes. Cleju and Saupe [53] evaluated the performance of image-based MVQ metrics on the simplfied 3D meshes, and compared the performance of three image-based MVQ metrics with the performance of three classical geometric distances. The image-based MVQ metrics are the structural similarity index (SSIM), mean of absolute differences of the pixels (IMN), and root mean squared distance (IRMS) while the classical geometric distances were geometric root mean squared distance (MRSMS), mean geometric distance (MMN) and Hausdorff geometric distance (MHAUS). Their experimental results showed that the image-based MVQ metrics perform generally better than the geometric distances. Since the geometric distances used in literature [53] were demonstrated to correlate poorly with human visual perception [45, 46], the study in literature [53] could not reflect the performance superiority between the image-based MVQ metrics and the model-based MVQ metrics. Lavoué et al. [54] investigated the efficiency of IVQ metrics for evaluating the visual quality of 3D meshes, and conducted a statistical analysis of the performance of image-based MVQ metrics with seven IVQ metrics [17, 55–59], two rendering algorithms, four lighting conditions, and five pooling algorithms. They compared the image-based MVQ metrics with the model-based MVQ metrics in the performance of mesh quality assessment. They concluded that the image-based MVQ metrics achieve similar performance as the model-based MVQ metrics in the case of evaluating the quality of distorted meshes of the same model under the same type of distortion. However, the model-based MVQ metrics perform significantly better than the image-based MVQ metrics in the case of evaluating the quality of either the distorted meshes with different types of distortions or the distorted meshes of different models. When using the image-based MVQ metrics to evaluate the visual quality of 3D meshes, two issues need to be addressed: how many viewpoints should be designated to obtain enough 2D images, and how to select the appropriate viewpoints to acquire representative 2D images for the 3D mesh. Thus, it is difficult to apply the image-based MVQ metrics effectively in the practical applications of mesh quality assessment.

Several well-performing model-based MVQ metrics [1–6] have been proposed by utilizing the geometric information of 3D model. Compared with the classical geometric distances, the model-based MVQ metrics reflect the perceptual characteristics of human visual system, and thus show stronger correlation with the visual perception of mesh quality [45, 46]. Table 2.4 lists state-of-the-art model-based MVQ metrics, which are identified by the category of the metric, the feature used in the metric, whether the metric is suitable for the meshes with changed topological connectivity or not, and whether the metric exploits machine learning methods or not. Based on the availability of the reference mesh, the MVQ metrics can be classified into three categories: full-reference metric, reduced-reference metric, and no-reference metric. The full-reference metric requires the reference mesh to be fully available, the reduced-reference metric requires partial information of the reference mesh, and the no-reference metric requires no information of the reference mesh. In this thesis, we focus on the full-reference MVQ metrics, which indicates the reference mesh is fully available.

In the community of mesh quality assessment, Karni and Gotsman [43] proposed the first model-based MVQ metric. They combined the vertex coordinates and the geometric Laplacian of vertex coordinates to evaluate the distance between the distorted mesh and the reference mesh. Later, Sorkine et al. [44] improved the metric by assigning a greater weight to the geometric Laplacian values. Corsini et al. [60] proposed two metrics, 3DWPM1 and 3DWPM2, based on per-vertex difference in roughness and two formulations of roughness measure. Bian et al. [61] developed the Strain Field (SF) metric based on the strain energy that causes the deformation to the reference mesh. Lavoué et al. proposed the Mesh Structural Distortion Measure (MSDM) metric [1] by utilizing the concept of structural similarity that was introduced in the IVQ metric [7]. The MSDM metric firstly computes the differences of curvature statistics in local windows between the reference mesh and the distorted mesh, and then uses the Minkowski method

Table 2.4 : The model-based mesh visual quality metrics.

Metrics	Reference	Feature	Changed Connection	Machine Learning
RMS [48, 49]	Full	Surface distance	Yes	No
HD [48, 49]	Full	Surface distance	Yes	No
GL1 [43]	Full	Geometric Laplacian	No	No
GL2 [44]	Full	Geometric Laplacian	No	No
3DWPM1 [60]	Reduced	Global roughness	No	No
3DWPM2 [60]	Reduced	Global roughness	No	No
SF [61]	Full	Strain energy	No	No
MSDM [1]	Full	Curvature	No	No
MSDM2 [2]	Full	Multiscale curvature	Yes	No
DAME [3]	Full	Dihedral angle	No	No
FMPD [4]	Reduced	Global roughness	Yes	No
MACM [16]	Full	Curvature, dihedral angle	No	Yes
TPDM [5]	Full	Curvature tensor	Yes	No
Dong [6]	Full	Curvature	No	No
SMQI [62]	Full	Saliency	No	No
Abouelaziz [63]	No	Dihedral angle, statistics	Yes	Yes
TPDMSP [20]	Full	Curvature tensor, statistics	Yes	Yes
TPDMPW [21]	Full	Curvature tensor, visual importance	Yes	No
TPDMVS [33]	Full	Curvature tensor, saliency	Yes	No

to pool the local distances. This metric requires that the distorted mesh should have the same topological connectivity with the reference mesh. Lavoué et al. also introduced the LIRIS/EPFL general-purpose database [1] that was constructed for mesh quality assessment. Lavoué later extended the MSDM metric [1] to allow the comparison between two meshes with changed connectivities and proposed a multi-scale metric, Multiscale Mesh Structural Distortion Measure (MSDM2) [2]. Váša and Rus introduced the Dihedral Angle Mesh Error (DAME) metric [3], which firstly calculates the difference of oriented dihedral angles between two meshes as the local distortion and then pools the local distortions into the quality score with the weighted average method. Váša and Rus also introduced the UWB compression database in literature [3] which was constructed for mesh quality assessment. Wang et al. [4] computed the perceptual distance between two meshes based on the local roughness derived from the Gaussian curvature of the vertices, and proposed the Fast Mesh Perceptual Distance (FMPD) metric by evaluating the global roughness of the mesh. Torkhani et al. [5] proposed the Tensor-based Perceptual Distance Measure (TPDM) metric to evaluate the visual difference between two meshes based on the curvature tensors of the meshes. The perceptual distance between the reference mesh and distorted mesh is computed through the weighted Minkowski summation of local distances. The TPDM metric assigns the roughness coefficients as the weighting coefficients of the difference between the curvature tensors to model the visual masking effect of human visual system. Dong et al. [6] proposed a MVQ metric based on the structure similarity and roughness similarity between the reference mesh and the distorted mesh. Both the structure similarity and roughness similarity are evaluated with the average pooling method. This metric requires that the two meshes under comparison should have the same topological connectivity. Nouri et al. proposed the Saliency-based Mesh Quality Index (SMQI) metric [62] based on a multiscale visual saliency map. Compared with the research works on image visual quality [52], the research works on mesh visual quality [9] are considerably fewer. There are still many issues to be addressed in the mesh quality assessment [9], such as the spatial pooling method with better performance, no-reference MVQ metric and quality metric for dynamic meshes [45, 64–66]. In the community of image quality assessment, many studies have been conducted into the spatial pooling methods [8, 17–19] and IVQ metrics [10, 11, 57, 67].

Most existing full-reference MVQ metrics consist of a two-step computation process: firstly constructing a local quality (either similarity or distortion) map by comparing the local regions between the reference mesh and the distorted mesh, and then aggregating the local quality values with a spatial pooling method to generate the quality score for the distorted mesh. The scholars in the community have conducted some research works on the construction of the local quality map of the mesh, and proposed some well-performing MVQ metrics, such as MSDM [1], MSDM2 [2], DAME [3], FMPD [4], TPDM [5], and Dong [6]. However, the studies on the spatial pooling methods in the MVQ metric are considerably fewer. Most MVQ metrics applied either the average summation method [3, 4, 6] or the Minkowski summation method [1, 2, 5] at the stage of spatial pooling. Until now, there is not a clear consensus yet on which one of the spatial pooling methods is most suitable for various MVQ metrics. The spatial pooling method was usually selected in an empirical manner in the algorithm design, and even the values of the parameters of each method were also determined empirically. The average summation method [3, 4, 6] assumes that all the local distortions on the mesh have an equal contribution to the overall quality of the mesh. This method would overestimate the mesh quality when only a part of the mesh is severely corrupted while the rest is uncorrupted or slightly corrupted. The reason is that human eyes typically pay more attention to the regions with severe distortions than the regions with slight or no distortions, especially in the task of mesh quality assessment. The regions with severe local distortions may have a great impact on the overall quality of the mesh. The Minkowski summation method [1, 2, 5] is intuitively closer to human visual judgment than the average summation method because the Minkowski summation method is capable of emphasizing the severely distorted regions for the mesh quality. The Minkowski exponent was determined typically in an ad-hoc way.

2.2.2 Spatial Pooling Method for the Visual Quality Metric

In the community of image quality assessment, full-reference IVQ metrics [7, 57, 67] follow a similar two-step procedure with MVQ metrics: firstly computing the local distortions or similarities between the reference image and the distorted image, and then pooling the local distortions or similarities into the quality score for the distorted image. Li et al. [8] proposed a spatial pooling method for the IVQ metrics through statistical analysis of the local quality map of the image. They extracted the statistical descriptors from the local quality map to represent the overall quality of the image, and demonstrated that the overall quality of the image is sensitive to the statistical descriptors of the local quality map. Since the visual quality of both 3D mesh and 2D image takes the subjective judgment of human visual system as the standard, there are certain similarities between mesh quality assessment and image quality assessment. The theories and methodologies in the IVQ metrics may provide some inspirations and heuristics for the investigation of MVQ metrics. The research works in literature [1, 7] indicated that there are some similarities in the structural information between mesh quality and image quality. While several IVQ metrics [9–15] have been proposed by exploiting the machine learning methods, most MVQ metrics [1–6] attempt to model the perceptual characteristics of human visual system explicitly.

Recently, some scholars have gradually explored using machine learning methods to develop MVQ metrics. Lavoué et al. [16] proposed the multi-attribute computational model (MACM) metric for mesh quality assessment based on a linear combination of several relevant geometric attributes, and determined the optimized weight coefficients through a linear regression for the combination function. However, the authors [16] did not present the cross-database generalization capability of the MACM metric, and the

MACM metric does not reflect the perceptual characteristics of human visual system. Abouelaziz et al. [68] proposed a no-reference MVQ metric using the convolutional neural network (CNN). They generated a set of patches from the local curvatures and dihedral angles for each distorted mesh and then trained the CNN network with the patches and the MOSs of the distorted meshes. Feng et al. [20] extracted the statistical descriptors from the local distortions as the feature vector of the mesh, and then used the support vector regression (SVR) model to learn the function that maps from the feature vector to the quality score of the mesh. They proposed the Tensor-based Perceptual Distance Measure-Spatial Pooling (TPDMSP) metric based on the statistical descriptors of the local distortions. Abouelaziz et al. [63] proposed a no-reference MVQ metric based on the statistical characteristics of dihedral angles and the SVR model. The dihedral angles of the mesh were computed as the feature to represent the structural information and were modulated to reflect the visual masking effect. Three statistical distributions were used to construct three sets of feature vectors, and then the SVR model was used to learn the mapping function from the feature vectors to the quality scores of the distorted meshes [63]. As the research on mesh quality assessment progresses and deepens, machine learning methods may play an increasingly important role in the investigation of MVQ metrics, especially when a large subjective mesh database that involves a great number of mesh samples is available in the future. We will investigate the MVQ metric by exploiting the meahcine learning method in Chapter 3.

Several spatial pooling strategies have been proposed for the IVQ metrics by putting more emphasis on the low quality regions of the image [17]. Wang and Shang [18] investigated three spatial pooling methods for the IVQ metrics, and demonstrated that the information content-weighted pooling method achieved the best performance among the three spatial pooling methods. Later, Wang and Li [17] proposed a multiscale information content weighting method based on the Gaussian scale mixture of natural images at the stage of spatial pooling. Moorthy and Bovik [19] introduced two spatial pooling strategies, the visual fixation-based weighting and the quality-based weighting, based on the visual importance for image quality assessment. They proposed the percentile scoring method by assigning a greater weight to the low quality image patches. Li et al. [8] proposed a spatial pooling method for the IVQ metrics through statistical analysis of the quality map of the image, and demonstrated that the overall quality of the image is sensitive to the distribution of the quality map. Inspired by the research work [19] in image quality assessment, Feng et al. [21] proposed a spatial pooling method for mesh quality assessment based on the percentile weighting strategy and developed the Tensor-based Perceptual Distance Measure-Percentile Weighting (TPDMPW) metric. The TPDMPW metric [21] emphasizes the severely distorted regions for the mesh quality by assigning a greater weight to a portion of vertices with large local distortions. We will investigate the percentile weighting method in the mesh quality assessment in Chapter 4.

2.2.3 Incorporation of Visual Saliency into the Visual Quality Metric

As another important research field related to visual perception, mesh saliency [22] has also attracted much attention in the community. Many computational saliency methods [22–27, 69–73] have been proposed to detect the perceptually important regions where human visual attention is focused on the mesh. Some mesh saliency detection methods [74–78] have been utilized to optimize the mesh processing algorithms. Liu et al. [22] reviewed state-of-the-art mesh saliency detection methods and their applications. They classified the mesh saliency detection methods into two categories: local contrast-based methods and global contrast-based methods, and described the advantages and drawbacks of the mesh saliency detection methods in literature [22]. Lee et al. [23] developed a mesh saliency detection method by applying the center-surround operator to the Gaussian-weighted mean curvatures. Song et al. [24] proposed a method for detecting mesh saliency by analyzing the properties of the log-Laplacian spectrum of the mesh. Limper et al. [25] proposed a mesh saliency detection method, named local curvature entropy, by applying Shannon entropy to the mean curvature of vertices of the mesh. The effectiveness of the three methods in literature [23–25] has been demonstrated in graphics applications, such as mesh simplification, mesh segmentation, and viewpoint selection. Nouri et al. [26] proposed a local surface descriptor based on the adapative patches to characterize the perceptual saliency of each vertex of the mesh. Tao et al. [27] detected mesh saliency via manifold ranking in a descriptor space that is composed of the patch descriptors based on Zernike coefficients. Kim et al. [34] conducted an user study experiment on five 3D meshes with an eye tracker, and recorded the human eye fixations on the 3D meshes. They evaluated the performance of the mesh saliency detection method proposed in literature [23] by computing the correlation between the human eye fixations and the saliency maps. Chen et al. [35] designed an online experiment to investigate the Schelling points on the 3D meshes. The participants in the experiment selected the points on the 3D meshes that they thought would be selected by other people. The selected points are called Schelling points on the 3D meshes. They generated a saliency map for each 3D mesh from the acquired Schelling points, and finally constructed a database that consists of 3D meshes with pseudo-ground truth saliency maps [35]. Wang et al. [79] conducted an eye-tracking experiment on the 3D printed objects, and produced a dataset of 3D meshes with fixation maps. They collected the gaze data from the human observers with a monocular pupil eye tracker [80], and extracted the fixations on the surface of the objects from the pupil positions. This was the first work on the study of visual attention on the 3D objects. Lavoué et al. [81] conducted an eye-tracking experiment on the rendered 3D shapes, and produced a benchmark with the fixation density maps for 3D meshes by mapping the human eye fixations onto the 3D shapes. They investigated the impact of shape, camera position, material and illumination on visual attention based on the acquired human eye fixations. Tasse et al. [37] proposed three metrics to quantitatively evaluate the performance of three mesh saliency detection methods [24, 38, 39] based on the mesh saliency database [35].

Since the receptors of both mesh visual quality and mesh saliency are human eyes, we believe that it is possible to improve the performance of MVQ metrics by incorporating mesh saliency. Actually, in the community of image quality assessment, many scholars [19, 28–32, 82] have investigated the correlation between image quality and either visual attention or computational image saliency by incorporating visual attention or computational image saliency into the image quality assessment. The visual attention data or the image saliency map was used to optimize the design of IVQ metrics at the stage of spatial pooling. These research works were basically based on the assumption that the distortions occurring on more salient regions of an image are more visible and thus have a greater impact on the image quality.

Moorthy et al. [19] proposed to weight the local quality by the visual fixation and improved the performance of the IVQ metric. Liu and Heynderickx [28] included visual attention in the design of IVQ metrics based on the eye-tracking data and achieved performance gain with the modified metrics. Farias and Akamine [29] concluded that the performance gain depends on the precision of computational saliency method and the distortion type when the saliency map is incorporated into the IVQ metrics. Liu et al. [30] investigated the influence of image content on the performance gain when including the visual attention in image quality assessment. Zhang et al. [31] used the visual saliency as a feature to compute the local quality map for the distorted image and employed the visual saliency as the weighting coefficient to emphasize the impact of perceptually important regions on the image quality. Zhang et al. [32] presented a statistical evaluation of the added value of incorporating the computational image saliency into the IVQ metric. They concluded that the computational saliency methods yield a performance gain statistically when integrating the computational saliency method in the IVQ metric though the amount of performance gain depends on the specific combination of the computational saliency method and the IVQ metric.

Compared with the research works on image quality assessment, the research works on mesh quality assessment are considerably fewer. There are even fewer studies that investigated the relationship between mesh saliency and mesh quality, not to mention the incorporation of mesh saliency into the MVQ metric. Nouri et al. [62] proposed the Saliency-based Mesh Quality Index (SMQI) metric by using a multiscale saliency map to compute the local statistics that reflect the structural information of the mesh. The study in literature [62] demonstrated that there exists a link between mesh quality and mesh saliency. The investigations on the relationship between the image quality and either visual attention or computational visual saliency [19, 28–32, 82] were based on the assumption that the local distortions appearing in the more salient regions of an image have a greater impact on the overall quality of the image. Since both mesh quality and image quality take the subjective judgment of human visual system as the standard, some relevant findings in the image quality assessment may inspire the research in the mesh quality assessment. Feng et al. [33] proposed a similar assumption in the mesh quality assessment as the aforementioned assumption made in the image quality assessment, that is, the local distortions occurring in the more salient regions of a mesh have a greater impact on the overall quality of the mesh. Based on this assumption, they proposed the Tensor-based Perceptual Distance Measure-Visual Saliency (TPDMVS) metric [33] by incorporating the mesh saliency map into the MVQ metric. Since the three mesh saliency detection methods proposed in literature [23–25] were demonstrated to be capable of enhancing the performance of graphics applications, they employed these three methods [23–25] to generate the saliency maps for the meshes [33]. We will investigate the incorporation of the mesh saliency map into the MVQ metric in Chapter 5 and Chapter 6.

2.3 Performance evaluation of mesh visual quality metrics

In the research field of mesh quality assessment, the performance of the MVQ metrics [1–6] is evaluated typically with two coefficients: Pearson linear correlation coefficient (PLCC) and Spearman rank-order correlation coefficient (SROCC). Specifically, the MVQ metrics are firstly used to compute the quality scores for the distorted meshes in the subjective database. The performance of the MVQ metric is then evaluated by calculating the PLCC and SROCC coefficients between the MOSs and the computed quality scores of the distorted meshes. The PLCC coefficient measures the prediction accuracy of the metric while the SROCC coefficient measures the prediction monotonicity [52, 83]. Both values of PLCC and SROCC lie within the range [-1, 1], where 1 indicates fully positive correlation, -1 indicates fully negative correlation, and 0 indicates no correlation. Let n denote the number of the meshes, and $x = \{x_1, x_2, \cdots, x_n\}$ and $y = \{y_1, y_2, \cdots, y_n\}$ represent the datasets for the MOSs and the quality scores of the meshes respectively. Each mesh corresponds to one MOS and one quality score. The PLCC between the MOSs and the quality scores is calculated as follows:

$$PLCC = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}, \tag{2.3}$$

where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ and $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ represent the mean value of the dataset x and the dataset y respectively. After the datasets x and y are sorted in the same order, either in the ascending order or in the descending order, two new datasets will be generated: $X = \{X_1, X_2, \cdots, X_n\}$ and $Y = \{Y_1, Y_2, \cdots, Y_n\}$. The value of X_i is the rank of x_i in the dataset x while the value of Y_i is the rank of y_i of in the dataset y. The SROCC between the MOSs and the quality scores is calculated as follows:

$$SROCC = 1 - \frac{6}{n(n^2-1)}\sum_{i=1}^{n} d_i^2, \tag{2.4}$$

where $d_i = X_i - Y_i$. The SROCC between the datasets x and y is also considered as the PLCC between the ranked datasets X and Y.

As the Video Quality Experts Group (VQEG) pointed out in the report on the validation of objective quality metrics [84], the nonlinear quality rating compression may exist at the extremes of the test range during the subjective tests. In order to remove the nonlinearity due to the subjective rating process and facilitate the comparison of quality metrics in a common analysis space, firstly, the relationship between the objective predictions of each metric and the subjective ratings should be estimated by performing a psychometric fitting between the computed quality scores and the corresponding MOS values. Then, a fitted nonlinear function with the parameter values obtained from the psychometric fitting is used to transform the set of objective quality scores to a set of predicted MOS values. Finally, the performance of the quality metric is evaluated by computing the correlation coefficients between the predicted MOS values and the actual MOS values acquired from the subjective tests. In many research works on both MVQ metrics and IVQ metrics [1, 4–6, 85, 86], the psychometric fitting was performed to remove the nonlinearity. The commonly used psychometric functions include the cumulative Gaussian function [1, 4, 5], the four-parameter logistic function [85], the five-parameter psychometric function [6, 86]. The parameters of the psychometric functions are determined by minimizing the sum of squared differences between the mapped quality scores and MOS values.

In literature [1, 4, 5], the cumulative Gaussian function was used to perform the psychometric fitting:

$$g(Q) = \frac{1}{\sqrt{2\pi}} \int_{a+bQ}^{+\infty} e^{-(t^2/2)} dt, \tag{2.5}$$

where Q is the quality score and $g(Q)$ is the mapped quality score. The values of parameters a and b are determined through the psychometric fitting.

In literature [85], the four-parameter logistic function was used to perform the psychometric fitting:

$$f(Q) = \frac{a-b}{1+exp(-\frac{Q-c}{d})} + b, \tag{2.6}$$

where Q is the quality score and $f(Q)$ is the mapped quality score. The values of parameters a, b, c and d are determined through the psychometric fitting.

In literature [6, 86], the five-parameter psychometric function was used to perform the psychometric fitting:

$$f(Q) = a \cdot \left(\frac{1}{2} - \frac{1}{1+exp\,(b \cdot (Q-c))} \right) + d{\cdot}Q + e, \tag{2.7}$$

where Q is the quality score and $f(Q)$ is the mapped quality score. The values of parameters a, b, c, d and e are determined through the psychometric fitting.

In the research field of image quality assessment [8, 87–89], the performance of the IVQ metrics is evaluated with several indicators, including PLCC, SROCC, Kendall rank-order correlation coefficient (KROCC), root mean square error and mean absolute error. When the PLCC, SROCC and KROCC coefficients between the quality scores and MOSs are computed for performance evaluation, the premise is that the MOS correctly reflects the visual quality of the object, either the mesh or the image. However, there is a limitation of the visual sensitivity for the human eyes, that is, a small amount of distortion on the mesh or the image may not be visually distinguishable and there is some uncertainty for the subjective opinion scores [90]. Therefore, it might be hard to fully reflect the performance of the quality metrics by only considering the correlation between the MOSs and the objective quality scores. In the research field of image quality assessment, some scholars investigated the performance evaluation method for the IVQ metrics. Cheon and Lee [91] addressed the issue of perceptual ambiguity of the IVQ metrics. They proposed an approach to define the degree of the ambiguity with the just noticeable difference (JND) [92] of the image. The ambiguity intervals at different quality levels were constructed to evaluate the reliability of the IVQ metric. Later, Cheon and Lee [93] evaluated the performance of seven IVQ metrics in the case of image compression by considering both the accuracy and the ambiguity of the metrics. The ambiguity of the metric was quantified by three statistics (maximum, mean, and standard deviation) of the ambiguity intervals. The literature [93] showed that the IVQ metric with a higher accuracy do not necessarily have a better ambiguity performance. Lukáš et al. [89] investigated the uncertainty of the subjective opinion scores and proposed a framework to evaluate the classification capability of the IVQ metrics. They tested the framework with five IVQ metrics and analyzed the similarity and the superiority of the metrics.

Compared with the research works on image quality assessment, the research works on mesh quality assessment are considerably fewer and the investigation on the MVQ metric is still in its infancy. When evaluating the performance of the MVQ metrics [1–6], the scholars typically used the PLCC and SROCC to measure the prediction accuracy and monotonicity of the metrics. With gradually more research works on mesh quality assessment, the investigation on the performance evaluation of the MVQ metrics may attract more attention from the community. Since there are certain similarities between mesh quality assessment and image quality assessment, some theories and methodologies on the performance evaluation of IVQ metrics [89, 91–93] may provide inspirations for the investigation of performance evaluation methods for the MVQ metrics. For example, it may be interesting to investigate the ambiguity performance of the MVQ metrics by constructing the ambiguity intervals at different quality levels. In this thesis, we propose five MVQ metrics in Chapter 3, Chapter 4, Chapter 5, and Chapter 6, and employ the PLCC and SROCC coefficients to evaluate the performance of the MVQ metrics. In our future work, we will investigate the performance evaluation methods for the MVQ metrics.

2.4 Summary

In this Chapter, we have introduced three publicly available databases for mesh quality assessment, and reviewed the literature on the MVQ metrics. The MVQ metrics contain two categories: image-based MVQ metrics and model-based MVQ metrics. The model-based MVQ metrics generally achieve better performance for mesh quality assessment than the image-based IVQ metrics, and attract more attention of the scholars in the community. We have reviewed the spatial pooling methods for both the MVQ metrics and the IVQ metrics. The research works on the spatial pooling method for the MVQ metric are considerably less than the research works on the spatial pooling method for the IVQ metric. It is critical to design better spatial pooling methods for the MVQ metrics in order to achieve better performance for the mesh quality assessment. We have reviewed the literature on the incorporation of mesh saliency into the MVQ metric and the incorporation of image saliency into the IVQ metric. We have introduced the performance indicators and the psychometric fitting process for the MVQ metrics.

Chapter 3

Mesh Quality Prediction Based on the Statistical Descriptors

3.1 Introduction

In the community of image quality assessment, Li et al. [8] proposed an IVQ metric based on statistical descriptors and concluded that the overall quality of the distorted image is sensitive to the statistical distribution of the local quality map. This chapter investigates the relationship between the overall quality of the distorted mesh and the statistical characteristics of the local distortion map. We extract the statistical descriptors from the local distortion map as the feature vector to represent the mesh quality, and propose the Tensor-based Perceptual Distance Measure-Spatial Pooling (TPDMSP) metric [20] based on the statistical descriptors. We use the Support Vector Regression (SVR) model to learn the function that maps from the feature vector to the quality score of the distorted mesh. The SVR model has been used widely in the research field of image quality assessment [8, 11, 12] and shows good performance for image quality prediction. The pair of the feature vector and MOS of the mesh constitutes one mesh sample. We build the training dataset and testing dataset with mesh samples in the subjective databases [1, 3, 40].

The mesh samples in the training dataset are fed to the SVR model for training, and then the trained SVR model is used to predict the quality scores of the distorted meshes in the testing dataset. We evaluate the performance of the trained SVR model by computing the correlation coefficients between the predicted quality scores and the MOSs of the distorted meshes. Among state-of-the-art MVQ metrics [1–6], the TPDM metric [5] shows relatively high performance for mesh quality assessment. The TPDM metric reflects the visual masking effect of human visual system when evaluating the local distortions between the reference mesh and the distorted mesh, and thus achieves high correlation with the subjective opinions. The TPDM metric does not require the reference mesh and the distorted mesh to share the same topological connectivity. In this chapter, we adopt the TPDM metric to compute the local distortions of the meshes.

The remainder of this chapter is organized as follows. In Section 3.2, we give a brief introduction to the TPDM metric and how it computes the local distances between the vertices of the reference mesh and the distorted mesh. In Section 3.3, we use the TPDM metric to compute the local distortions of the mesh, and analyze the relationship between the statistical distribution of local distortions and the overall quality of the distorted mesh. In Section 3.4, we propose the TPDMSP metric based on statistical descriptors. We extract the statistical descriptors from the local distortions as the feature vector of the mesh, and employ the SVR model to learn the function that maps from the feature vector to the quality score of the mesh. In Section 3.5, we evaluate the performance of the TPDMSP metric on three publicly available databases. We demonstrate the performance superiority, and the inter-model and cross-database generalization capability of the TPDMSP metric in three experiments. We summarize this chapter in Section 3.6.

3.2 TPDM Metric

Torkhani et al. [5] evaluated the perceptual distance between the reference mesh and the distorted mesh based on the differences of curvature tensors, and proposed the Tensor-based Perceptual Distance Measure (TPDM) metric. The TPDM metric consists of a two-step computation process: firstly computing the local distortions of the vertices for the distorted mesh, and then pooling the local distortions into the quality score with the Minkowski method. It derives the visual difference between the reference mesh and the distorted mesh based on the distance between the curvature tensors of two meshes. It addresses the issue of changed connectivity for the distorted meshes by establishing a correspondence between the distorted mesh and the reference mesh. The Axis-Aligned Bounding Box (AABB) tree data structure is used to perform the vertex projection from the reference mesh M_r to the surface of the distorted mesh M_d. Each vertex v_i on the reference mesh has a corresponding point v_i' on the surface of the distorted mesh. The triangular facet T_i' that contains the point v_i' has three vertices $v_{i,1}'$, $v_{i,2}'$ and $v_{i,3}'$.

A number of excellent methods [41, 94] have been proposed to estimate the curvature tensor for the vertices of polyhedral surfaces. The most popular one was the method proposed by Cohen-Steiner and Morvan [41], who estimated the curvature tensor on the mesh based on the normal cycle theory. The curvature tensor of the vertex was calculated by averaging the tensors computed on the edges incident to the vertex on a geodesic disk window [41]. Let v denote a vertex on the mesh, B denote the geodesic disk window of vertex v, E denote the set of edges that are incident to vertex v, and e denote an edge in the set E. The curvature tensor $\mathscr{T}$ of vertex v is calculated as follows:

$$\mathscr{T}_v = \frac{1}{|B|}\sum_{e\in E}\beta(e)|e\cap B|\overrightarrow{e}\otimes\overrightarrow{e}, \tag{3.1}$$

where $|B|$ is the area of the geodesic disk, $\beta(e)$ is the angle between the normals of the triangles incident to edge e, $|e\cap B|$ is the length of the part of edge e inside B, and $\overrightarrow{e}$ is the unit vector with same direction as the edge e. The minimum curvature amplitude κ_{min} and maximum curvature amplitude κ_{max} are the absolute values of the eigenvalues of the tensor $\mathscr{T}$. The minimum principal curvature direction γ_{min} and the maximum principal curvature direction γ_{max} are the two eigenvectors associated with the eigenvalues.

The TPDM metric used the method in literature [41] to estimate the curvature tensor of each vertex on the reference mesh M_r and distorted mesh M_d. Let $\mathscr{T}_{v_i}$ and $\mathscr{T}_{v'_{i,k}}$ ($1 \leq \text{k} \leq 3$) denote the curvature tensors of the vertices v_i and $v'_{i,k}$ respectively. The correspondence relationship between the principal curvature directions / amplitudes of $\mathscr{T}_{v_i}$ and $\mathscr{T}_{v'_{i,k}}$ is established based on the minimum angular distance criterion. For the minimum principal curvature direction γ_{min} of $\mathscr{T}_{v_i}$, the principal curvature direction γ'_1 of $\mathscr{T}_{v'_{i,k}}$ that has the smallest angular distance to γ_{min} is found as the corresponding direction. Accordingly, the minimum curvature amplitude κ_{min} of $\mathscr{T}_{v_i}$ corresponds to the curvature amplitude κ'_1 of $\mathscr{T}_{v'_{i,k}}$ that is associated to γ'_1. By using the criterion, for the maximum principal curvature direction γ_{max} and maximum curvature amplitude κ_{max} of $\mathscr{T}_{v_i}$, the corresponding principal curvature direction γ'_2 and curvature amplitude κ'_2 of $\mathscr{T}_{v'_{i,k}}$ can be found in a similar way. Then the local distance $LPD_{v_i,v'_{i,k}}$ between the vertex v_i in the reference mesh and the vertex $v'_{i,k}$ of triangular facet T'_i in the distorted mesh is computed as:

$$LPD_{v_i,v'_{i,k}} = RW_i^{(\gamma)} \cdot RW_i^{(\kappa)} \cdot \left(\frac{\theta_{min}}{(\pi/2)} \delta_{\kappa_{min}} + \frac{\theta_{max}}{(\pi/2)} \delta_{\kappa_{max}} \right), \tag{3.2}$$

where θ_{min} is the angle between the principal curvature directions γ_{min} and γ'_1, and θ_{max} is the angle between the principal curvature directions γ_{max} and γ'_2. $\delta_{\kappa_{min}}$ is the Michelson-like contrast of the curvature amplitudes κ_{min} and κ'_1: $\delta_{\kappa_{min}} = |\frac{\kappa_{min}-\kappa'_1}{\kappa_{min}+\kappa'_1+\varepsilon}|$ while $\delta_{\kappa_{max}}$ is the Michelson-like contrast of the curvature amplitudes κ_{max} and κ'_2: $\delta_{\kappa_{max}} = |\frac{\kappa_{max}-\kappa'_2}{\kappa_{max}+\kappa'_2+\varepsilon}|$, where the constant ε is fixed as 5% of the average mean curvature of the reference mesh. $RW_i^{(\gamma)}$ and $RW_i^{(\kappa)}$ are the roughness-based coefficients that are derived from the surface principal curvature directions and curvature amplitudes in the one-ring neighborhood of vertex v_i respectively. The coefficient $RW_i^{(\gamma)}$ is computed as follows: firstly, the principal curvature directions in the one-ring neighborhood of vertex v_i are projected on the tangent plane of vertex v_i, secondly, a local roughness value $LR_i^{(\gamma)}$ of vertex v_i is computed as the sum of two angular standard deviations of the projected minimum and maximum curvature directions, finally, all the local roughness values $LR_i^{(\gamma)}$ on the mesh are mapped into the range [0.1, 1.0] and the mapped roughness value of vertex v_i is taken as the coefficient $RW_i^{(\gamma)}$. The coefficient $RW_i^{(\kappa)}$ is computed as follows: firstly, the Laplacian of mean curvature amplitudes in the one-ring neighborhood of vertex v_i is normalized by the mean curvature of vertex v_i to generate a local roughness value $LR_i^{(\kappa)}$ for vertex v_i, secondly, all the local roughness values $LR_i^{(\kappa)}$ on the mesh are mapped into the range [0.1, 1.0], finally, the mapped roughness value of vertex v_i is taken as the coefficient $RW_i^{(\kappa)}$. Since $RW_i^{(\gamma)}$ and $RW_i^{(\kappa)}$ reflect the local surface characteristics of each vertex, the local distance $LPD_{v_i,v'_{i,k}}$ computed via Eq. (3.2) accounts for the visual masking effect of human visual system.

The local perceptual distance $LTPDM_{v_i}$ is computed as the barycentric interpolation of three local distances between vertex v_i and vertices $v'_{i,1}$, $v'_{i,2}$ and $v'_{i,3}$ respectively:

$$LTPDM_{v_i} = \sum_{k=1}^{3} b_k(v'_i) LPD_{v_i,v'_{i,k}}, \tag{3.3}$$

where $b_k(v'_i)$ is the k-th barycentric coordinate of point v'_i within facet T'_i. The TPDM metric computes the global perceptual distance between the reference mesh and the distorted mesh via the local surface area-weighted Minkowski pooling of the local perceptual distances $LTPDM_{v_i}$, $i = 1, 2, \cdots, N$:

$$TPDM = \left(\sum_{i=1}^{N} w_i \left| LTPDM_{v_i} \right|^q \right)^{\frac{1}{q}}, \tag{3.4}$$

where $w_i = S_i / \sum_{i=1}^{N} S_i$ is the surface weighting coefficient with S_i one third of the total area of all the incident facets of v_i. The Minkowski exponent q is set as $q = 2.5$ empirically.

3.3 Statistical Descriptors

Li et al. [8] analyzed the statistical characteristics of the local quality map of the image, and proposed an IVQ metric based on statistical descriptors. They demonstrated that the overall quality of the image is sensitive to the statistical distribution of the local quality map. They provided two feature vectors to characterize the statistical distribution of the local quality map: histogram and statistical descriptors. Their experimental results indicated that the IVQ metric based on statistical descriptors achieves slightly better performance than the IVQ metric based on the histogram. In this chapter, we compute the local distortion vector via Eq. (3.3), and propose to extract statistical descriptors from the local distortion map as the feature vector to represent the overall quality of the distorted mesh. The statistical descriptors consist of seven statistical characteristics of the local distortion map: mean ($Mean$), standard deviation (Std), minimum (Min), maximum (Max), first quartile (Q_1), second quartile (Q_2), and third quartile (Q_3).

Let $M(i)$ denote the local distortion vector that is sorted in the ascending order, and N denote the number of vertices in the reference mesh. Then the seven statistical characteristics are defined successively as follows:

$$Mean = \frac{1}{N}\sum_{i=1}^{N} M(i), \tag{3.5}$$

$$Std = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left(M(i) - \overline{M(i)}\right)^2}, \tag{3.6}$$

$$Min = min(M(i)) = M(1), \tag{3.7}$$

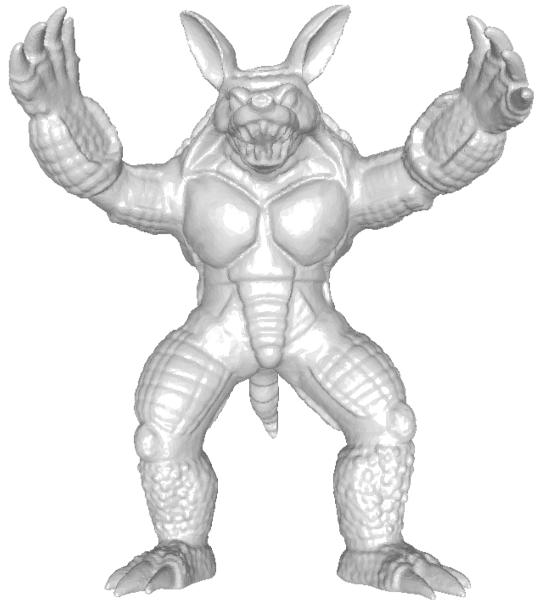
$$Max = max(M(i)) = M(N), \tag{3.8}$$

$$Q_1 = M\left((N+1)/4\right), \tag{3.9}$$

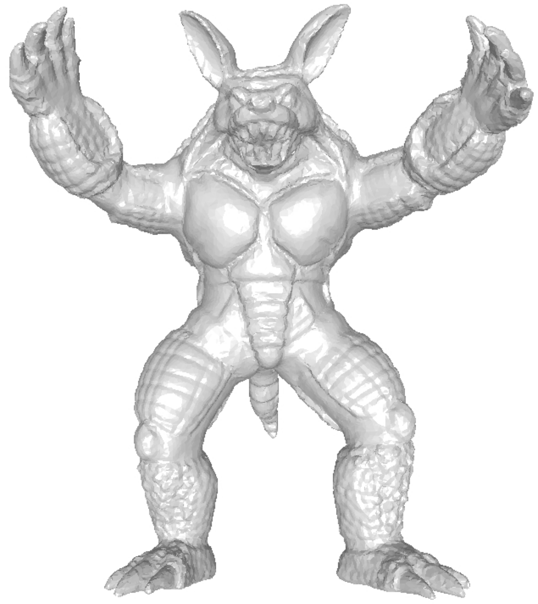
$$Q_2 = M\left((N+1)/2\right), \tag{3.10}$$

$$Q_3 = M\left(3(N+1)/4\right). \tag{3.11}$$

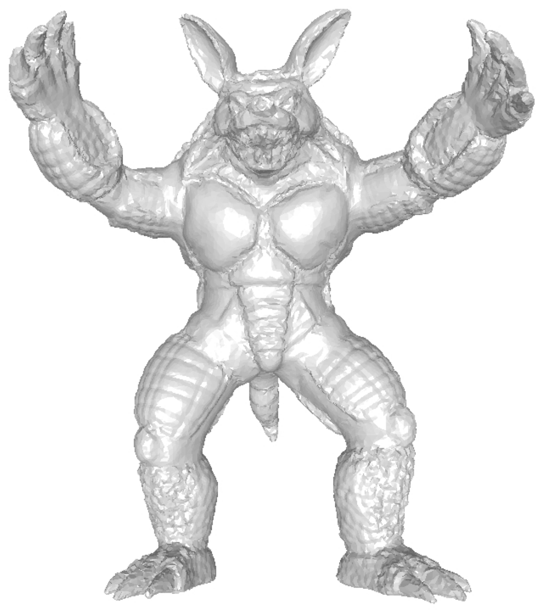
In order to demonstrate the relationship between the statistical distribution of the local distortions and the overall quality of the mesh, we analyze the statistical distribution of the local distortions with three distorted meshes of the Armadillo model that suffer distortions of different strengths in the LIRIS/EPFL general-purpose database [1]. Fig. 3.1 illustrates the reference mesh and the three distorted meshes of the Armadillo model. The three distorted meshes are generated by adding noise with different strengths to the rough



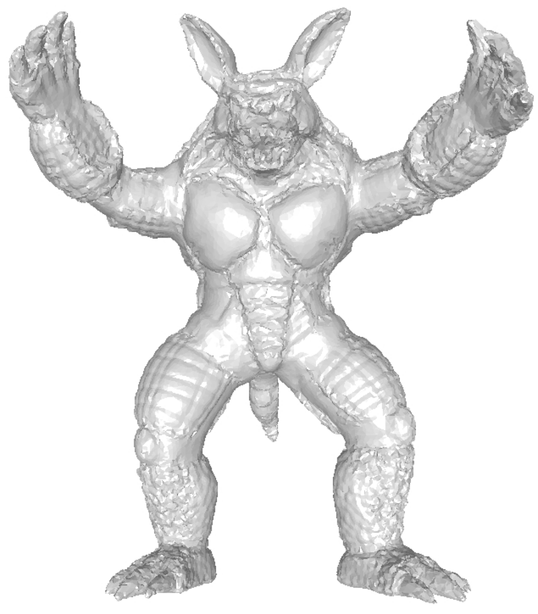
(a)



(b)



(c)



(d)

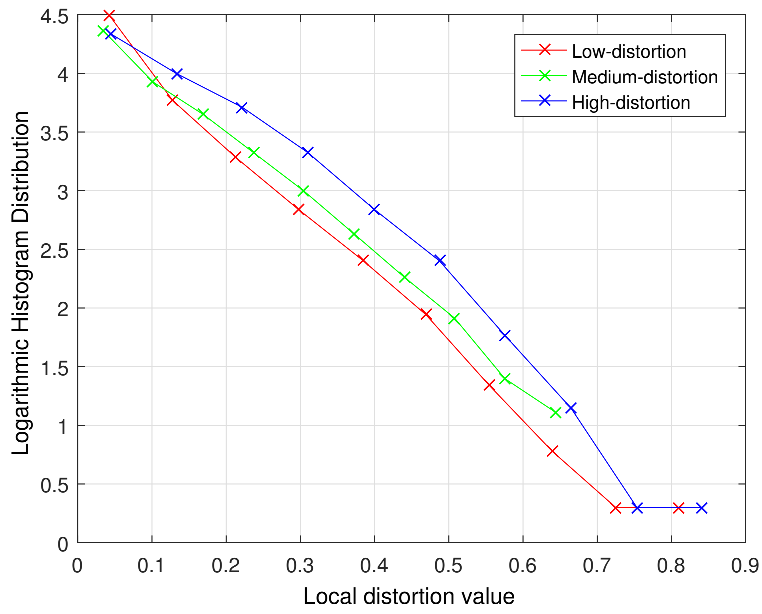
Figure 3.1 : The reference mesh and distorted meshes of the Armadillo model. (a) Reference mesh. (b) Low-distortion mesh. (c) Medium-distortion mesh. (d) High-distortion mesh. Their MOSs are 3.082, 3.393, 6.412, and 8.142, respectively.

Table 3.1 : The statistical descriptors and the MOSs of three distorted meshes of the Armadillo model.

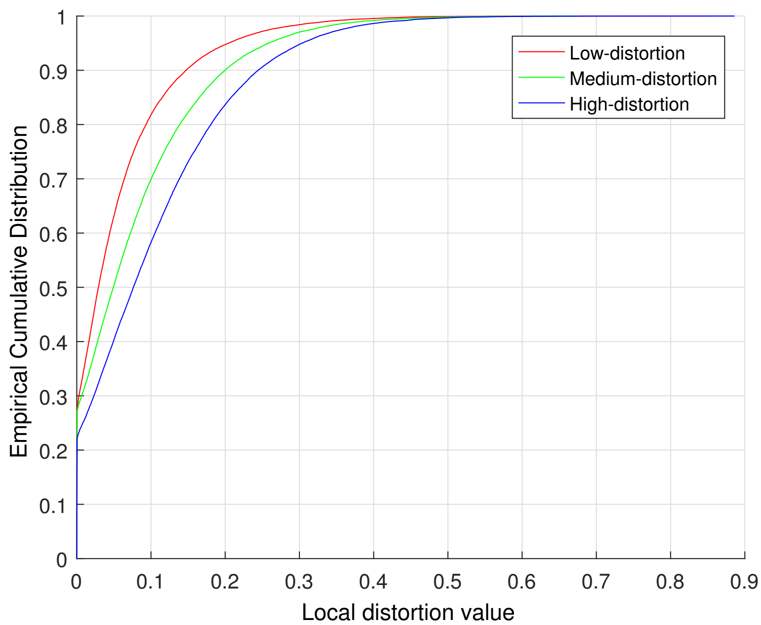
Distorted Mesh	Statistical Descriptors							MOS
	$Mean$	Std	Min	Max	Q_1	Q_2	Q_3	
Low-distortion Mesh	0.055	0.073	0	0.853	0	0.029	0.076	3.393
Medium-distortion Mesh	0.077	0.090	0	0.677	0	0.049	0.118	6.412
High-distortion Mesh	0.101	0.103	0	0.886	0.008	0.076	0.158	8.140

regions of the reference mesh. The three distorted meshes are referred to as the low-distortion mesh, medium-distortion mesh, and high-distortion mesh respectively according to the strength of the noise addition. A higher strength of noise addition corresponds to a higher-distortion mesh. In the LIRIS/EPFL general-purpose database, each of the meshes including the reference meshes has one MOS which lies in the range [0, 10]. A higher MOS value indicates a lower quality of the mesh.

We compare the distorted mesh with the reference mesh and construct a local distortion map for each distorted mesh via Eq. (3.3). As illustrated in Fig. 3.2, we generate a logarithmic histogram distribution and an empirical cumulative distribution (ECD) for each local distortion map of three distorted meshes of the Armadillo model. In Fig. 3.2a, we partition the distortion range into 10 equal ranges and build the histogram with 10 bins. We use the logarithm to represent the number of distortion samples within each bin. When the value of local distortion lies within the range [0.15, 0.65], we observe that the high-distortion mesh always has the largest number of distortion samples while the low-distortion mesh always has the smallest number of distortion samples. The values of local distortion within the range [0.15, 0.65] correspond to the regions with moderate or severe distortions on the mesh. The regions with severe local distortions typically have a great impact on the overall quality of the mesh. Thus, the differences in the logarithmic histogram distribution of three local distortion maps in Fig. 3.2a reflects the comparison



(a) Logarithmic histogram distribution



(b) Empirical cumulative distribution

Figure 3.2 : The statistical distributions of the distortion map for each of three distorted meshes of the Armadillo model.

among the MOSs of three distorted meshes. Fig. 3.2b illustrates the ECD for each local distortion map of three distorted meshes. This figure shows an intuitive comparison in the statistical characteristics of three distortion maps. For example, on the three curves in Fig. 3.2b, at the second quartile point Q_2 where the value of ECD is equal to 0.5, the local distortion values of three distortion maps are 0.029, 0.049, and 0.076, respectively. A higher-distortion mesh has a larger value for Q_2. Similar comparison can be observed for the third quartile point Q_3. The values of the statistical descriptors together with the MOS of each distorted mesh are listed in Table 3.1. The comparative analysis indicates that there is a strong correlation between the statistical descriptors of the local distortions and the overall quality of the mesh.

3.4 Mesh Visual Quality Metric Based on the Statistical Descriptors

Most existing MVQ metrics [1–6] firstly construct a local distortion map or local similarity map, and then accumulate the local distortions or similarities into the quality score of the mesh using the average summation or Minkowski summation method at the spatial pooling stage. In this chapter, we compute the local distortions of the mesh with the TPDM metric [5]. We present a spatial pooling method based on machine learning for mesh quality assessment and propose the TPDMSP metric [20] based on statistical descriptors. We firstly extract the statistical descriptors from the local distortions as the feature vector, then feed the feature vectors and MOSs of the meshes to the SVR model in order to learn the function that maps from the feature vector to the quality score of the mesh, and finally use the trained SVR model to predict the quality score of the mesh.

The schematic diagram of the TPDMSP metric is illustrated in Fig. 3.3. Given a reference mesh and a distorted mesh, the TPDMSP metric firstly computes the local distances between the vertices of the reference mesh and the vertices of the distorted mesh with

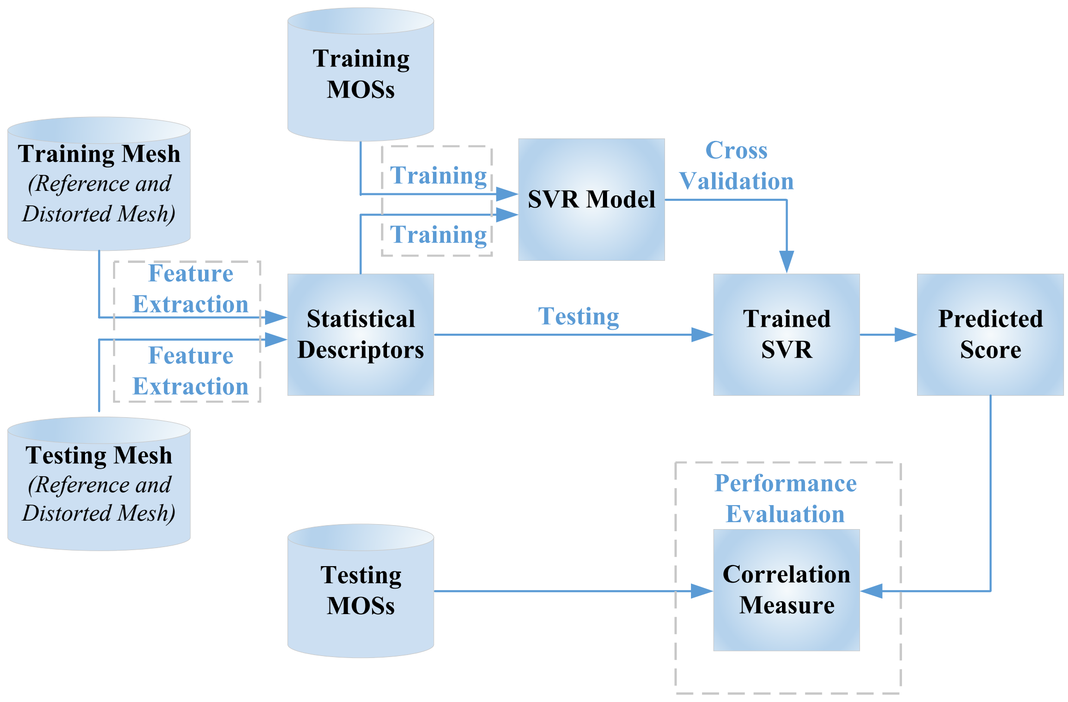


Figure 3.3 : The schematic diagram of the TPDMSP metric.

Eq. (3.3) as the local distortions of the mesh. A distortion vector that consists of all the local distortions is constructed for the distorted mesh and sorted in the ascending order. Secondly, the statistical descriptors are extracted from the sorted distortion vector as the feature vector that represents the overall qyality of the distorted mesh. The MOS and the feature vector of each distorted mesh constitute a pair of data sample. We construct the training dataset and testing dataset from the subjective database of mesh quality assessment [1, 3, 40]. Then, the training dataset is fed to the SVR model to learn the function that maps from the feature vector to the MOS of the distorted mesh. The parameters of the SVR model are determined via cross validation on the training dataset. After the training process, the trained SVR model is used to predict the quality scores of the distorted meshes in the testing dataset. The prediction accuracy of the trained SVR model is evaluated by computing the correlation coefficients between the predicted quality scores and the MOSs of the distorted meshes in the testing dataset. Finally, given a new pair of reference mesh and distorted mesh, we compute the local distortions through Eq. (3.3) and extract the feature vector from the local distortions. The feature vector is fed to the trained SVR to compute the quality score for the distorted mesh.

We take the statistical descriptors of the local distortions as the feature vector $\boldsymbol{x}$ of the distorted mesh, and represent the quality score Q of the distorted mesh as a function of the feature vector $\boldsymbol{x}$:

$$Q = f(\boldsymbol{x}), \tag{3.12}$$

where f the function that maps from the feature vector $\boldsymbol{x}$ to the quality score Q. Due to the complexity of human visual system (HVS) and limited knowledge about HVS, it is difficult to define such a function explicitly. We exploit the advantage of machine learning method to learn the nonlinear complex relationship between the feature vector and the perceptual quality score of the mesh. The SVR model has been well established and widely used in the community of image quality assessment [8, 11, 12]. A famous implementation of the SVR model was presented in LIBSVM [95]. In this chapter, we train the SVR model with the MOSs and the feature vectors of the meshes to find the optimal function f. As a popular regression method in the SVR model family [95], given the training sample set of sample-target pairs $< \boldsymbol{x}_i, y_i >$, the model ε-SVR tries to find an optimal function f subject to that the maximum deviation between the target y_i and the mapped value $f(\boldsymbol{x}_i)$ doed not exceed the predefined threshold ε. The function f to be learned is:

$$f(x) = w^T \psi(\boldsymbol{x}) + b, \tag{3.13}$$

where w is the weight vector, $\psi(\boldsymbol{x})$ is a nonlinear function of $\boldsymbol{x}$, and b is the bias. The function $\psi(\boldsymbol{x})$ maps the feature vector $\boldsymbol{x}$ to a higher-dimensional space and finds a linear separating hyperplane with the maximal margin in the higher-dimensional space. At the training stage, the regularization parameter C is used to control the optimization of the separating hyperplane. Let $\boldsymbol{x}_i$ denote the feature vector of the i-th mesh in the training dataset and y_i denote the MOS of the i-th mesh. The values of the weight vector w and the bias b are determined in the training process subject to the condition:

$$|f(\boldsymbol{x}_i) - y_i| \leq \varepsilon. \tag{3.14}$$

Let $K(\boldsymbol{x}_i, \boldsymbol{x}_j) \equiv \psi(\boldsymbol{x}_i)^T \psi(\boldsymbol{x}_j)$ denote the kernel function in the SVR model. Several kernel functions are available for training the SVR model. Since the radius basis function (RBF) kernel performs well for fitting a nonlinear function [95], we choose the RBF kernel as the kernel function of SVR model to fit the function $f(\boldsymbol{x})$. The RBF kernel is expressed as follows:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp(-\gamma \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2), \tag{3.15}$$

where the parameter $\gamma\,(\gamma > 0)$ controls the radius. We use the cross-validation strategy on the training dataset to determine the optimal values for the parameter set $\{C, \gamma, \varepsilon\}$.

3.5 Experimental Results and Analysis

3.5.1 Experimental Protocol

In this section, we evaluate the performance of the TPDMSP metric on three subjective databases: LIRIS/EPFL general-purpose database [1], LIRIS masking database [40], and UWB compression database [3]. Compared with the subjective databases [8] for image quality assessment, there are a smaller number of data samples in the subjective databases for mesh quality assessment. Accordingly, the number of data samples in the training dataset and the testing dataset is also small. However, the SVR model shows good performance even with a small number of training data samples [95]. The proper size of training data samples depends on the length and the representational capability of the feature vector. The length of the feature vector of the TPDMSP metric is 7. The LIRIS masking database [40] consists of only 24 distorted meshes. It is difficult to divide the database into the training dataset and testing dataset. So it is not appropriate to train the SVR model on the LIRIS masking database. The UWB compression database [3] includes 5 models, each of which corresponds to one reference mesh and twelve or thirteen distorted meshes. The database consists of 68 meshes, but the number of distorted meshes for each model is relatively small and the MOSs of distorted meshes for different models are uncorrelated. Therefore, the UWB compression database is not suitable for training either.

The experiments in this chapter include three parts: random sample testing experiment on a single database, inter-model testing experiment on the single database, cross-database testing experiment. (1) In the random sample testing experiment, the mesh samples in the training dataset and testing dataset are obtained randomly from the LIRIS/EPFL general-purpose database [1]. The feature vectors and the MOSs of the meshes in the training dataset are fed to the SVR model for training. Afterwards, the trained SVR model is used to predict the quality scores of the meshes in the testing dataset. Then, we compute the correlation coefficients between the predicted quality scores and the MOSs of the meshes in the testing dataset as the performance indicators of the trained SVR model. We evaluate the accuracy of the TPDMSP metric in predicting the mesh quality on the same database by comparing the performance of the trained SVR model with the performance of state-of-the-art MVQ metrics. (2) In the inter-model testing experiment, the LIRIS/EPFL general-purpose database [1] is divided into four datasets. Each dataset consists of the MOSs and feature vectors of the meshes for each model. Three datasets constitute the training dataset and the fourth dataset constitutes the testing dataset. The training dataset is fed to the SVR model for training. After the training, the trained SVR model is used to predict the quality scores of the meshes in the testing dataset. By taking one of the four datasets as the testing dataset each time, we conduct the training and testing experiment for four times. We compute the correlation coefficients between the predicted quality scores and the MOSs of the meshes in the testing dataset, and evaluate the inter-model generalization capability of the TPDMSP metric. (3) In the cross-database testing experiment, all the meshes in the LIRIS/EPFL general-purpose database [1] constitute the training dataset. The distorted meshes in the LIRIS masking database [40] and the UWB compression database [3] constitute two testing datasets respectively. The MOSs and feature vectors of the meshes in the training dataset are fed to the SVR model for training. After the training process, the trained SVR model is used to predict the quality scores of the meshes in each of two testing datasets. We evaluate the cross-database generalization capability of the TPDMSP metric by calculating the correlation coefficients between the predicted quality scores and MOSs of the meshes on each testing dataset.

In this chapter, we use the PLCC and SROCC coefficients between the the predicted quality scores and MOSs of the meshes as the performance indicators of the TPDMSP metric. Most existing MVQ metrics perform a curve fitting between the predicted quality scors and MOSs with a psychometric function before computing the correlation coefficients. We apply the cumulative Gaussian function for the psychometric fitting in this chapter, and compute the correlation coefficients between the mapped quality scores and MOSs as the performance indicators of the trained SVR model. The MOSs and the predicted quality scores are normalized before the psychometric fitting. Since the MOSs and feature vectors of the meshes are used to train the SVR model, the predicted quality scores of the testing meshes lie in the same scale with the MOSs.

3.5.2 Random Sample Testing Experiment

In the random sample testing experiment, we divide the meshes in the LIRIS/EPFL general-purpose database [1] randomly into two separate datasets: training dataset and testing dataset. The training dataset consists of 80% of the meshes in the database while the testing dataset consists of the remaining 20% of the meshes. After the training and testing process, we compute the PLCC and SROCC coefficients between the MOSs and predicted quality scores of the meshes in the testing dataset. We repeat the procedure 50 times. We denote the median performance and optimal performance of the TPDMSP metric among 50 tests as TPDMSP(m) and TPDMSP(o) respectively. Table 3.2 lists the performance values of TPDMSP(m), TPDMSP(o) and state-of-the-art MVQ metrics including MSDM [1], MSDM2 [2], DAME [3], FMPD [4], TPDM [5], Dong [6], 3DWPM1 [60] and 3DWPM2 [60] on the LIRIS/EPFL general-purpose database.

Table 3.2 : The performance comparison between the TPDMSP metric and state-of-the-art MVQ metrics on the LIRIS/EPFL general-purpose database.

Metrics	Correlation coefficients(%)	
	PLCC	SROCC
HD	11.4	13.8
RMS	28.1	26.8
GL1	35.5	33.1
GL2	42.4	39.3
SF	7.0	15.7
3DWPM1	61.8	69.3
3DWPM2	49.6	49.0
MSDM	75.0	73.9
MSDM2	81.4	80.4
FMPD	83.5	81.9
DAME	75.2	76.6
TPDM	84.1	84.3
Dong	87.7	86.6
TPDMSP(m)	89.0	87.5
TPDMSP(o)	95.9	96.1

From Table 3.2, we observe that the performance values of both TPDMSP(o) and TPDMSP(m) are higher than the performance values of state-of-the-art MVQ metrics. The TPDMSP metric uses the TPDM metric to generate the local distortions of the mesh, but the performance of the TPDMSP metric is significantly better than the performance of the TPDM metric. The difference between the TPDMSP metric and the TPDM metric is that the TPDMSP metric takes the statistical descriptors of local distortions as the feature vector and uses the SVR model to learn the function that maps from the feature vector to the quality score. The comparison result reveals that the statistical descriptors of local distortions reflect the quality of the mesh well and the SVR model has learned the mappig function well. The LIRIS/EPFL general-purpose database involves two types of distortions: noise addition and smoothing. As shown in Table 3.2, the TPDMSP metric achieves good prediction performance when a random 80% subset of the meshes constitutes the training dataset and the remaining 20% subset of the meshes constitutes the testing dataset. This indicates the TPDMSP metric works well for both noise addition and smoothing. The TPDMSP metric will work well for the distorted meshes that are generated from watermarking, compression, and simplification of the reference meshes since mesh watermarking and mesh compression are both noise-like distortions, and mesh simplification is a smoothing-like distortion that has a low-pass effect on the mesh [1]. The TPDMSP metric is a full-reference MVQ metric and requires the reference mesh to be completely available. In the case of mesh reconstruction, the original underlying mesh (the reference mesh) is typically unknown and only the reconstructed mesh is available. Thus, the TPDMSP metric is not applicable to the distortions that result from mesh reconstruction. Fig. 3.4 illustrates the scatter plots of MOSs versus predicted quality scores of the testing meshes and the fitted psychometric curve in one random sample testing experiment on the LIRIS/EPFL general-purpose database. From Fig. 3.4, we observe that the fitted psychometric curve fits the *Predicted Score–MOS* pairs of the testing meshes well.

In order to investigate the dependency of the performance of the SVR model on the size of the training dataset, we show how the performance of the TPDMSP metric varies with different amounts of training samples. We record the performance indicators of the TPDMSP metric in six cases of dataset division: 10%, 20%, 30%, 40%, 60%, and 80% of distorted meshes constitute the training dataset while the remaining 90%, 80%, 70%,

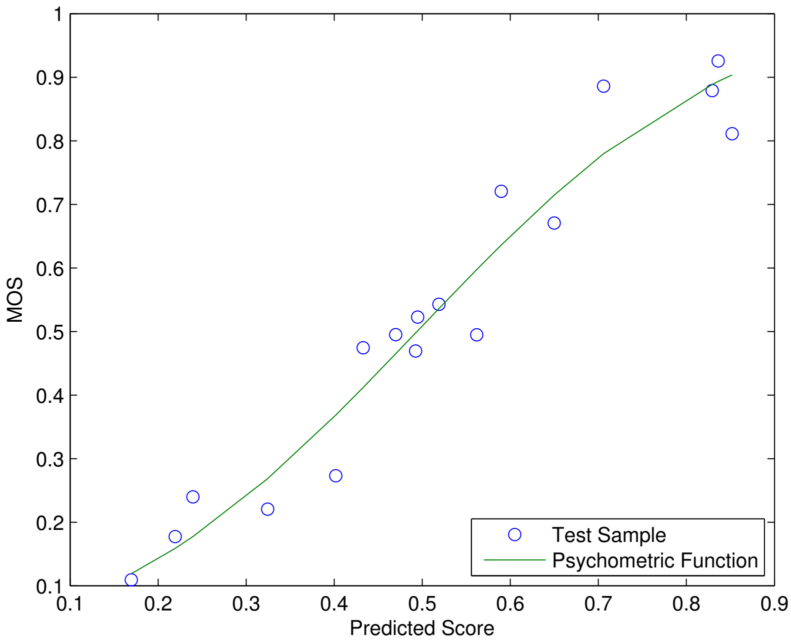


Figure 3.4 : The scatter plots of MOSs versus predicted quality scores of the testing meshes and the fitted psychometric curve in one random sample testing experiment on the LIRIS/EPFL general-purpose database.

60%, 40%, and 20% of distorted meshes constitute the testing dataset. In each case, we compute the PLCC and SROCC coefficients of the TPDMSP metric for the testing meshes after the training and testing process, and repeat the training and testing process 50 times. We denote the median performance of PLCC and SROCC by PLCC(m) and SROCC(m), and denote the optimal performance of PLCC and SROCC by PLCC(o) and SROCC(o). Fig. 3.5 illustrates the prediction performance of the TPDMSP metric in the case of different amounts of training samples on the LIRIS/EPFL general-purpose database. From Fig. 3.5, we observe that the performance of the TPDMSP metric generally improves as the amount of training samples increases. The TPDMSP metric shows good performance even in the case of a small amount of training samples.

The feature vector of the distorted mesh in the TPDMSP metric consists of seven statistical characteristics of the local distortions: mean ($Mean$), standard deviation (Std),

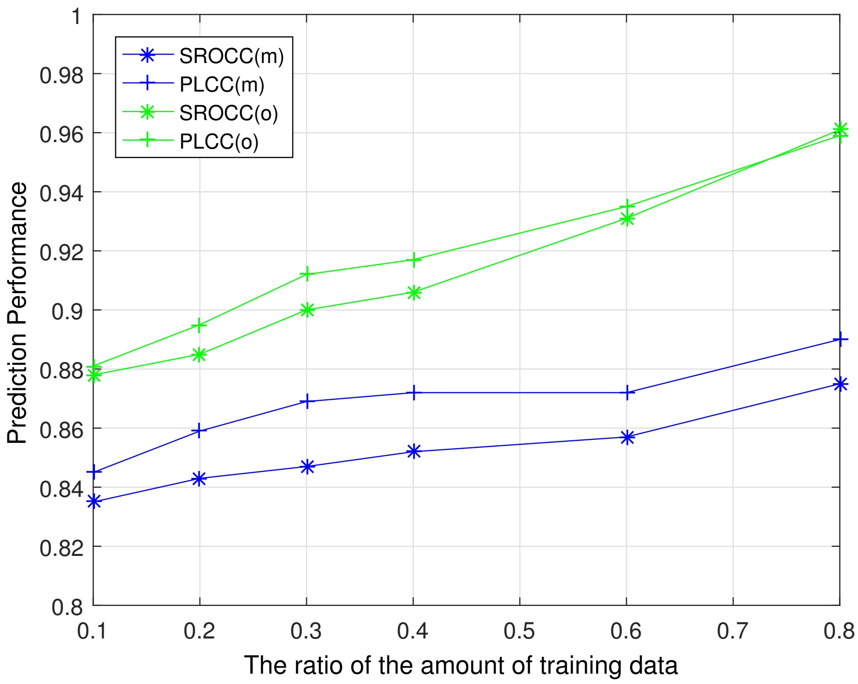


Figure 3.5 : The prediction performance of the TPDMSP metric in the case of different amounts of training samples on the LIRIS/EPFL general-purpose database.

minimum (Min), maximum (Max), the first quartile (Q_1), the second quartile (Q_2), and the third quartile (Q_3). We investigate the contribution of each statistical characteristic to the prediction performance of the TPDMSP metric. We assume that removing a feature in the feature vector will lead to a degradation of the prediction performance, and the degradation of the prediction performance will be greater when the removed feature has a larger contribution. We remove one of the seven statistical characteristics and form a new feature vector with the other six statistical characteristics. We feed the new feature vectors to the SVR model, and evaluate the prediction performance of the TPDMSP metric when removing any of the seven statistical characteristics. Fig. 3.6 illustrates the PLCC and SROCC coefficients of the TPDMSP metric on the LIRIS/EPFL general-purpose database in the case that none of the seven statistical characteristics is removed and in the case of removing any of the seven statistical characteristics. The PLCC(m) and SROCC(m) denote the median value of the PLCC and SROCC coefficients in the 50 tests respectively.

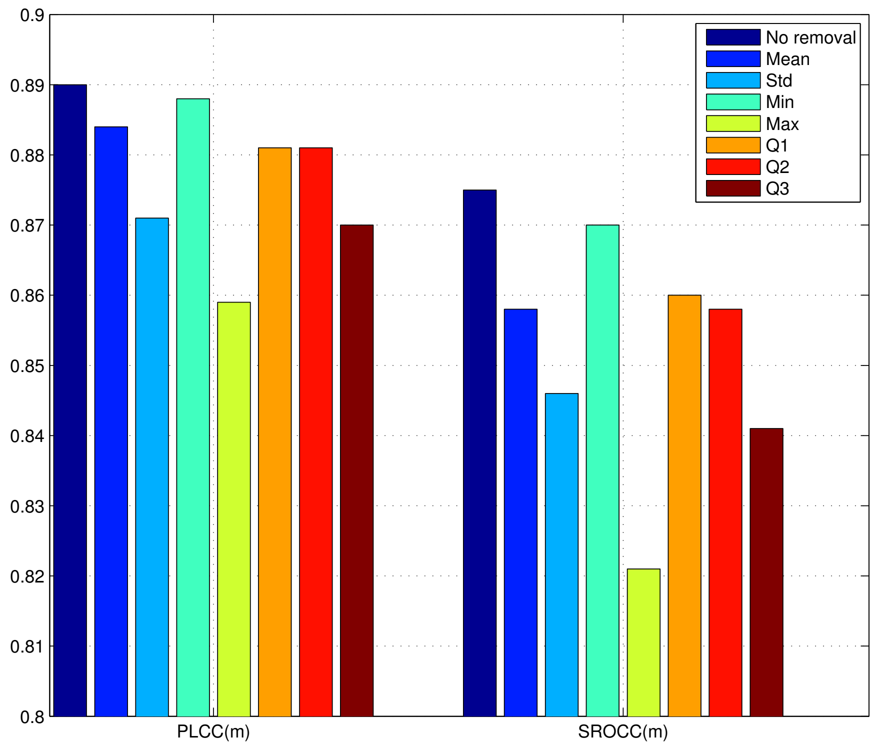


Figure 3.6 : The prediction performance of the TPDMSP metric on the LIRIS/EPFL general-purpose database in the case that none of the seven statistical characteristics is removed and in the case of removing any of the seven statistical characteristics.

In the legend of Fig. 3.6, the '*No removal*' label indicates that none of the seven statistical characteristics is removed while the other seven labels indicate that the current statistical characteristic is removed from the feature vector. The performance of the TPDMSP metric in the case of '*No removal*' is exactly the performance of TPDMSP(m) which is reported in Table 3.2. From Fig. 3.6, we observe that when removing any statistical characteristic, both values of PLCC(m) and SROCC(m) are lower than the values of PLCC(m) and SROCC(m) when none of the seven statistical characteristics is removed. When the *Max* feature is removed, the values of PLCC(m) and SROCC(m) are smallest and the degradation of the prediction performance is greatest. This indicates that the *Max* feature has the greatest contribution among the seven statistical characteristics. When comparing the *Max* feature or the Q_3 feature with the other five features, we observe that the values of PLCC(m) and SROCC(m) in the case of removing either the Max feature or the Q_3 feature are lower than the values of PLCC(m) and SROCC(m) in the case of removing the other five features. This reflects the subjective judgement of human visual system, that is, the regions with severe local distortions always attract more attention from human eyes and have a greater impact on the overall quality of the mesh. We also observe that the degradation of the prediction performance is smallest when the Min feature is removed. This indicates that the regions with slight local distortions have a less impact on the overall quality of the mesh. The values of PLCC(m) and SROCC(m) in the case of removing the Std feature are lower than the values of PLCC(m) and SROCC(m) in the case of removing the $Mean$ feature, which indicates that the feature Std has a greater contribution than the $Mean$ feature to the prediction performance.

Table 3.3 : The average running time performance of the main computational tasks in the TPDMSP metric.

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

The TPDMSP metric mainly involves the following computational tasks: computing local perceptual distortions, extracting statistical descriptors, SVR training, and SVR testing. The TPDMSP metric 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. Let n denote the number of vertices in the reference mesh. The time complexity of computing local distortions is $O(n)$. After generating local perceptual distortions, the metric extracts statistical descriptors from the local distortion vector. We adopt the sort routine of

MATLAB, which achieves vector sorting with the Quick Sort algorithm, to sort the local distortion vector. The time complexity of sorting the local distortion vector is $O(n{\cdot}\log_2^n)$. We then compute seven elements of statistical descriptors from the sorted local distortion vector. 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 MOSs of each distorted mesh into a data sample. At the training stage, the SVR model is trained with the training dataset. We use the RBF kernel as the kernel function of SVR model. Let n_{sample} denote the number of training samples and d denote the length 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 is 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 $O(n_{sv}{\cdot}d)$.

We use the LIRIS/EPFL general-purpose database to demonstrate the running time performance of the TPDMSP metric. 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 is a reference mesh for each of four models: Armadillo, Dinosaur, RockerArm, and Venus. The vertex numbers of four reference meshes are in the same order of magnitude, and they are 40002, 42146, 40177, and 49666, respectively. In Table 3.3, we report the average running time performance of the 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 the 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 ms. 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 ms. 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 for training and testing. The average time for training the SVR model is 1.16 ms while the average time for testing with the trained SVR model is 0.14 ms. Since both the sample size of training dataset and the dimension of feature vector are small, as expected, the time for SVR training is short. 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 the TPDMSP metric is in computing local perceptual distortions. Compared with computing local perceptual distortions, the time consumed by extracting statistical descriptors, SVR training, and SVR testing is almost negligible. Overall, the TPDMSP metric has good running time performance and can evaluate the quality of a distorted mesh in a short time.

3.5.3 Inter-model Testing Experiment

In order to evaluate the inter-model generalization capability of the TPDMSP metric, we conduct the inter-model testing experiment on four models (Armadillo, Dinosaur, RockerArm, and Venus) in the LIRIS/EPFL general-purpose database [1]. We use the distorted meshes of three models as the training dataset and the distorted meshes of the fourth model as the testing dataset. We compute the PLCC (r_p) and SROCC (r_s) coefficients between the MOSs and the predicted quality scores of the testing meshes as the performance indicators of the inter-model generalization capability of the TPDMSP metric.

We provide a comparison between the inter-model generalization performance of the TPDMSP metric and the quality assessment performance of state-of-the-art metrics in Table 3.4. Note that state-of-the-art metrics in Table 3.4 are not suitable for inter-model

Table 3.4 : The comparison between the inter-model generalization performance of the TPDMSP metric and the quality assessment performance of state-of-the-art MVQ metrics on the LIRIS/EPFL general-purpose database.

Metric	Armadillo		Dinosaur		RockerArm		Venus		All models		Training
	r_p	r_s	r_p	r_s	r_p	r_s	r_p	r_s	r_p	r_s	
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
TPDMSP	86.3	82.5	90.9	90.9	87.2	90.8	88.0	86.2	88.1	87.6	Yes

prediction because they are not based on a machine learning paradigm. The performance values of these metrics for each geometry model in Table 3.4 are obtained by using an explicit pooling method and setting the value of parameter properly. However, it is non-trivial to determine the pooling method and the parameters in different experimental settings. Most state-of-the-art metrics calculate the overall quality score for the mesh using the average summation [3, 4, 6] or the Minkowski summation [1, 2, 5], and choose the values for the parameters in an ad-hoc way. Moreover, their performance values are calculated by considering all the meshes in the whole database while the TPDMSP metric does not involve the testing meshes at the training stage. The PLCC and SROCC values of state-of-the-art metrics are extracted from the literature [4]. The inter-model generalization performance values of the TPDMSP metric are listed in the last row of Table 3.4. The *All models* performance values of the TPDMSP metric are calculated by averaging the four performance values for each testing model.

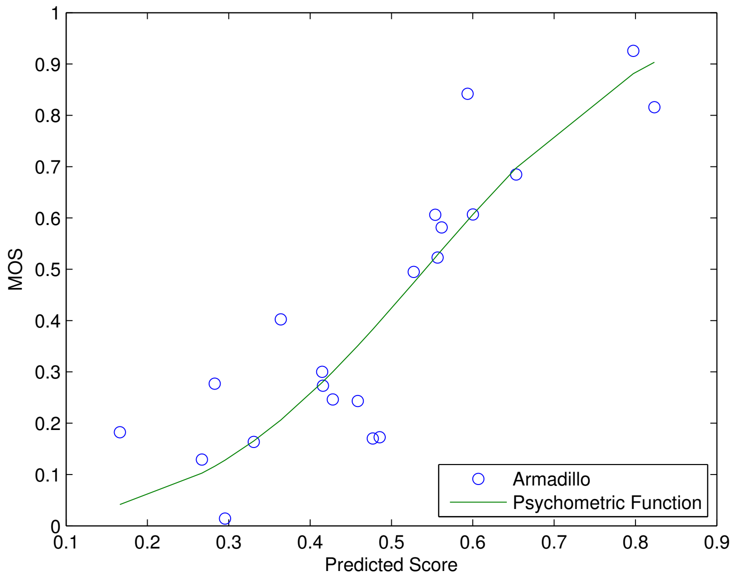


Figure 3.7 : The scatter plots of MOSs versus predicted quality scores and the fitted psychometric curve for the Armadillo model of the LIRIS/EPFL general-purpose database in the inter-model testing experiment.

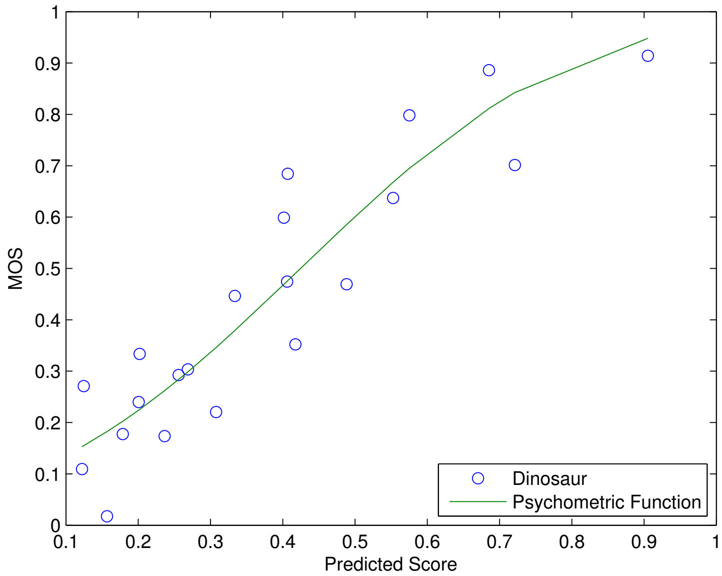


Figure 3.8 : The scatter plots of MOSs versus predicted quality scores and the fitted psychometric curve for the Dinosaur model of the LIRIS/EPFL general-purpose database in the inter-model testing experiment.

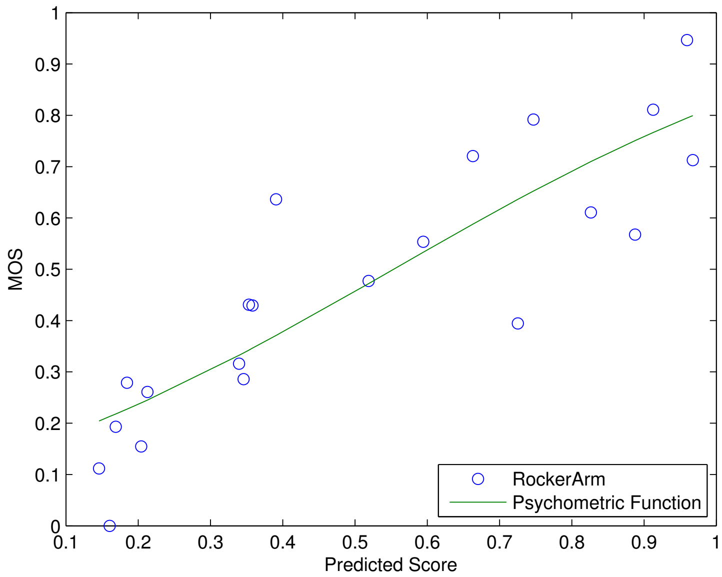


Figure 3.9 : The scatter plots of MOSs versus predicted quality scores and the fitted psychometric curve for the RockerArm model of the LIRIS/EPFL general-purpose database in the inter-model testing experiment.

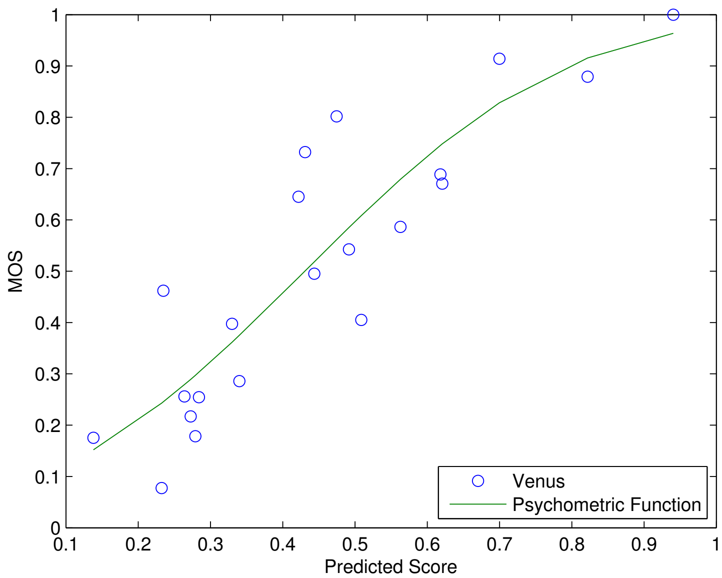


Figure 3.10 : The scatter plots of MOSs versus predicted quality scores and the fitted psychometric curve for the Venus model of the LIRIS/EPFL general-purpose database in the inter-model testing experiment.

From Table 3.4, we observe that the TPDMSP metric achieves consistently high prediction accuracy for different testing models and the average performance values of both PLCC and SROCC for the TPDMSP metric are highest in the *All models* column. The comparison indicates the TPDMSP metric has a good inter-model generalization capability with a high prediction accuracy. Fig. 3.7, Fig. 3.8, Fig. 3.9, and Fig. 3.10 illustrate the scatter plots of MOSs versus predicted quality scores and the psychometric fitting curve for each testing model in the LIRIS/EPFL general-purpose database in the inter-model testing experiment. For example, in Fig. 3.7, all the distorted meshes of the Armadillo model constitute the testing dataset while all the distorted meshes of the other three models constitute the training dataset. The training dataset is used to train the SVR model, and then the trained SVR model is used to predict the quality scores of the meshes of the Armadillo model. The cumulative Gaussian function is used to fit the *Predicted Score–MOS* pairs of the meshes for the Armadillo model. From Fig. 3.7, Fig. 3.8, Fig. 3.9, and Fig. 3.10, we observe that the *Predicted Score–MOS* pairs are fitted well by the cumulative Gaussian function curve.

3.5.4 Cross-database Testing Experiment

In this subsection, we evaluate the cross-database generalization capability of the TPDMSP metric on three databases. The meshes in the LIRIS/EPFL general-purpose database constitute the training dataset which is used to train the SVR model. The meshes in the LIRIS masking database and UWB compression database constitute the testing datasets, which are used to test the trained SVR model. Since there is a significant difference in the surface area size and vertex sample density between three databases, our preliminary experiment reveals that it is inappropriate to neglect the local area of the vertex when computing the local distortion. In order to reflect the influence of the local area on the cross-database generalization performance, we use the local area to weight the local distortion for each vertex:

$$WLTPDM_{v_i} = w_i^{\beta} \cdot LTPDM_{v_i}, \tag{3.16}$$

where $w_i = S_i / \sum_{i=1}^{N} S_i$ is the weighting coefficient of local area, S_i is one third of the total area of all the incident facets of vertex v_i, and β is the exponent for regulating the weight of the local area. The value of β is set as $\beta = -5$ empirically. After computing the area-weighted local distortion $WLTPDM_{v_i}$ for each vertex, we extract the statistical descriptors from the area-weighted local distortions as the feature vector, and conduct the cross-database testing experiment with the feature vectors and MOSs of the meshes.

Table 3.5 : The cross-database testing performance of the TPDMSP metric.

Performance	UWB compression						LIRIS masking
	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

Table 3.5 lists the cross-database testing performance values of the TPDMSP metric on the UWB compression database and the LIRIS masking database. Since the MOSs of distorted meshes for different models are uncorrelated in the UWB compression database, we compute the performance values for the distorted meshes of each model and then use the average performance value to represent the overall performance in the UWB compression database. From Table 3.5, we observe that the TPDMSP metric generally achieves high PLCC and SROCC performance values on both the UWB compression database and the LIRIS masking database. Note that the TPDMSP metric does not involve MOSs of testing meshes at the training stage. There are various differences between different subjective databases, including the coverage of distortion types, the coverage of geometric models, and scoring protocols. The cross-database testing performance in Table 3.5 is encouraging and it indicates that the TPDMSP metric has a good cross-database gener-

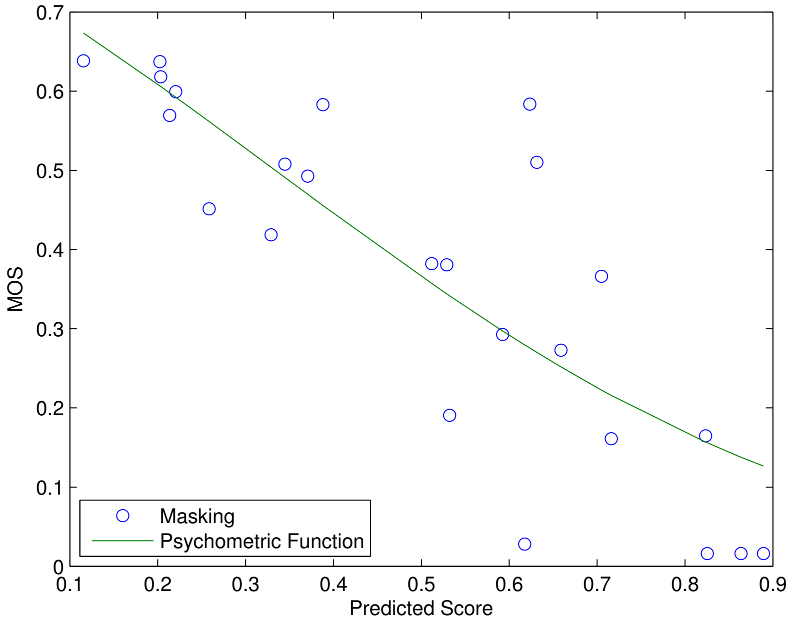
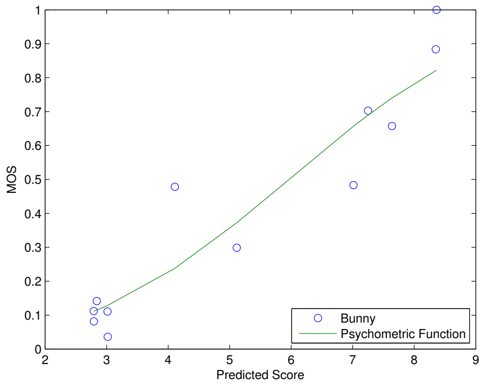
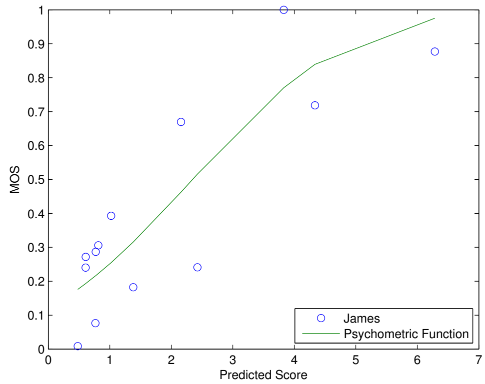


Figure 3.11 : The scatter plots of MOSs versus predicted quality scores and the fitted psychometric curve for the meshes in the LIRIS masking database in the cross-database testing experiment.

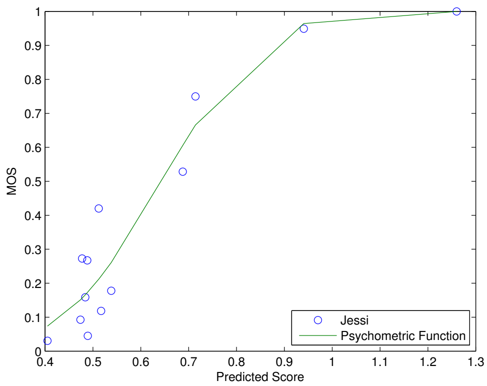
alization capability. We believe that such differences can be alleviated by constructing a larger database with a wide coverage of various factors, and the machine learning-based metric will show its strength when a large subjective database with a large number of ground-truth samples is constructed in the future. Fig. 3.11 illustrates the scatter plots of MOSs versus predicted quality scores and psychometric fitting curve for the meshes in the LIRIS masking database while Fig. 3.12 illustrates the scatter plots of MOSs versus predicted quality scores and psychometric fitting curve for the meshes of each model in the LIRIS masking database in the cross-database testing experiment. The MOSs of all the testing meshes are normalized. Fig. 3.11 and Fig. 3.12 show that the psychometric curves generally fit the *Predicted Quality Score–MOS* pairs of the meshes well.



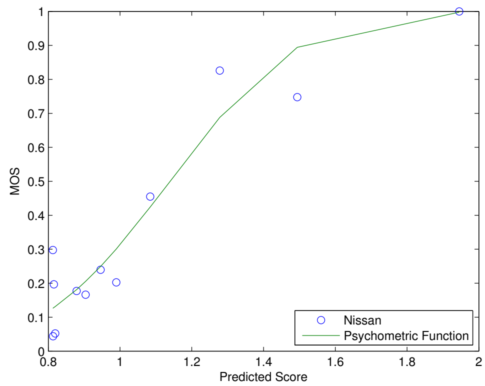
(a) Bunny



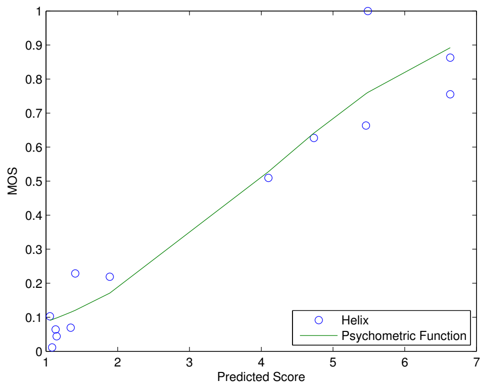
(b) James



(c) Jessi



(d) Nissan



(e) Helix

Figure 3.12 : The scatter plots of MOSs versus predicted quality scores and the fitted psychometric curve for the meshes of each model in the UWB compression database in the cross-database testing experiment.

3.6 Summary

In this chapter, we have proposed the TPDMSP metric based on statistical descriptors. The experimental results demonstrate that the statistical descriptors of the local distortions reflect the mesh quality well, and the severely distorted regions on the mesh have a great impact on the mesh quality. The experiments on three publicly available databases demonstrate the performance superiority of the TPDMSP metric over state-of-the-art metrics, and good generalization capability of the TPDMSP metric among different models and different databases. The TPDMSP metric achieves good performance for mesh quality prediction even with a small amount of training samples. The experiment reveals that, among the seven features in the statistical descriptors, the Max feature has the greatest contribution to the feature vector while the Min feature has the least contribution to the feature vector. The TPDMSP metric shows good runtime performance and is able to compute the quality score for the distorted mesh in a short time.

We used the TPDM metric to compute the local distortions in this chapter, but the mesh quality prediction method based on the statistical descriptors and SVR model can be generalized to other MVQ metrics that are based on the computation of local quality map. The existing subjective databases for mesh quality assessment have a small number of mesh samples. When a larger subjective database that involves more geometric models and more types of distortions is available in the future, the TPDMSP metric will show its strength in machine learning. The experiment results demonstrate that the statistical descriptors reflect the mesh quality well, but how the seven statistical characteristics in the statistical descriptors interact with each other remains an open question and needs further investigation. In Chapter 4, we will further investigate the perceptual importance of the severely distorted regions based on the conclusions in this chapter, and build a visual importance model to emphasize the impact of the severely distorted regions on the mesh quality.

Chapter 4

Spatial Pooling with the Percentile Weighting Strategy

4.1 Introduction

In the community of image quality assessment, Moorthy and Bovik [19] proposed a spatial pooling method for the image visual quality metric based on the percentile weighting strategy. After computing the local similarity map of the image, they assigned a greater weight to the pixels with low similarity values to emphasize the impact of the severely distorted regions on the image quality. They demonstrated the effectiveness of the percentile weighting strategy in the image visual quality metric [19]. In the community of mesh quality assessment, most state-of-the-art mesh visual quality metrics [1–6] evaluate the visual quality of the mesh by following a two-step process: firstly constructing a local quality map (either distortion map or similarity map) between the reference mesh and the distorted mesh, and then aggregating the local quality values into a score that represents the quality of the distorted mesh with a spatial pooling method. Many scholars have conducted excellent research works on the evaluation of the local quality map [1–6]. However, the investigation on the spatial pooling method for the MVQ metric is much less. The study in Chapter 3 concluded that the severely distorted regions have a great impact on the mesh quality. Based on this conclusion, we investigate the visual importance model for mesh quality based on local distortion in this chapter. Inspired by the percentile weighting strategy [19] proposed in the image visual quality metric, we propose a spatial pooling method [21] with the percentile weighting strategy in the mesh visual quality metric. We emphasize the impact of the severely distorted regions on the mesh quality by assigning a greater weight to a portion of vertices with large local distortions at the stage of spatial pooling. We propose the TPDMPW metric [21] by applying the percentile weighting method to aggregate the local distortions.

Before applying the percentile weighting strategy in the mesh visual quality metric, we need to consider the differences between the image visual quality metric and the mesh visual quality metric. The exploration of the percentile weighting strategy in the mesh quality assessment turns out to be more complex than in the image quality assessment. Firstly, the objects under assessment are different. An image contains pixels that are distributed uniformly in the two-dimensional domain while a mesh contains vertices that are typically distributed non-uniformly and irregularly in the three-dimensional domain. Secondly, when the percentile weighting method [19] was incorporated in the Structural Similarity Index (SSIM) metric [7], the performance improvement was intuitive since the SSIM metric used a simple average pooling strategy without any emphasis on the severely distorted regions on the image. However, in the mesh quality assessment, several MVQ metrics, such as the TPDM metric [5], adopted the Minkowski pooling strategy which has certain capability to emphasize the impact of the severely distorted regions on the mesh quality. It is more challenging to explore whether the percentile weighting strategy achieves superior performance over the Minkowski pooling strategy or not. Thirdly, the authors in literature [19] aggregated the local similarities of the image while we attempt to aggregate the local distortions of the mesh at the stage of spatial pooling. We use the TPDM metric [5] to generate the local distortions of the vertices, and then pool the local distortions into the quality score with the percentile weighting method. Lastly, in literature [19], the authors recommended the optimal values for the parameters of the percentile weighting method on one single image database [96]. If there is indeed performance improvement by incorporating the percentile weighting method into the TPDM metric, we investigate how the parameters of the percentile weighting method influence the performance of the metric on three mesh databases [1, 3, 40]. We explore the optimal values of the parameters for each mesh database and the unified values of the parameters for three mesh databases.

The remainder of this chapter is organized as follows: In Section 4.2, We introduce the spatial pooling method with the percentile weighting strategy, and propose the TPDMPW metric by applying the percentile weighting method to aggregate the local distortions. We demonstrate the effectiveness of the percentile weighting method by comparing the percentile weighting method with the average method and the Minkowski method. In Section 4.3, we evaluate the performance of the TPDMPW metric and determine the optimal values and unified values for the parameters of the percentile weighting method through empirical tests on three publicly available databases. In Section 4.4, we summarize this chapter.

4.2 Mesh Visual Quality Metric with the Percentile Weighting Strategy

4.2.1 Percentile Weighting Method

In the community of image quality assessment, Moorthy and Bovik proposed a spatial pooling method based on the percentile weighting strategy [19]. In this section, we propose a percentile weighting method in mesh quality assessment, and use the percentile weighting method to aggregate the local distortions of the mesh at the stage of spatial pooling. We propose the TPDMPW metric [21] based on the percentile weighting method. Since the pixels are distributed uniformly across the image, the local surface area of the pixel is not involved in the percentile weighting method in image quality assessment. However, the vertices are typically distributed non-uniformly and irregularly on the surface of the mesh. The local surface area of the vertex may have an impact on the mesh quality. Therefore, we involve the local surface area of the vertex in the design of the percentile weighting method in mesh quality assessment. We weight the local distortion by the local surface area for each vertex at the stage of spatial pooling in order to reflect the impact of the local surface area on the overall quality of the mesh.

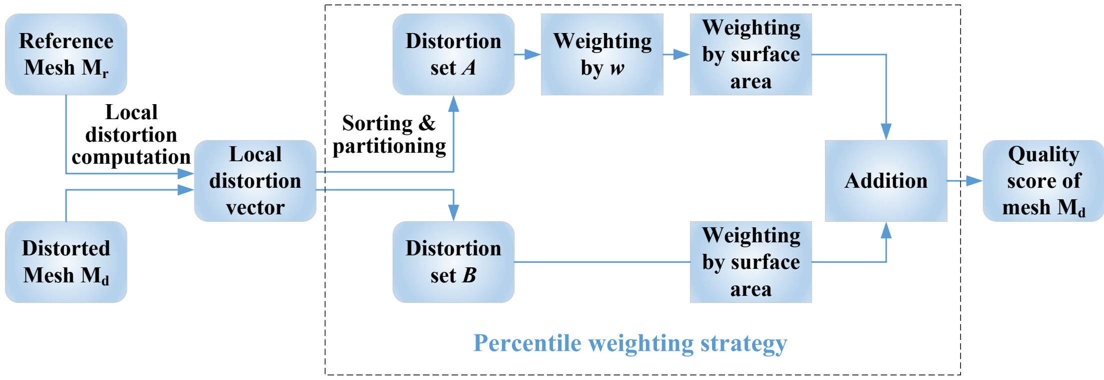


Figure 4.1 : The flowchart of the TPDMPW metric based on the percentile weighting method.

Fig. 4.1 illustrates the flowchart of the TPDMPW metric based on the percentile weighting method. We compute the local perceptual distances between the vertices of the reference mesh and the vertices of the distorted mesh via Eq. (3.3) as the local distortions of the mesh. Then we pool the local distortions with our proposed percentile weighting method to generate the quality score for the distorted mesh. A higher score indicates a poorer quality of the distorted mesh. After computing the local perceptual distances of the vertices via Eq. (3.3), we construct a local distortion vector L from all the local perceptual distances of the vertices. We sort the local distortion vector in the descending order and generate a sorted local distortion vector D. The vector D is then partitioned into two distortion sets: A and B, where the distortion set A contains the local distortion elements that rank within the top p percentile of vector D while the distortion set B contains the local distortion elements that rank beyond the top p percentile of the vector D. Fig. 4.2 illustrates the process of extracting two distortion sets from the local distortion vector L. Note that the percentile weighting method in image quality assessent [19] sorts the local similarity vector in the ascending order and then assigns a greater weight to the similarity elements that rank within the top p percentile of the sorted similarity vector, while our percentile weighting method sorts the distortion set in the descending order and then assigns a greater weight to the distortion elements that rank within the top p percentile of the sorted distortion vector. Thus, the regions with severe local distortions have a greater impact on the overall quality of the mesh.

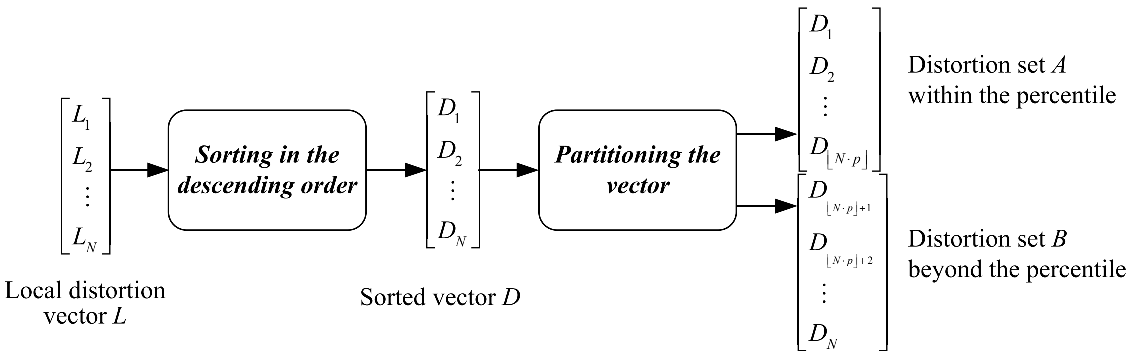


Figure 4.2 : The process of extracting two distortion sets from the local distortion vector.

Among state-of-the-art MVQ metrics, there is not a clear consensus yet on whether or not to weight the local distortion by local surface area of the vertex at the stage of spatial pooling, in either the average method or Minkowski method. The FMPD [4] and TPDM [5] metrics use the local surface area to weight the local distortion while the MSDM [1], MSDM2 [2], DAME [3] and Dong [6] metrics do not weight the local distortion by the local surface area. Whether using the local surface area to weight the local distortion is determined empirically in respective algorithm design. In this section, we use the local surface area to weight the local distortion for each vertex in the percentile weighting method and employ a weighting exponent to adjust the influence of the local surface area of the vertex on the overall quality of the mesh. We investigate the impact of the local surface area on the performance of mesh quality assessment. We build a visual importance model based on the local distortion:

$$PW = \sum_{d_i \in A} w \left(\frac{NS_i}{S}\right)^{\alpha} d_i + \sum_{d_j \in B} \left(\frac{NS_j}{S}\right)^{\alpha} d_j, \tag{4.1}$$

where w ($w \gg 1$) is the weight for the distortion elements in the distortion set A, d_i ($0 \leq d_i \leq 1$) represents the local distortion value of vertex v_i in the distortion set A while d_j ($0 \leq d_j \leq 1$) represents the local distortion value of vertex v_j in the distortion set B. N is the number of vertices in the reference mesh. S_i and S_j are the local surface areas of the vertices v_i and v_j on the reference mesh respectively, and S is the overall surface area of reference mesh. NS_i/S and NS_j/S are the scaled local surface areas. Compared with the local surface area $S_i(S_j)$, the scaled local surface areas eliminate the difference in the number of vertices between different meshes. The parameter α is the weighting exponent that regulates the influence of the local surface area on the overall quality of the mesh. In the visual importance model represented by Eq. (4.1), the distortion elements in the distortion set A are overall greater than the distortion elements in the distortion set B. The importance of the regions with severe local distortions to the mesh quality will be highlighted by using the parameter w to weight the the distortion elements in the distortion set A. In Eq. (4.1), the first part $\sum_{d_i \in A} w(Ns_i/s)^{\alpha} d_i$ typically dominates the result of the addition since the weight w is always set to be far greater than 1 and the local distortion value d_i is always greater than local distortion value d_j. The value of PW computed by Eq. (4.1) is highly correlated with the weight w and the number of the distortion elements in the distortion set A. We normalize PW with N, p and w, and finally obtain the normalized global distortion as the quality score of the distorted mesh:

$$TPDMPW = \frac{1}{Npw} \left(\sum_{d_i \in A} w \left(\frac{Ns_i}{s}\right)^{\alpha} d_i + \sum_{d_j \in B} \left(\frac{Ns_j}{s}\right)^{\alpha} d_j \right). \tag{4.2}$$

In Eq. (4.2), we have designed the metric $TPDMPW$ [21] to compute the global distance between the reference mesh and the distorted mesh based on the percentile weighting method. The adjustable parameters of the percentile weighting method are $\{p, w, \alpha\}$.

4.2.2 Comparative Analysis

In order to demonstrate the effectiveness of our proposed percentile weighting method in mesh quality assessment, we compare the percentile weighting method with the average method and Minkowski method at the stage of spatial pooling in the MVQ metric. We perform the comparison using the reference mesh and three distorted meshes of Armadillo model and Dinosaur model in the LIRIS/EPFL general-purpose database [1]. We represent three distorted meshes of Armadillo model as D_1, D_2, and D_3 and three distorted meshes of Dinosaur model as E_1, E_2, and E_3 respectively according to the rank of their MOS values. The distorted mesh D_1 is generated by adding noise to the rough regions of the reference mesh of Armadillo model, the distorted mesh D_2 is generated by applying the Taubin smoothing filter uniformly on the reference mesh of Armadillo model, and the distorted mesh D_3 is generated by applying the Taubin smoothing filter on the intermediately rough regions of the reference mesh of Armadillo model. The distorted mesh E_1 is generated by applying the Taubin smoothing filter uniformly on the reference mesh of Dinosaur model, the distorted mesh E_2 is generated by applying the Taubin smoothing filter on the intermediately rough regions of the reference mesh of Dinosaur model, and the distorted mesh E_3 is generated by adding noise to the intermediately rough regions of the reference mesh of Dinosaur model.

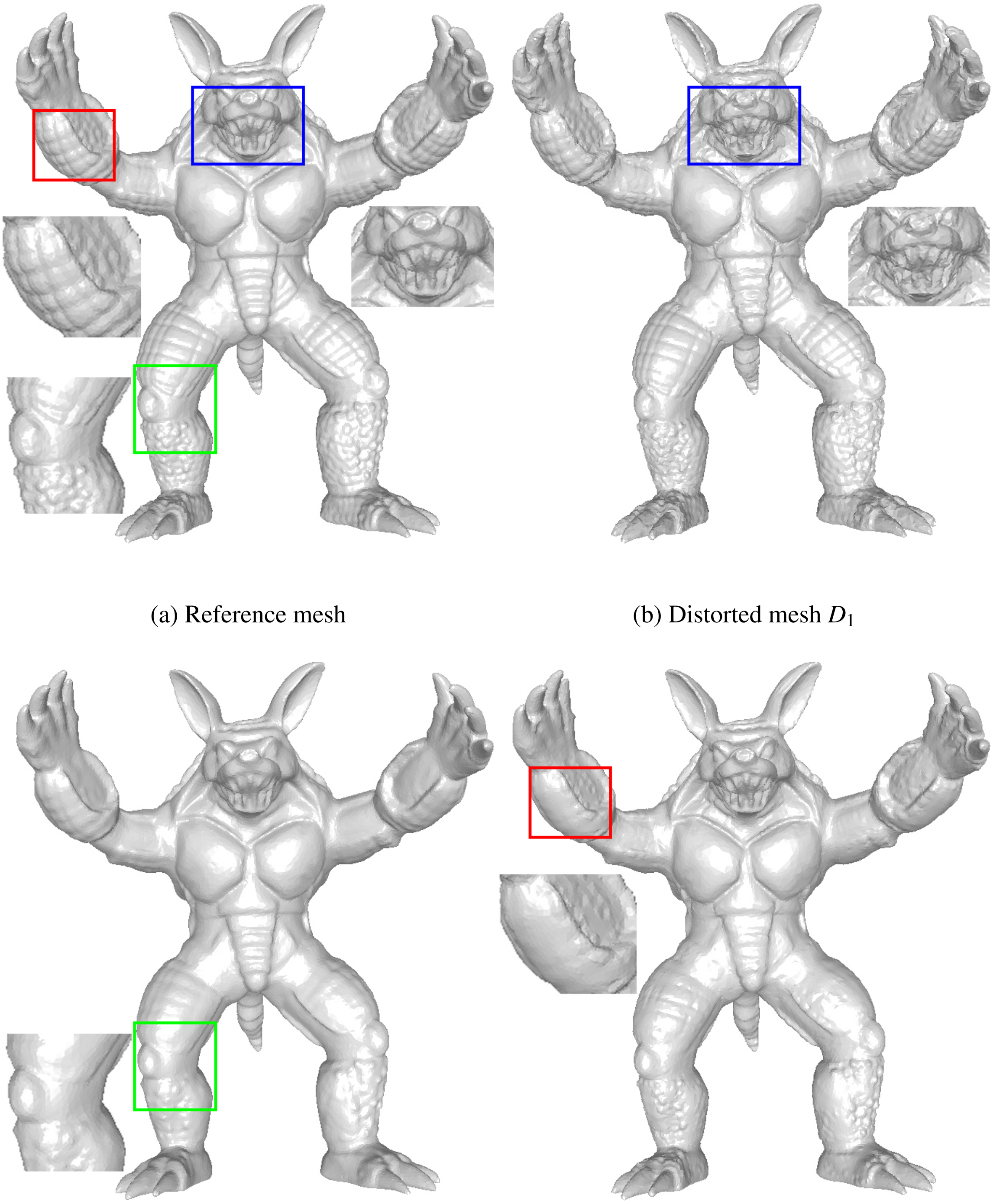
The MOS values of the distorted meshes of the Armadillo model and Dinosaur model are listed in Table 4.1. In Fig. 4.3 and Fig. 4.4, we illustrate the reference mesh and three distorted meshes of the Armadillo model and Dinosaur model, as well as the enlarged views of some representative distorted regions on the meshes as marked in the rectangles. The rough regions of the distorted mesh D_1 are noisy compared with the reference mesh of Armadillo model due to the noise addition, as illustrated in the blue rectangles of Fig. 4.3a and Fig. 4.3b. The distorted mesh D_2 is subject to the smoothing artifact

Table 4.1 : The MOS values versus the quality scores computed by the average method, the Minkowski method and the percentile weighting method for the distorted meshes of the Armadillo model and Dinosaur model.

Distorted mesh		MOS	Quality score		
			average	Minkowski	percentile weighting
Armadillo	D_1	3.3933	0.0545	0.1082	0.2805
	D_2	3.9152	0.0751	0.1233	0.3162
	D_3	3.9380	0.0543	0.1210	0.3202
Dinosaur	E_1	3.4010	0.0488	0.0823	0.2103
	E_2	3.8902	0.0350	0.0820	0.2145
	E_3	4.5824	0.0328	0.0834	0.2220

uniformly regardless of the roughness of the region on the surface, as illustrated in the green rectangles of Fig. 4.3a and Fig. 4.3c. The intermediately rough regions of the distorted mesh D_3 are subject to the smoothing artifact, as illustrated in the red rectangles of Fig. 4.3a and Fig. 4.3d. The distorted mesh E_1 is subject to the smoothing artifact uniformly regardless of the roughness of the region on the surface, as illustrated in the blue rectangles of Fig. 4.4a and Fig. 4.4b. The intermediately rough regions of the distorted mesh E_2 are subject to the smoothing artifact, as illustrated in the green rectangles of Fig. 4.4a and Fig. 4.4c. The intermediately rough regions of the distorted mesh E_3 are noisy compared with the reference mesh of Dinosaur model due to the noise addition, as illustrated in the green rectangles of Fig. 4.4a and Fig. 4.4d.

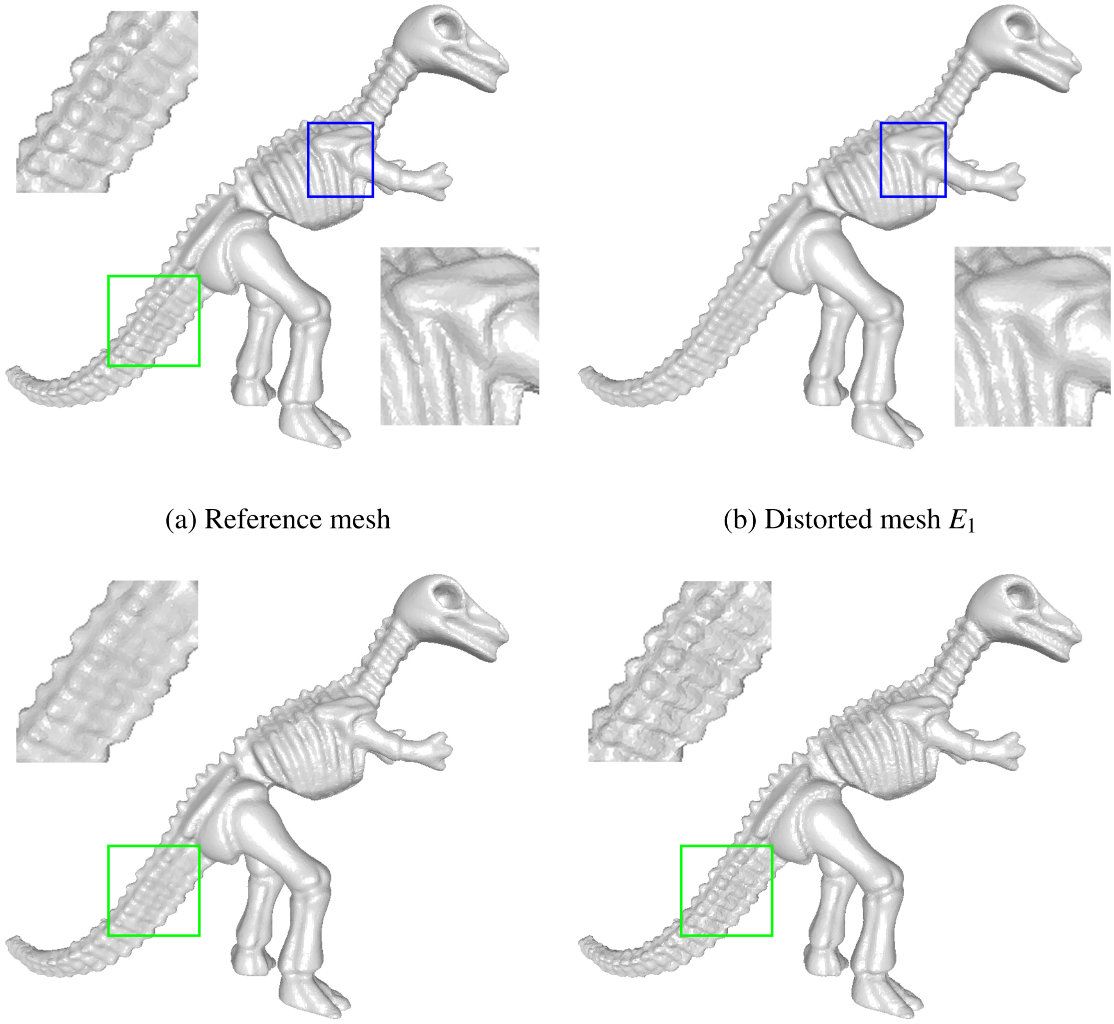
We generate a local distortion vector for each distorted mesh via Eq. (3.3) and sort the local distortion vector in the ascending order. At the stage of spatial pooling, we compute the quality scores of the distorted meshes with each of three methods: the average method, the Minkowski method and the percentile weighting method. The exponent of the Minkowski method is set as 2.5, the same value as the exponent in the literature [5].



(a) Reference mesh (b) Distorted mesh D_1

(c) Distorted mesh D_2 (d) Distorted mesh D_3

Figure 4.3 : The reference mesh and three distorted meshes of the Armadillo model, and the enlarged views of some representative distorted regions on the meshes as marked in the rectangles.



(a) Reference mesh

(b) Distorted mesh E_1

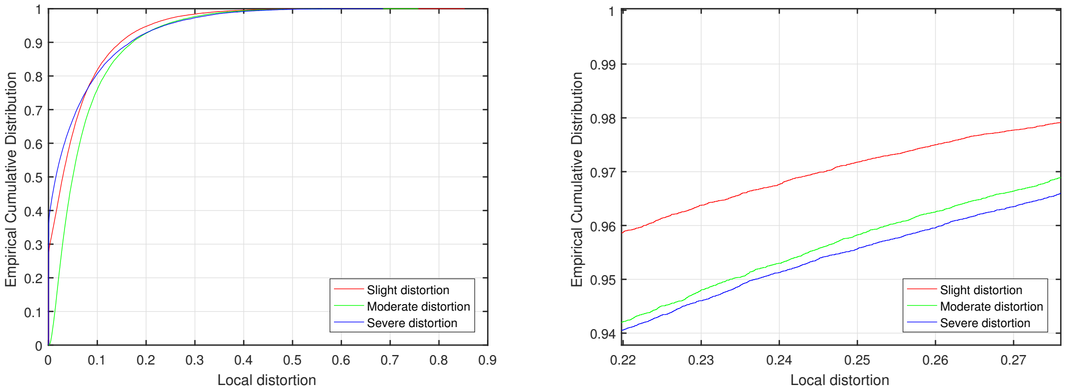
(c) Distorted mesh E_2

(d) Distorted mesh E_3

Figure 4.4 : The reference mesh and three distorted meshes of the Dinosaur model, and the enlarged views of some representative distorted regions on the meshes as marked in the rectangles.

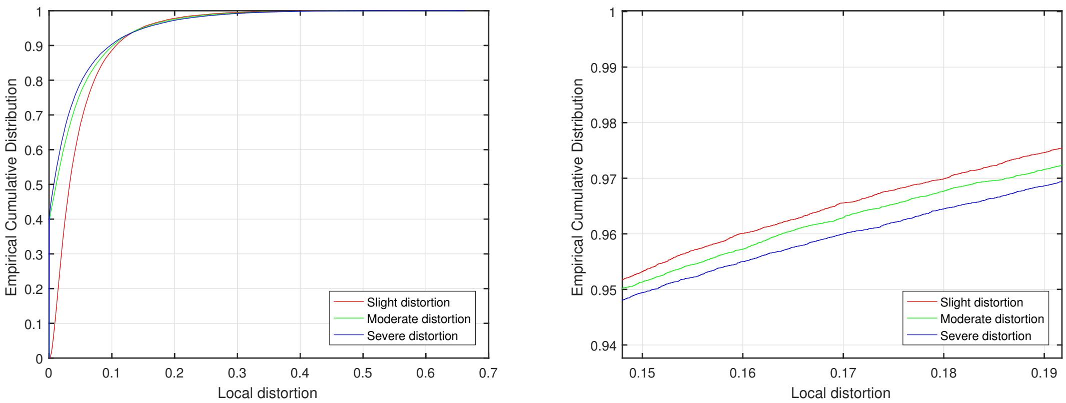
The parameters of the percentile weighting method are set as $p = 0.06$, $w = 60$, $\alpha = 0$. When comparing the three methods at the stage of spatial pooling, we do not use the local surface area to weight the local distortion. Fig. 4.5 and Fig. 4.6 illustrate the empirical cumulative distribution (ECD) of the local distortions for the distorted meshes of Armadillo model and Dinosaur model respectively. The slight distortion, moderate distortion and severe distortion in Fig. 4.5a and Fig. 4.5b correspond to the distorted mesh D_1, D_2 and

D_3 of Armadillo model respectively. The slight distortion, moderate distortion and severe distortion in Fig. 4.6a and Fig. 4.6b correspond to the distorted mesh E_1, E_2 and E_3 of Dinosaur model respectively. First, we look at the range [0.0, 0.7] on the vertical axis in Fig. 4.5a, where the local distortion values of each distorted mesh of Armadillo model rank within top 70% of the ascending-ordered distortion vector. In this range, the local distortion values for three distorted meshes of Armadillo model lie within the range [0.0, 0.1]. We observe that in this range, the local distortion values of both distorted meshes D_1 and D_2 are generally greater than the local distortion values of the distorted mesh D_3. However, when we look at the range [0.94, 1.0] on the vertical axis in Fig. 4.5b, the comparison result is different. In this range, the local distortion values for three distorted meshes of Armadillo model lie within the range [0.22, 0.28]. We observe that in this range the local distortion values of the distorted mesh D_1 are generally smaller than the local distortion values of the distorted mesh D_2, and the local distortion values of the distorted mesh D_2 are generally smaller than the local distortion values of the distorted mesh D_3. A similar reversal phenomenon can be observed for the distorted meshes of Dinosaur model in Fig. 4.6. In Fig. 4.6a, in the range [0.0, 0.7] on the vertical axis where the local distortion values of each distorted mesh of Dinosaur model rank within top 70% of the ascending-ordered distortion vector, the local distortion values of the distorted mesh E_1 are generally greater than the local distortion values of the distorted mesh E_2, and the local distortion values of the distorted mesh E_2 are generally greater than the local distortion values of the distorted mesh E_3. But in Fig. 4.6b, in the range [0.94, 1.0] on the vertical axis, the local distortion values of the distorted mesh E_1 are generally smaller than the local distortion values of the distorted mesh E_2, and the local distortion values of the distorted mesh E_2 are generally smaller than the local distortion values of the distorted mesh E_3. Since the human eyes usually pay more attention to the severely distorted regions when one attempts to judge the visual quality of a distorted mesh, the regions with severe local distortions intrinsically have a greater influence on the subjective judgment than the regions with slight local distortions. If we pool the local distortions simply in an average manner without putting any emphasis on the regions with severe local distortions, the overall quality of the distorted mesh may be overestimated.



(a) (b)

Figure 4.5 : The empirical cumulative distribution of the local distortions for the distorted meshes of the Armadillo model. (a) ECD values in the range [0.0, 1.0]. (b) Close-up of subfigure (a) for the ECD values in the range [0.94, 1.0].



(a) (b)

Figure 4.6 : The empirical cumulative distribution of the local distortions for the distorted meshes of the Dinosaur model. (a) ECD values in the range [0.0, 1.0]. (b) Close-up of subfigure (a) for the ECD values in the range [0.94, 1.0].

After generating the local distortion vector of the distorted mesh via Eq. (3.3), we use the average method, the Minkowski method and the percentile weighting method to compute the quality scores of the distorted meshes for Armadillo model and Dinosaur model at the stage of spatial pooling respectively. In Table 4.1, we list the quality scores of the distorted meshes of Armadillo model and Dinosaur model that are computed by using three methods. Note that a higher MOS value in the LIRIS/EPFL general-purpose database [1] indicates a poorer quality of the distorted mesh, and a higher quality score computed by using each of three methods also indicates a poorer quality of the distorted mesh. A well-performing MVQ metric is expected to generate the quality scores that keep consistent with the subjective opinion scores for the distorted meshes as much as possible. A higher consistency between the quality scores and subjective opinion scores indicates a better performance of mesh quality assessment for the MVQ metric, which means that a higher MOS value should pair with a higher quality score for the distorted mesh. From Table 4.1, we observe that the average method leads to contradictory quality scores with MOS values for three distorted meshes of either Armadillo model or Dinosaur model. The Minkowski method leads to contradictory quality scores with MOS values for the distorted meshes D_2 and D_3 of Armadillo model, and contradictory quality scores with MOS values for the distorted meshes E_1 and E_2 of Dinosaur model. The percentile weighting method results in completely consistent quality scores with MOS values for three distorted meshes of both Armadillo model and Dinosaur model. Thus, we conclude that, compared with the average method and the Minkowski method, the percentile weighting method has a stronger capability to emphasize the impact of the local regions with severe distortions on the overall quality of the mesh. The quality scores computed by the percentile weighting method have a higher correlation with the MOS values of the distorted meshes, which indicates that the percentile weighting method has a better performance for mesh quality assessment.

4.3 Experimental Results and Analysis

4.3.1 Experiment Protocol

In this section, we evaluate the performance of the TPDMPW metric on three publicly available databases: LIRIS/EPFL general-purpose database [1], LIRIS masking database [40] and UWB compression database [3]. We compare the performance of the TPDMPW metric with state-of-the-art MVQ metrics.

- The LIRIS/EPFL general-purpose database [1] contains 4 reference meshes and 84 distorted meshes. Each 3D geometric model corresponds to one reference mesh and 21 distorted meshes. The distorted meshes were generated by applying artificial distortions on the reference mesh. Two types of distortions, noise addition and smoothing, were applied with three noticeable and distinguishable strengths either locally or globally on the reference mesh. The distorted regions are classified into four categories: smooth regions, rough regions, intermediate rough regions, and global surface. Twelve subjects participated in the subjective rating experiment. Each subject rated the visual quality of all the meshes and provided a subjective opinion score for each mesh. The opinion scores lie in the range [0, 10], where 0 indicates the best quality for the mesh while 10 indicates the worst quality. After collecting the opinion scores of all the observers, the authors in literature [1] validated the effectiveness of the opinion scores across the observers through a statistical analysis. They calculated the mean value of the opinion scores of all the subjects for each mesh and then normalized the opinion scores. The mean value of the normalized opinion scores is used as the mean opinion score for the mesh. The average duration for rating the visual quality of each mesh by each subject is 20 seconds.

- The LIRIS masking database [40] contains 4 reference meshes and 24 distorted meshes. Each 3D geometric model corresponds to one reference mesh and 6 distorted meshes. The distorted meshes were generated by adding random noise with different intensities to the vertices on the smooth or rough regions. The database was constructed with the original purpose of evaluating the capability of the MVQ metric in capturing the visual masking effect. Each subject rated the visual quality of all the distorted meshes and provided a subjective opinion score for each distorted mesh. The opinion scores lie in the range [0, 4], where 0 indicates the worst quality for the distorted mesh while 4 indicates the best quality. Eleven subjects participated in the subjective rating experiment. Each subject completed the subjective rating for all the distorted meshes of the models within 3 minutes.

- The UWB compression database [3] contains 5 reference meshes and 63 distorted meshes. Each 3D geometric model corresponds to one reference mesh and 12 or 13 distorted meshes. The distorted meshes were generated by applying 13 types of artificial distortions on the reference meshes. The opinion score of each mesh was calculated based on the selected preferences for the meshes of each model. The opinion scores of the meshes lie in the range [0, 1], where 0 indicates the best quality for the mesh while 1 indicates the worst quality. Sixty-nine subjects participated in the subjective rating experiment. Each subject rated the visual quality of 40 pairs of distorted meshes and selected the preferences for each pair. It took 15 minutes in average for each subject to complete the preference selection for 40 pairs of distorted meshes. The mean opinion score of each mesh was estimated via a designated calculation process with the selected preferences of all the meshes for each model. There is a quality ranking among the MOSs of different meshes of the same model, but there is no coherence between the MOSs of the meshes of different models. Hence, the overall performance of mesh qaulity assessment on the UWB compression database [3] is typically derived by firstly computing the performance value with the meshes of each individual model and then averaging the performance values of 5 models, as suggested in literature [3, 5, 45].

As described in literature [1, 3, 40], the distorted meshes in the three databases were generated by applying artificial distortions with appropriate strengths to the reference meshes. The strengths of distortions were chosen carefully to reflect the visible distortions that occur in the common mesh processing operations, and were adapted to the surface resolutions of the reference meshes. The applied distortions, such as noise addition and smoothing, in the three databases do not change the number of vertices in the mesh. However, the TPDMPW metric is applicable to not only the case where the number of vertices is fixed but also the case where either the number of vertices or the topological connectivity of the mesh is changed since the TPDMPW metric inherits the capability of the TPDM metric [5] in addressing the issue of changing connectivity in the mesh. In the subjective rating experiment of each database, the interactions, such as rotation, scaling and translation, were allowed for the subjects so that they could observe the meshes from different viewpoints. Thus, the MOSs of the meshes reflect the influence of the viewpoints on the perception of mesh quality.

We use the TPDMPW metric to compute the quality scores for all the distorted meshes in three databases, and evaluate the performance of the TPDMPW metric with two coefficients: Pearson linear correlation coefficient (PLCC) and Spearman rank-order correlation coefficient (SROCC). In literature [1–6], the scholars adopted the absolute values of PLCC and SROCC to compare the performance of different MVQ metrics. In this section, we also use the absolute values of PLCC and SROCC to compare the performance between the TPDMPW metric and state-of-the-art MVQ metrics for the purpose of intuitive description. Note that the values of PLCC and SROCC computed by the MVQ metric on the UWB compression database are typically negative. The operation of taking the absolute value will flip the sign of the values of PLCC and SROCC on the UWB compression database, but it does not affect the performance comparison. The three databases used different ranges for the MOSs of the meshes. In our experiments, we normalize the MOSs of the meshes in each database into the range [0, 1], and then perform a psychometric curve fitting between the MOS values and quality scores of the meshes. We compute the correlation coefficients between the MOS values and the mapped quality scores of the meshes as the performance indicators of the TPDMPW metric.

It is recommended to perform a psychometric fitting to remove the nonlinearity between the quality scores and MOS values of the meshes [1], [4–6, 86] before computing the correlation coefficients, especially the PLCC coefficient. We apply the cumulative Gaussian psychometric function [97] for the fitting:

$$g(a,b,S) = \frac{1}{\sqrt{2\pi}} \int_{a+bS}^{+\infty} e^{-(t^2/2)} dt, \tag{4.3}$$

where S is the quality score of the mesh. The parameters a and b are obtained by minimizing the sum of squared differences between the mapped quality scores and the MOS values through the non-linear least squares fitting method.

4.3.2 Performance Comparison

We compare the performance of the TPDMPW metric with the performance of state-of-the-art MVQ metrics, including Hausdorff Distance (HD) [48, 49], Root Mean Square Error (RMS) [48, 49], Geometric Laplacian (GL1, GL2) [43, 44], Strain Field (SF) [61], 3DWPM1 [60], 3DWPM2 [60], MSDM [1], MSDM2 [2], FMPD [4], DAME [3], TPDM [5], Dong [6]. We obtain the results of state-of-the-art MVQ metrics in Table 4.2 from the literature [3–5, 45, 46] and the erratum of MVQ metrics [98]. The performance values of HD and RMS were generated by using the widely used Metro tool [48, 49]. We generate the performance values for the TPDM metric with the source code publicly released online [5], which are officially confirmed by the authors. Table 4.2 lists the values of PLCC and SROCC for the TPDMPW metric and state-of-the-art MVQ metrics on each of three databases. We provide the performance values of the TPDMPW metric in two cases: TPDMPW(u) and TPDMPW(o). The performance values of TPDMPW(u) are generated by setting the unified values to the parameters of the percentile weighting method on three databases while the performance values of TPDMPW(o) are generated by setting the optimal values to the parameters of the percentile weighting method on each database.

Table 4.2 : The PLCC and SROCC values (%) of the TPDMPW metric and state-of-the-art MVQ metrics on each of three databases.

Metrics	General-Purpose		Masking		Compression	
	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
HD	11.4	13.8	20.2	26.6	14.0	24.5
RMS	28.1	26.8	41.2	48.8	49.0	52.0
GL1	35.5	33.1	39.6	42.0	70.6	66.9
GL2	42.4	39.3	38.3	40.1	76.1	73.9
SF	7.0	15.7	15.5	38.6	34.8	57.4
3DWPM1	61.8	69.3	31.9	29.4	84.1	81.9
3DWPM2	49.6	49.0	42.7	37.4	82.3	80.9
MSDM	75.0	73.9	69.2	65.2	91.5	83.1
MSDM2	81.4	80.4	87.3	89.6	89.3	78.0
FMPD	83.5	81.9	80.8	80.2	88.8	81.8
DAME	75.2	76.6	58.6	68.1	93.5	85.6
TPDM	84.1	84.3	88.6	90.0	90.7	82.9
Dong	87.7	86.6	92.6	92.5	96.3	84.2
TPDMPW(u)	85.2	84.6	88.8	90.5	96.2	89.3
TPDMPW(o)	87.7	87.2	91.7	94.2	96.4	91.3

From Table 4.2, we observe that the TPDMPW metric achieves significant performance gain over the TPDM metric on each of three databases in the case of either the unified parameter values or optimal parameter values, which demonstrates the effectiveness of the percentile weighting method and the superiority of the TPDMPW metric. The comparison result also indicates that the visual importance model based on the local distortion facilitates to improve the performance of mesh quality assessment. In the case of optimal parameter values, the TPDMPW metric achieves the best performance on the LIRIS/EPFL general-purpose database and the UWB compression database among all the metrics, and comparable performance to the metric of Dong [6], which is one of the best MVQ metrics, on the LIRIS masking database.

The experiment shows that, among three parameters of the percentile weighting method, the parameter α has a greater impact on the performance of the TPDMPW metric than the parameters w and p. In both cases of unified parameter values and optimal parameter values, the values of the parameters w and p are set as: $w = 60$, $p = 0.06$. In the case of unified parameter values, the value of the parameter α is set as: $\alpha = 0.3$. In the case of optimal parameter values, there is a significant difference in the value of the parameter α on three databases. The optimal value of the parameter α is set as: $\alpha = 0$, $\alpha = 0.7$ and $\alpha = 0.55$ respectively on the LIRIS/EPFL general-purpose database, the LIRIS masking database, and the UWB compression database. This indicates that when the local surface area of the vertex is used to weight the local distortion, the impact on the performance of the TPDMPW metric on three databases is different. The reason may be that the types of distortions on three databases are different and the local surface area shows different effects on the quality of the meshes with different types of distortions. In Section 4.3.3, we will investigate the influence of the parameters α, w and p on the performance of the TPDMPW metric, and introduce how we determine the optimal values and unified values for the parameters on three databases.

We perform the psychometric fitting on each of three databases. Fig. 4.7, Fig. 4.8 and Fig. 4.9 illustrate the scatter plots of MOSs versus quality scores and the fitted psychometric curve for the meshes in each of three databases. The quality scores of the meshes in Fig. 4.7, Fig. 4.8 and Fig. 4.9 are computed by using the TPDMPW metric in the case of

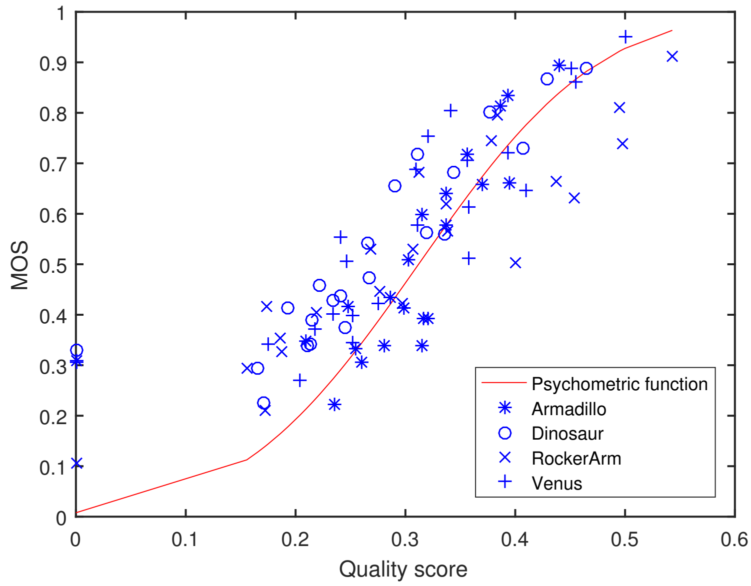


Figure 4.7 : The scatter plots of MOSs versus quality scores and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database.

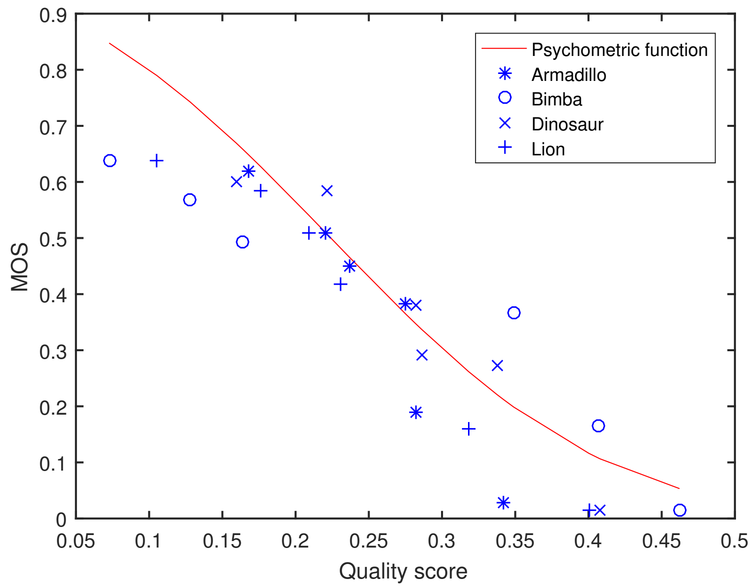
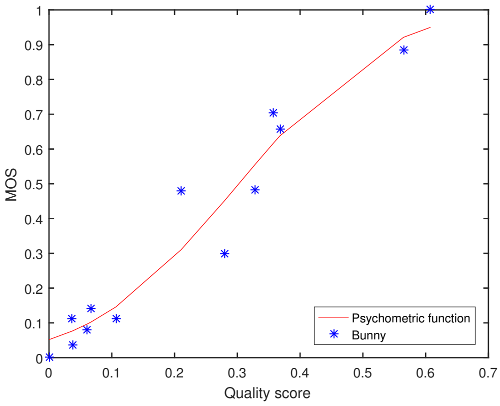
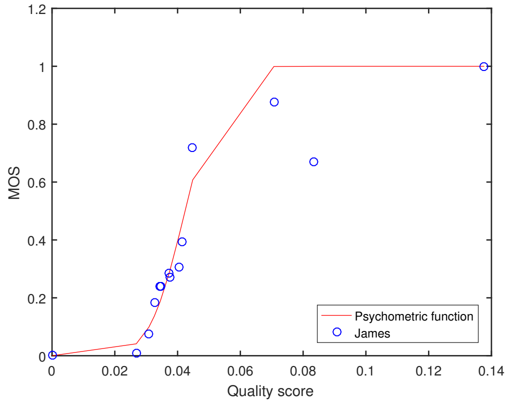


Figure 4.8 : The scatter plots of MOSs versus quality scores and the fitted psychometric curve for the meshes in the LIRIS masking database.

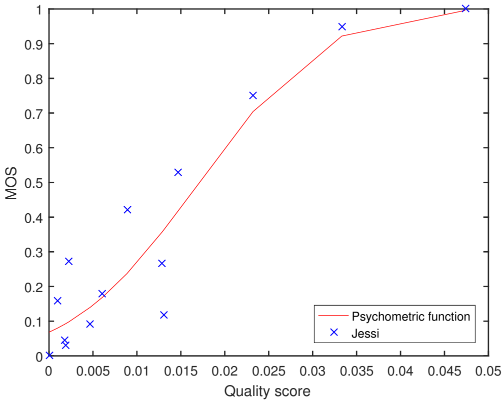
optimal parameter values. From these figures, we observe that the *Quality Score – MOS* pairs are fitted well by the psychometric fitting curve. In the subjective rating experiment



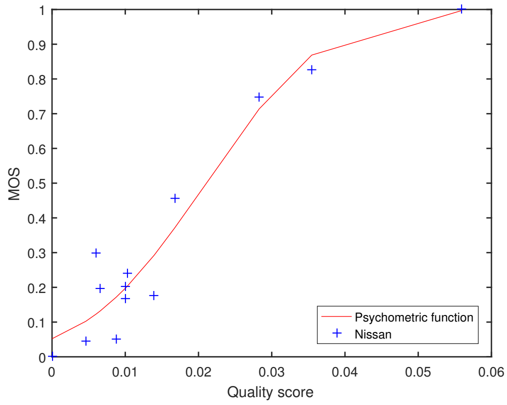
(a) Bunny



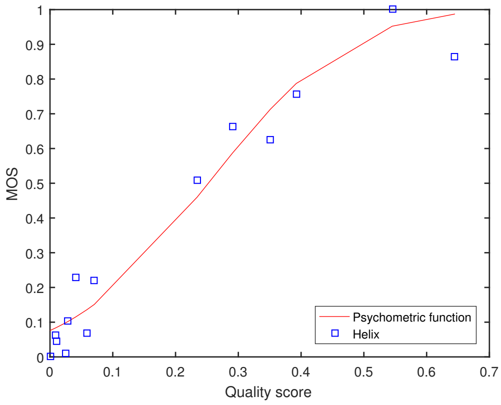
(b) James



(c) Jessi



(d) Nissan



(e) Helix

Figure 4.9 : The scatter plots of MOSs versus quality scores and the fitted psychometric curve for the meshes of each model in the UWB compression database.

of the LIRIS/EPFL general-purpose database [1], each subject rated the visual quality of all the meshes including the reference meshes and provided a score that reflects the overall degree of perceived distortion to each mesh. Due to the variation of perception among the subjects, the MOS values of four reference meshes are not equal to 0. However, the quality scores of the reference meshes computed by the full-reference MVQ metrics including the TPDMPW metric are always 0. In practical applications, the quality scores of the reference meshes are expected to be 0 since the distance between a mesh and itself is always 0. This explains why there are four outliers on the vertical axis in Fig. 4.7.

During the construction of the UWB compression database [3], the MOS values of the distorted meshes were estimated via a designated calculation process with the selected preferences from 40 random pairs of distorted meshes for each model. The MOS values reflect the quality ranking of the distorted meshes for each model but do not reflect the degree of perceived distortion of each distorted mesh accurately. From Fig. 4.9, we observe that the $Quality\ Score - MOS$ pairs are generally fitted well by the psychometric fitting curve for each model.

As mentioned in literature [1, 3, 40], the three databases used in this section cover several types of distortions that may occur in mesh processing operations, including mesh simplification, mesh compression, and mesh watermarking. The TPDMPW metric is capable of addressing the issue of changing the topological connectivity or the number of vertices in the mesh, and thus can be used to evaluate the quality of the meshes that go through the common processing operations, such as simplification, compression, and watermarking.

4.3.3 Parameters of the Percentile Weighting Method

The percentile weighting method has three adjustable parameters: the local surface area weighting exponent α, the weight w and the percentile p. The investigation on these parameters incurs three questions: (1) how much should the contribution of local surface area be ? (2) how much should we weight the local distortions within the percentile ? (3) what percentile should be used ? In order to answer these questions, we investigate the impact of three parameters on the performance of the TPDMPW metric through empirical tests. We change the values of three parameters and then observe how the SROCC performance of the TPDMPW metric varies as a function of these parameters. Since the parameter α is related to the local surface area of the vertex while the parameters w and p are related to the local distortions within the percentile, we simplify the questions and assume that the impact of the parameter α is independent on the impact of the parameters w and p.

4.3.3.1 Parameter α

We firstly change the value of the parameter α while fixing the values of the parameters w and p as $w = 60$, $p = 0.06$. Each time we change the value of the parameter α, we compute the SROCC value of the TPDMPW metric on each database. We take the SROCC value of the TPDMPW metric as a function of the parameter α. Fig. 4.10 illustrates how the SROCC value of the TPDMPW metric varies on each of three databases as the value of the parameter α changes. The horizontal axis in Fig. 4.10 indicates that the value of the parameter α varies within the range [0, 1] with a step-size of 0.05. The SROCC value is shown on the vertical axis. From Fig. 4.10, we observe that the SROCC value reaches to the highest point at different values of the parameter α for three databases. The optimal values of the parameter α that achieve the highest SROCC values are 0 for LIRIS/EPFL general-purpose database, 0.7 for LIRIS masking database and 0.55 for UWB compression database respectively. When the value of the parameter α is equal to 0.3, the SROCC values are overall high on three databases. Thus, we take 0.3 as the unified value of the parameter α on three databases, and take 0, 0.7, 0.55 as the optimal values of the parameter α on each database.

From Fig. 4.10, we observe that there is a significant difference in the influence of the parameter α on the SROCC value between the LIRIS/EPFL general-purpose database and the other two databases. Since the parameter α adjusts the influence of the local surface

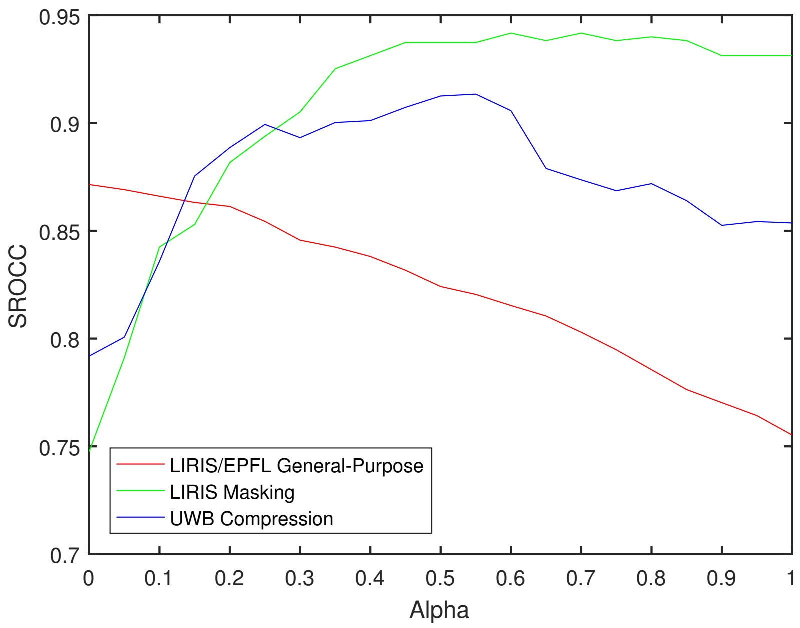
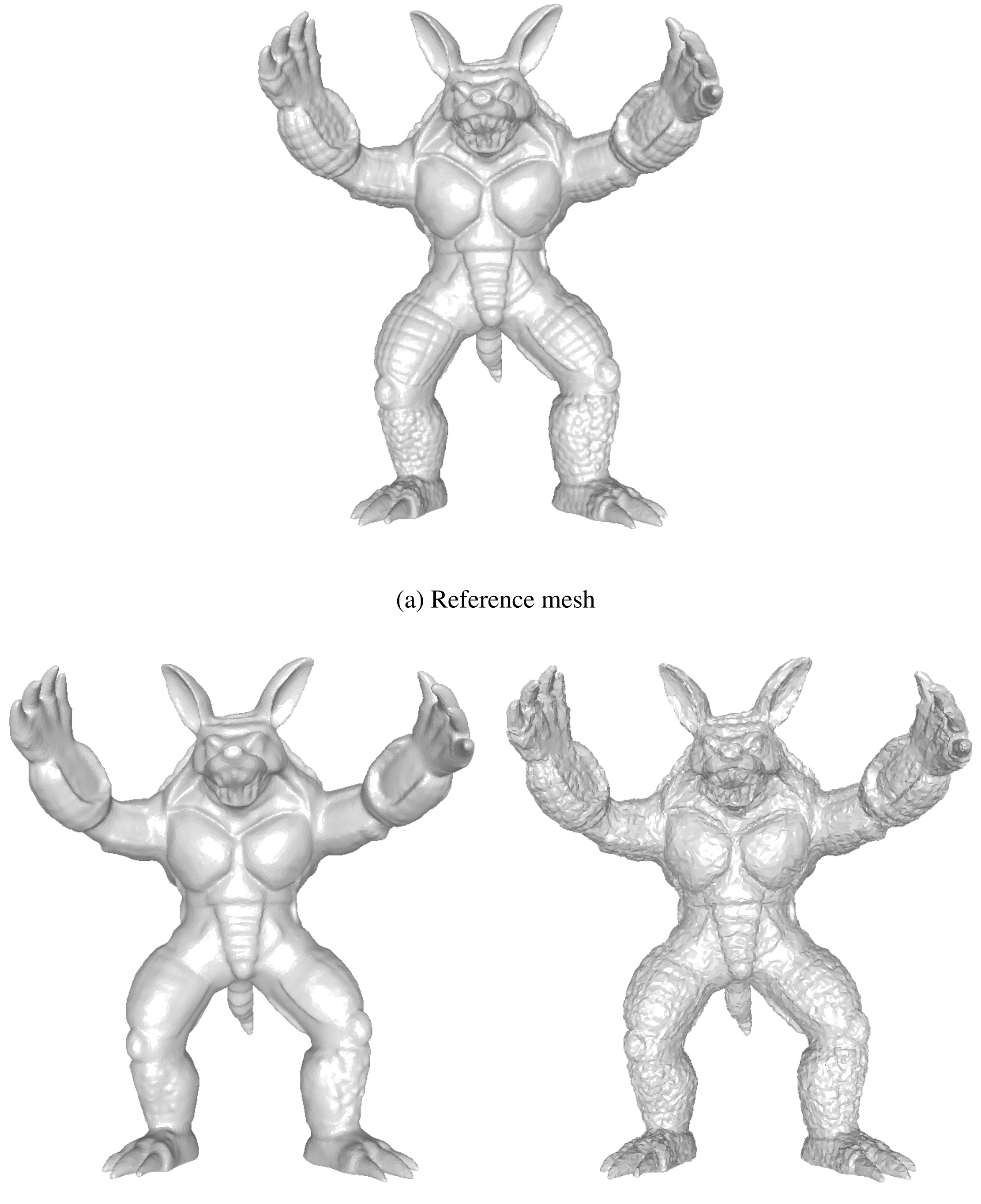


Figure 4.10 : The plot of the SROCC value as a function of the parameter α on each of three databases.

Table 4.3 : The MOS values and the quality scores of two distorted meshes of the Armadillo model in the LIRIS/EPFL general-purpose database.

Score	Smoothing	Noise addition
MOS	6.5929	8.3354
Quality score, α=0	0.3697	0.3927
Quality score, α=1	0.3934	0.3899

area on the overall quality of the mesh, we analyze the influence of local surface area on the performance of the TPDMPW metric in depth on the LIRIS/EPFL general-purpose database. The LIRIS/EPFL general-purpose database involves two types of distortions: smoothing and noise addition. We perform the analysis with three meshes of Armadillo model in the LIRIS/EPFL general-purpose database: the reference mesh, a distorted mesh with the smoothing distortion and a distorted mesh with the noise addition distortion. The smoothing distortion was implemented by applying the Taubin smoothing filter uniformly



(a) Reference mesh

(b) The distorted mesh with smoothing

(c) The distorted mesh with noise addition

Figure 4.11 : The reference mesh and two distorted meshes of the Armadillo model.

to the geometric coordinates of the vertices in the reference mesh. The noise addition distortion was implemented by adding random noise uniformly to the geometric coordinates of the vertices in the reference mesh. The reference mesh and two distorted meshes of the

Armadillo model are illustrated in Fig. 4.11.

We use the TPDMPW metric to compute the quality scores of two distorted meshes in the cases of $\alpha = 0$ and $\alpha = 1$ respectively. The case of $\alpha = 0$ indicates that the impact of the local surface area is not considered in the computation of the quality score while the case of $\alpha = 1$ indicates that the impact of the local surface area is fully considered in the computation of the quality score. The MOS values and quality scores of two distorted meshes of the Armadillo model are listed in Table 4.3. The quality scores are computed by setting $p = 0.06$ and $w = 60$. From Table 4.3, we observe that, in the case of $\alpha = 0$, the quality scores of two distorted meshes have a consistent ranking with the MOS values of two distorted meshes. However, in the case of $\alpha = 1$, the quality scores of two distorted meshes have a contradictory ranking with the MOS values. This indicates that using the local surface area to weight the local distortion would lead to a negative effect on the quality ranking of the meshes, and thus cause a degradation to the performance of the TPDMPW metric on the LIRIS/EPFL general-purpose database. It is inappropriate to include the local surface area in the TPDMPW metric on the LIRIS/EPFL general-purpose database.

In order to explore the reason of the contradiction between the quality scores and the MOS values in the case of $\alpha = 1$, we further investigate the impact of local surface area on two types of distortions: smoothing and noise addition. We calculate the local roughness for each vertex of the reference mesh with the method in literature [40] and analyze the effect for the rough regions and smooth regions when weighting the local distortion by the local surface area. Fig. 4.12 illustrates the empirical cumulative distribution of local roughness of the vertex on the reference mesh of Armadillo model. The intersecting dashed lines indicate that the values of local roughness that are above 46.1 fall within the top 0.06 percentile of the descending-ordered roughness vector. We observe that the roughness elements within the bottom 0.06 percentile of the descending-ordered roughness vector have the value of 0. We refer to the regions that correspond to the top 0.06

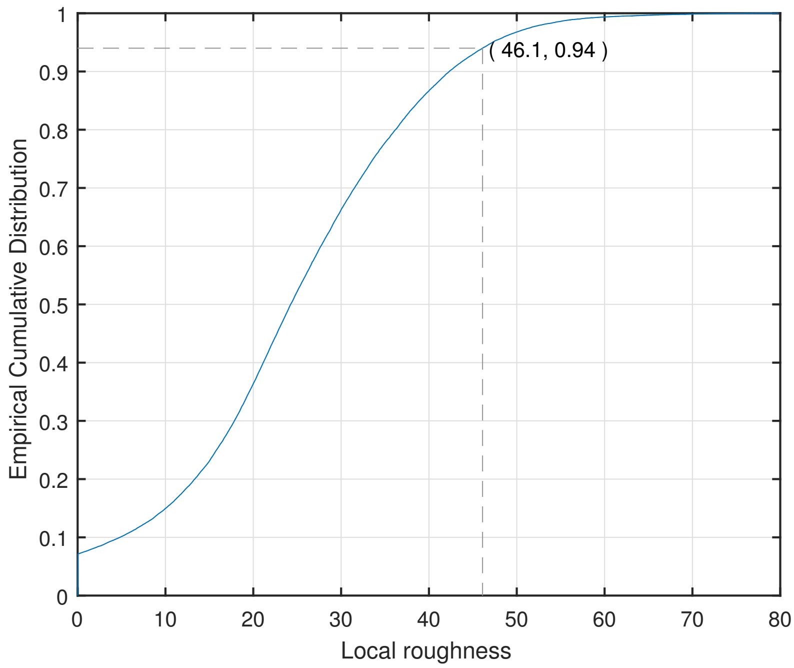


Figure 4.12 : The empirical cumulative distribution of local roughness of the vertex on the reference mesh of the Armadillo model.

percentile as the rough regions and the regions that correspond to the bottom 0.06 percentile as the smooth regions. We compute the local distortion $LTPDM_{v_i}$ for each vertex of the reference mesh under each type of distortion and the local surface area Ns_i/s for each vertex of the reference mesh. Then, on both the rough regions and smooth regions in the reference mesh, we compute the mean value of local distortions for each type of distortion as well as the mean value of local surface areas.

Table 4.4 lists the mean value of the local distortions on the rough regions and smooth regions for two distorted meshes, and the mean value of the local surface areas on the rough regions and smooth regions of the reference mesh. From Table 4.4, we observe that for the distorted mesh with the smoothing distortion, the mean value of the local distortions on the rough regions is greater than on the smooth regions. This is easy to explain because the smoothing operation typically induces more significant visual distortions on the rough regions than on the smooth regions. We also observe that for the distorted mesh

Table 4.4 : The mean value of the local distortions on the rough regions and smooth regions for two distorted meshes, and the mean value of the local surface areas on the rough regions and smooth regions of the reference mesh.

Regions	Mean of local distortions		Mean of local surface areas
	Smoothing	Noise addition	
Rough regions	0.0871	0.1171	0.6906
Smooth regions	0.0544	0.1309	0.9617

with the noise addition distortion, the mean value of the local distortions on the rough regions is smaller than on the smooth regions. This can be explained by the visual masking effect that the rough regions can mask the visual distortions to some extent. We observe that the mean value of the local surface areas on the rough regions is smaller than on the smooth regions. The reason is that the rough regions generally need small-area triangles to characterize the highly curved surface shape while the smooth regions allow large-area triangles to characterize the relatively flat surface shape. If we use the local surface area to weight the local distortion for the smoothing distortion, the local distortions on the smooth regions may be overemphasized, which will finally lead to an overestimation of the quality degradation. This explains why the quality score of the distorted mesh with the smoothing distortion is greater than the quality score of the distorted mesh with the noise addition distortion in the case of $\alpha = 1$, as shown in Table 4.4. If we use the local surface area to weight the local distortion for the noise addition distortion, there will not be such a problem. On the contrary, the local surface area may help to emphasize the local distortions on the smooth regions under the distortion of noise addition. In the LIRIS/EPFL general-purpose database, 36 out of 84 distorted meshes are generated by applying the smoothing distortion. This explains why the SROCC value on LIRIS/EPFL general-purpose declines on the whole when the value of the parameter α increases from 0 to 1 in Fig. 4.10. In the LIRIS masking database, the type of distortion only involves the noise addition, so the SROCC value keeps relatively stable when the value of the parameter α increases from a medium value 0.4 to 1. In the UWB compression database, one out of 13 types of distortions is the smoothing distortion. This explains why there is a decreasing trend for the SROCC value on the UWB compression database after the value of the parameter α increases to be greater than the optimal value 0.55.

4.3.3.2 Parameters w and p

In Section 4.3.3.1, we investigated the influence of the parameter α on the SROCC performance of the TPDMPW metric and determined the unified value of the parameter α on three databases. In this section, we investigate the influence of the parameters w and p on the SROCC performance of the TPDMPW metric while setting the value of the parameter α to be the unified value: $\alpha = 0.3$. Each time we change the values of the parameters α and p, we compute the SROCC values of the TPDMPW metric on three databases. We plot the SROCC value of the TPDMPW metric as a function of the parameters w and p on each database in Fig. 4.13, Fig. 4.14 and Fig. 4.15. The curves in Fig. 4.13, Fig. 4.14 and Fig. 4.15 indicate that the SROCC value changes as the value of the parameter w varies when the value of the parameter p is set to some value within a given range. Our preliminary experiment shows that the SROCC values of the TPDMPW metric are relatively stable on three databases when the value of the parameter p varies within the range [0.06, 0.2] and the value of the parameter w varies within the range [20, 70]. Thus, we change the value of the parameter p within the range [0.06, 0.2] with a step-size of 0.02 and change the value of the parameter w within the range [20, 70] with a step-size of 5. As illustrated in Fig. 4.13, Fig. 4.14 and Fig. 4.15, the SROCC value on the LIRIS/EPFL general-purpose database lies within the range [0.844, 0.855], the SROCC value on the LIRIS masking database lies within the range [0.887, 0.906], and the SROCC value on the UWB compression database lies within the range [0.891, 0.898].

From Fig. 4.13, Fig. 4.14 and Fig. 4.15, we observe that the fluctuation of the SROCC

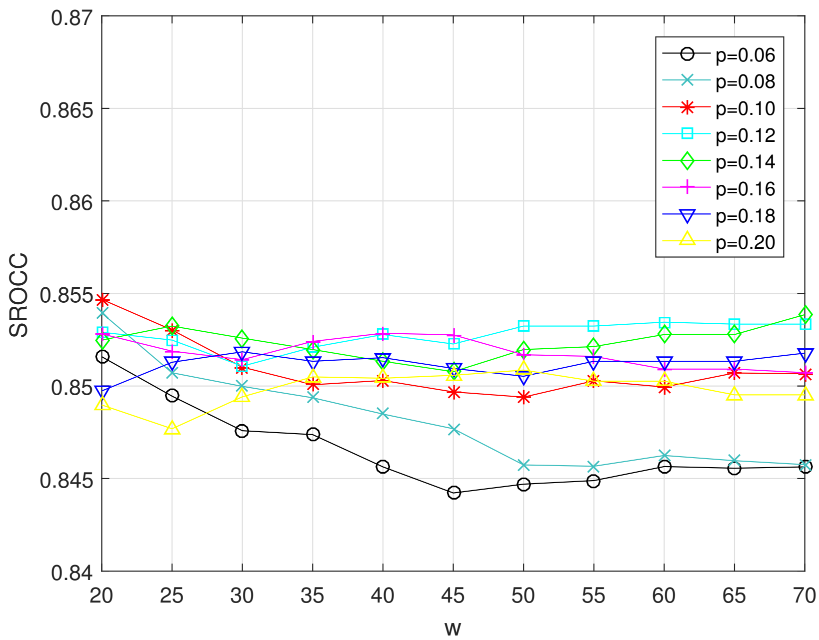


Figure 4.13 : The plot of the SROCC value as a function of the parameters *w* and *p* on the LIRIS/EPFL general-purpose database.

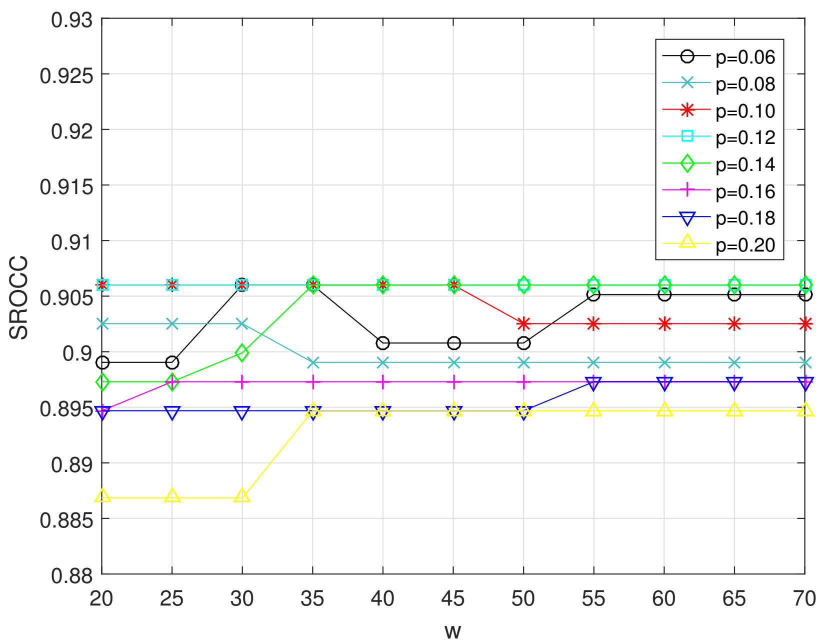


Figure 4.14 : The plot of the SROCC value as a function of the parameters *w* and *p* on the LIRIS masking database.

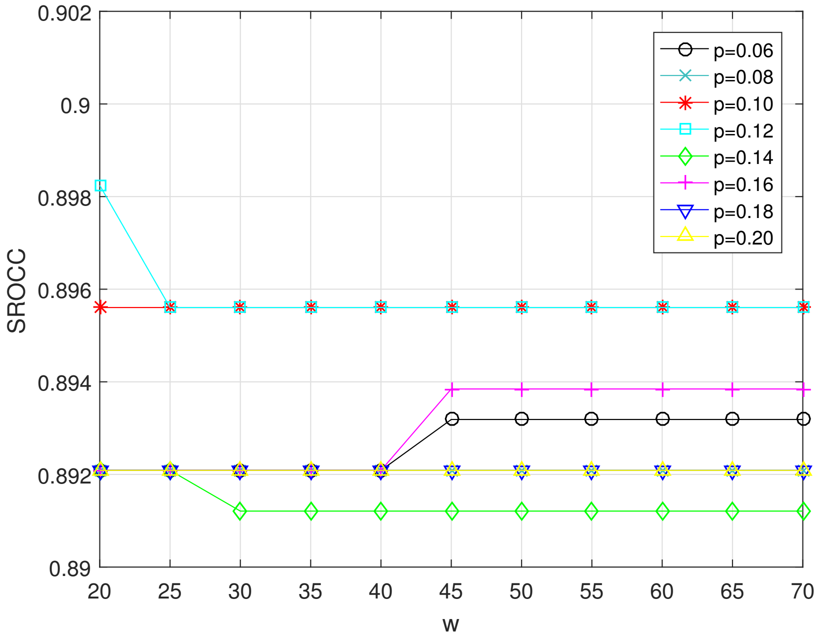


Figure 4.15 : The plot of the SROCC value as a function of the parameters w and p on the UWB compression database.

value is overall slight on each database when the value of the parameter p varies within the range [0.06, 0.20]. We choose 0.06 as the unified value of the parameter p on three databases. In the image quality assessment [19], the value of the parameter p of the percentile weighting method was also set to be 0.06. This indicates that the human eyes can judge the visual quality of the object, either 3D mesh or 2D image, confidently after observing a portion of severe local distortions within a small percentage. We observe that when the value of the parameter w increases gradually to be greater than 55, the SROCC value becomes stable on three databases. Thus, we choose 60 as the unified value of the parameter w on three databases. In the image quality assessment [19], the value of the parameter w of the percentile weighting method was set to be 4000. There is a difference in two orders of magnitude between the unified value of the parameter w in this section and the value of the parameter w in literature [19].

The reason is that in literature [19] the local similarity values were used to represent the local distances between the images, and the similarity values within the percentile are smaller than the similarity values beyond the percentile. So a large value of the weight w was needed in order to emphasize the impact of the regions with low local similarity values on the image quality. However, in the TPDMPW metric, we use the local distortion values to represent the local distances between the meshes. The local distortion values within the percentile are greater than the local distortion values beyond the percentile. A moderate value is enough for the weight w to emphasize the impact of the regions with severe local distortions on the mesh quality.

4.4 Summary

In this chapter, we have proposed a spatial pooling method for the mesh visual quality metric based on the percentile weighting strategy. We assign a greater weight to a portion of vertices with large local distortions to emphasize the impact of the severely distorted regions on the mesh quality. The local surface area is used to weight the local distortion to reflect the influence of the local surface area on the mesh quality. We have proposed the TPDMPW metric by applying the percentile weighting method to aggregate the local distortions. The experimental results on three databases demonstrate the effectiveness of the percentile weighting method and the performance superiority of the TPDMPW metric over state-of-the-art metrics. The percentile weighting method has a stronger capability to emphasize the impact of the severely distorted regions on the mesh quality than the Minkowski method and the average method. We investigate the influence of the parameters of the percentile weighting method on the performance of the TPDMPW metric, and determine the optimal values and unified values for the parameters through the empirical tests. These values can be used as the recommended values when the percentile weighting method is applied in other mesh visual quality metrics. The experimental results demonstrate that it is inappropriate to include the local surface area in the TPDMPW metric on the LIRIS/EPFL general-purpose database.

We concluded that the severely distorted regions have a great impact on the mesh quality in Chapter 3. Based on this conclusion, we have proposed the percentile weighting method to emphasize the impact of the severely distorted regions on the mesh quality in this chapter. In Chapter 5, we will investigate the perceptual importance of the local regions from the perspective of mesh saliency, and build a visual importance model based on mesh saliency to emphasize the impact of the salient regions on the mesh quality.

Chapter 5

Mesh Quality Assessment by Incorporating Mesh Saliency

5.1 Introduction

In the community of mesh quality assessment, the scholars investigated the mesh visual quality metrics [1–6] mainly by utilizing geometric information, such as the vertex curvature, surface roughness and dihedral angle. As another important research field related to visual perception, mesh saliency [22] has also attracted much attention in the community. Many computational saliency methods [23–27] have been proposed to detect the perceptually important regions of the mesh where the human eyes are focused on. In the community of image quality assessment, some scholars [19, 28–31] have investigated the incorporation of either visual attention or computational image saliency into the image visual quality (IVQ) metrics. Most of the research works [19, 28–31] were based on the assumption that the distortions occurring in more salient regions of the image are more visible and thus more important to the overall quality of the image. They basically weighted the local distortion of the pixel by the visual fixation map or the image saliency map, and then verified the assumption through the comparative experiment.

Zhang et al. [32] conducted a statistical analysis on the added value of incorporating the image saliency map into the IVQ metric. They concluded that the computational saliency methods can yield a performance gain statistically when incorporating the image saliency map into the IVQ metric, and the amount of performance gain depends on the individual combination of the computational saliency method and the IVQ metric [32]. Since the receptor of both mesh quality and mesh saliency is the human visual system, we believe that it is possible to improve the performance of the MVQ metric by incorporating mesh saliency map into the metric. In this chapter, we investigate the incorporation of the mesh saliency map into the MVQ metric. We assume that the distortions appearing on the more salient regions of the surface are more important to the overall quality of the mesh. We will verify this assumption through the comparative experiment.

Many computational methods [22] have been proposed to detect the saliency on the mesh. These methods employed different perceptual models to compute the saliency map and predict which regions on the mesh attract more attention from the human eyes. In the literature [23–25], three well-known computational methods were proposed to detect the mesh saliency and used to optimize the mesh processing operations. The effectiveness of these three computational methods were demonstrated in the graphics applications, such as mesh simplification and viewpoint selection. In this chapter, we adopt these three mesh saliency detection methods [23–25] to generate the saliency maps for the meshes.

The remainder of this chapter is organized as follows: In Section 5.2, we introduce the visual importance model based on mesh saliency and propose the TPDMVS metric. In Section 5.3, we give a brief introduction to three mesh saliency detection methods that are used in this chapter and analyze the saliency maps generated by these three methods. In Section 5.4, we evaluate the performance of the TPDMVS metric and analyze the experimental results. We summarize this chapter in Section 5.5.

5.2 Mesh Visual Quality Metric with the Saliency-based Visual Importance Model

We incorporate the saliency map of the mesh in the mesh visual quality metric. We build a visual importance model based on mesh saliency to emphasize the impact of the salient regions on the mesh quality, and propose the TPDMVS metric [33] by applying the visual importance model at the stage of spatial pooling. The flowchart of the TPDMVS metric is illustrated in Fig. 5.1. Among state-of-the-art MVQ metrics [1–6], the TPDM metric [5] correlates well with the subjective opinions and shows good performance for mesh quality assessment. So we use the TPDM metric to evaluate the local distortions of the mesh.

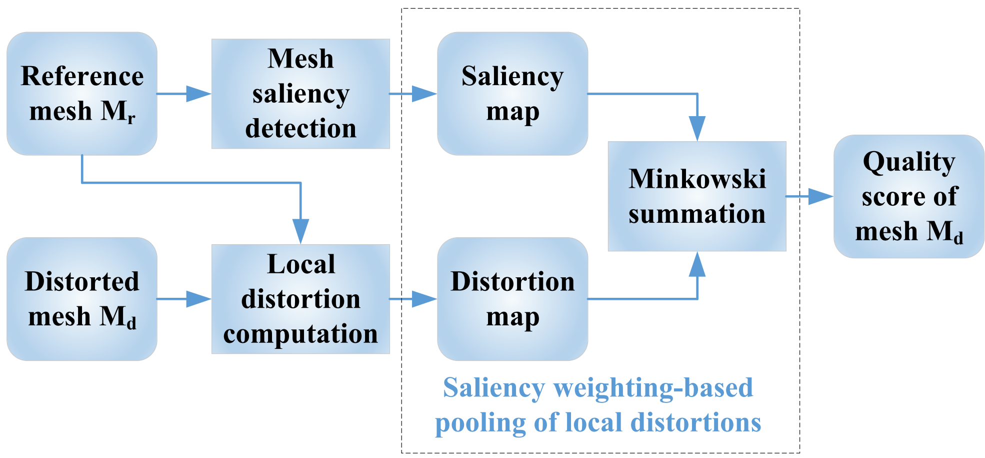


Figure 5.1 : The flowchart of the TPDMVS metric.

Given a reference mesh and a distorted mesh, firstly we compute the local distortion map between the reference mesh and the distorted mesh via Eq. (3.3). Secondly, we generate the saliency map for the reference mesh with each of three mesh saliency detection methods [23–25]. The saliency map is normalized so that the saliency value of each vertex lies within the range [0, 1]. Thirdly, we build the saliency-based visual importance model by assigning the saliency value as the weighting coefficient of the local distortion for each vertex. When two vertices with different saliency values have the same intensity of local distortions, the vertex with a higher saliency value will have a greater contribution to the quality score than the vertex with a lower saliency value. Consequently, the local distortions occurring on the salient regions will have a greater impact on the mesh quality than the local distortions with the same strength occurring on the non-salient regions. Finally, we aggregate the saliency-weighted local distortions with the Minkowski method to generate the quality score for the distorted mesh. The Minkowski method is capable of emphasizing the impact of the severely distorted regions on the mesh quality. The TPDMVS metric is formulated as follows:

$$TPDMVS = \left(\frac{1}{N} \sum_{i=1}^{N} s_i d_i^q \right)^{\frac{1}{q}}, \tag{5.1}$$

where N is the number of vertices in the reference mesh, s_i is the saliency value of vertex v_i in the reference mesh, d_i is the local distortion of vertex v_i, and q is the Minkowski exponent. The Minkowski method has been used for spatial pooling in several MVQ metrics [1, 2, 5], where the value of the Minkowski exponent was determined empirically to achieve the best performance. As suggested in literature [2], the value of the Minkowski exponent typically lies within the range [2.0, 4.0]. Our preliminary experiment demonstrates that the TPDMVS metric shows overall good performance in the case of $q = 4$, so we set $q = 4$ in the TPDMVS metric. Note that the TPDMVS metric does not include the local surface area while the TPDM metric [5] used the local surface area to weight the local distortion for each vertex at the stage of spatial pooling. We will analyze the influence of the local surface area on the performance of the metric in Section 5.4.3.

In the community of mesh quality assessment, there are few works that investigated the application of visual saliency into MVQ metrics. Nouri et al. [62] proposed the Saliency-based Mesh Quality Index (SMQI) metric by using a multiscale saliency map to compute the local statistics that reflect the structural information of the mesh. Though the SMQI metric [62] also involves the mesh saliency map, the TPDMVS metric differs from the SMQI metric in two major aspects. Firstly, the role of the saliency map in the TPDMVS metric is different from the role of the saliency map in the SMQI metric. The SMQI metric uses a saliency map generated by the mesh saliency detection method in literature [27] to compute the local structural distortions, which are then pooled into the quality score via a weighted Minkowski summation method. The TPDMVS metric firstly generates a local distortion map with the TPDM metric [5] and a saliency map with each of three mesh saliency detection methods [23–25], then assigns the saliency value as the weighting coefficient of the local distortion for each vertex, and finally aggregates the weighted local distortions into the quality score of the mesh with the Minkowski method. Secondly, the TPDMVS metric was developed based on the TPDM metric which is one of the best-performing state-of-the-art metrics while the SMQI metric was developed based on the structural information [7]. The TPDMVS metric inherits both the merit of detecting perceptual distortions that reflect the mechanism of human visual system, and the merit of detecting perceptually important regions that reflect the preference of human perception. The performance comparison in Section 5.4.2 demonstrates the performance superiority of the TPDMVS metric over the SMQI metric.

5.3 Mesh Saliency Detection Methods

Many computational methods [22–27] have been proposed to detect the saliency on the mesh. In this chapter, we employ three well-known mesh saliency detection methods [23–25] to investigate the benefit of integrating the mesh saliency map into the MVQ metric since they were demonstrated to be effective in graphics applications. We generate a saliency map for the reference mesh with each of three methods and use the saliency map in the TPDMVS metric. We denote the method in literature [23] as MS, the method in literature [24] as MSSP and the method in literature [25] as MSLCE.

5.3.1 Mesh Saliency Detection Based on the Curvature Difference

Lee et al. [23] proposed a mesh saliency detection method MS based on the curvature difference and computed the saliency value of the vertex in a multi-scale space by applying the center-surround operators on the Gaussian-weighted curvatures. The MS saliency method uses Taubin's method [94] to generate a mean curvature map $\mathscr{C}$ that maps from each vertex v of the mesh to its mean curvature $\mathscr{C}(v)$. Let $\mathscr{N}(v,\sigma) = \{x|\ ||x-v|| < \sigma, x\ \textit{is a mesh point}\}$ denote the neighbourhood point set of the vertex v within Euclidean distance σ. The Gaussian-weighted average of mean curvature for the vertex v, $G(\mathscr{C}(v),\sigma)$, is computed from its neighbourhood points as follows:

$$G(\mathscr{C}(v),\sigma) = \frac{\sum\limits_{x\in\mathscr{N}(v,2\sigma)} \mathscr{C}(x)exp[-||x-v||^2/(2\sigma^2)]}{\sum\limits_{x\in\mathscr{N}(v,2\sigma)} exp[-||x-v||^2/(2\sigma^2)]}. \tag{5.2}$$

The saliency value $\mathscr{S}(v)$ of vertex v is derived as the absolute difference between the Gaussian-weighted averages that are computed at the fine and coarse scales. The standard deviation of the Gaussian filter at the coarse scale is the twice of the standard deviation of the Gaussian filter at the fine scale. The saliency value of vertex v at the scale level t is defined as:

$$\mathscr{S}_t(v) = |G(\mathscr{C}(v),\sigma_t) - G(\mathscr{C}(v),2\sigma_t)|, \tag{5.3}$$

where σ_t is the standard deviation of the Gaussian filter at the scale t. In literature [23], five scales were used: $\sigma_t \in \{2\varepsilon, 3\varepsilon, 4\varepsilon, 5\varepsilon, 6\varepsilon\}$, where the parameter ε is set to be 0.3% of the length of the diagonal of the bounding box of the mesh.

After the saliency map $\mathscr{S}_t$ at the scale level t is normalized, the maximum saliency value M_t and the average value $\bar{m}_t$ of the local maxima that exclude the global maximum at the scale level t are computed. Then the normalized saliency map $\mathscr{S}_t$ is multiplied by the factor $(M_t - \bar{m}_t)^2$. Finally, a non-linear suppression operator $\mathscr{O}$ is applied to the saliency map at each scale, and the final saliency map s of the mesh is derived by adding the saliency maps at all the scales: $s = \sum_t \mathscr{O}(\mathscr{S}_t)$. The suppression operator $\mathscr{O}$ suppresses the saliency maps with a large number of similar peaks and promotes the saliency maps with a small number of high peaks [23], and thus will reduce the number of salient vertices on the mesh.

5.3.2 Mesh Saliency Detection via Spectral Processing

Song et al. proposed a method MSSP to detect mesh saliency by analyzing the spectral properties of mesh [24]. Given a mesh M, the MSSP method firstly decomposes the geometric Laplacian matrix L of mesh M via eigenvalue decomposition: $L = B\Lambda B^T$, where Λ denotes a diagonal matrix whose entries are eigenvalues of L, and B denotes an orthogonal matrix whose columns are the eigenvectors of L. Let R denote a diagonal matrix whose entries are the exponentials of the elements of the spectral irregularity matrix, and W denote the distance-weighted adjacency matrix of the mesh. A matrix S in the spatial domain is generated via $S = BRB^T \cdot W$, where "$\cdot$" denotes the element-by-element multiplication. For vertex v_i, a saliency value $S(v_i)$ is calculated by summing all the elements in i-th row of the matrix S. Then, the spectral saliency value $S(v_i,t)$ of vertex v_i at scale t is computed in the Difference-of-Gaussian scale space. Let $k(i)$ denote the multiplicative factor computed from the one-ring neighbor vertices of vertex v_i. The scale saliency value $\tilde{S}(v_i,t)$ of vertex v_i at scale t is computed as the absolute difference between $S(v_i,k(i)t)$ and $S(v_i,t)$. Multiple scale saliency maps $\tilde{S}_t$ are generated by varying the value of the scale parameter t. In literature [24], five scales are used: $t \in \{\varepsilon^2, 2\varepsilon^2, 3\varepsilon^2, 4\varepsilon^2, 5\varepsilon^2\}$, where ε is 0.2% of the length of the diagonal of the bounding box of the mesh.

Since the eigenvalue decomposition of the Laplacian matrix of the mesh has a high computational complexity with respect to the number of vertices of the mesh, the running time and consumed memory will be prohibitively high when the number of vertices in the mesh is large. In this chapter, we employ the QSlim method [99] to simplify the original high-resolution mesh M and obtain a simplified mesh M'. The saliency map $\tilde{S}'_t$ of the simplified mesh M' at each scale t is firstly computed and then the saliency map $\tilde{S}_t$ of mesh M at the scale t is obtained by mapping the saliency map $\tilde{S}_t'$ to the mesh M using a k-d tree. After the saliency map $\tilde{S}_t$ of mesh M at each scale is obtained, a saliency map $\tilde{S}$ of mesh M is computed by adding the saliency maps $\tilde{S}_t$ at all the scales and then smoothed using the Laplacian smoothing method. The saliency map s of mesh M is finally generated by performing a logarithmic transformation operation on $\tilde{S}$: $s = \log \tilde{S}$.

5.3.3 Mesh Saliency Detection via Local Curvature Entropy

Limper et al. proposed a method MSLCE [25] to detect the mesh saliency by computing the local curvature entropy for each vertex of the mesh within the geodesic neighborhood. The MSLCE method generates a saliency map by aggregating the saliency maps at multiple scales using an average weighting scheme. The mean curvature $\mathscr{C}(v_i)$ for each vertex v_i of the mesh is firstly computed in the same way as in literature [23]. By considering the neighbourhood vertex set $\mathscr{N}(v_i, r) = \{v_0', v_1', \cdots, v_m'\}$ of vertex v_i within the geodesic distance r, the curvature values of the vertices in the vertex set $\mathscr{N}(v_i, r)$ are partitioned into n_1 bins using a uniform sampling, which results in a set of discrete symbols $\{\rho_0, \rho_1, \cdots, \rho_{n_1}\}$. The number of bins n_1 is set as 256 in literature [25]. Let A_k denote the local surface area of each vertex v_k' in the vertex set $\mathscr{N}(v_i, r)$. The probability of the symbol ρ_j ($0 \leq \mathrm{j} \leq n_1$) in the neighbourhood vertex set $\mathscr{N}(v_i, r)$ of vertex v_i is computed as follows:

$$p_{\mathscr{N}(v_i,r)}(\rho_j) = \frac{\sum_k^m A_k \chi_k(\rho_j)}{\sum_k^m A_k}, \tag{5.4}$$

where $\chi_k(\rho_j)$ is defined as:

$$\chi_k(\rho_j) = \begin{cases} 1, & \text{if } binnedCurvature(v_k') = \rho_j \\ 0, & \text{otherwise} \end{cases}. \tag{5.5}$$

By applying the Shannon entropy to the set of symbols ρ_j, the saliency value of vertex v_i is computed as its local curvature entropy through Eq. (5.6):

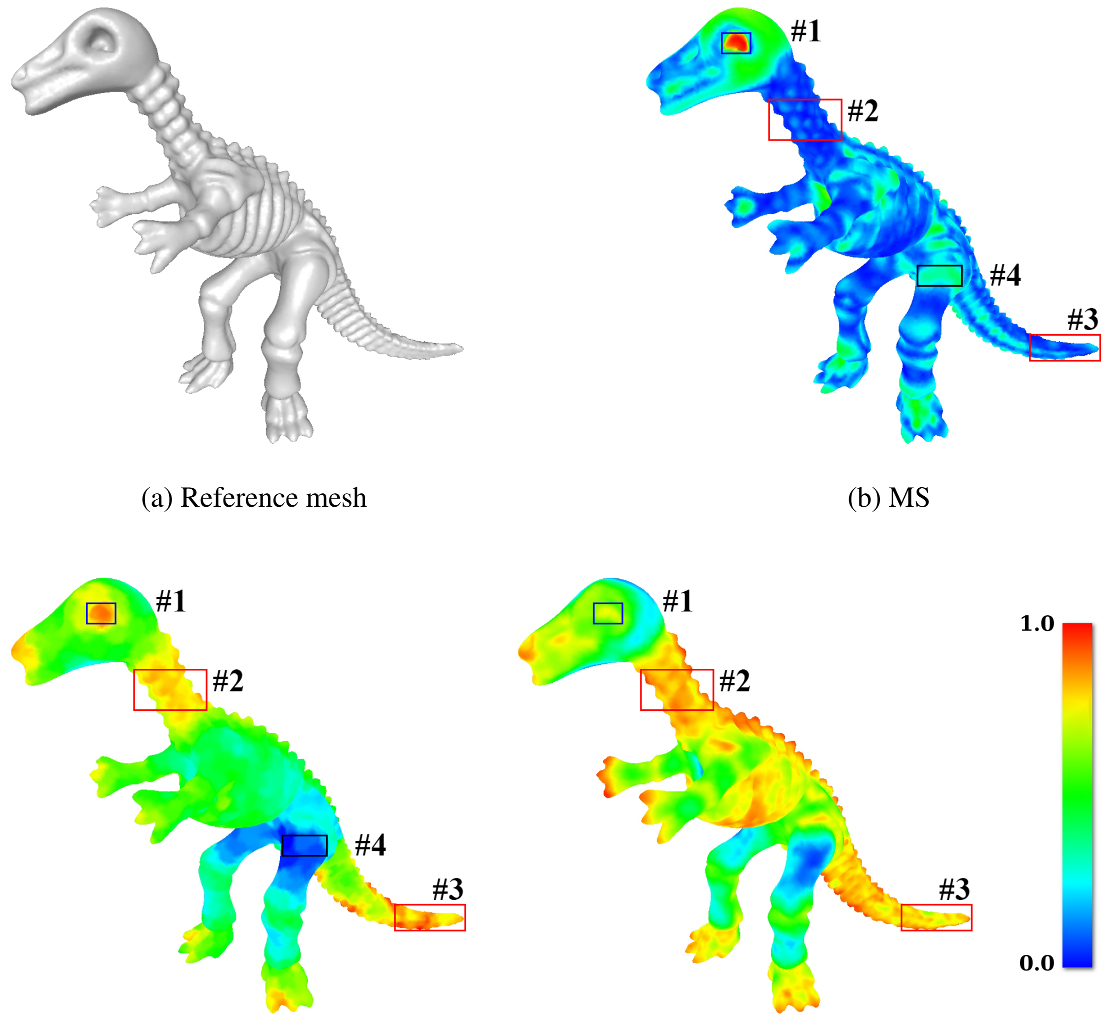
$$H(v_i) = -\sum_j^{n_1} p_{\mathscr{N}(v_i,r)}(\rho_j) \log_2(p_{\mathscr{N}(v_i,r)}(\rho_j)). \tag{5.6}$$

In order to compute the saliency values of the vertices at multiple scales, the value of the radius parameter r is varied up to a maximum value r_{max}. The saliency maps are computed at multiple levels $l_0, \cdots, l_{t_0-1}$, where the value of the radius parameter for each level l_t is equal to $r_t = 2^{-t} r_{max}$. In literature [25], five different scales are used: $t_0 = 5$, and r_{max} is set to be $0.08\sqrt{A_M}$ where A_M is the total area of the surface. The final saliency map s is generated for the mesh by aggregating the saliency maps at all the levels using the average weighting method.

5.3.4 Analysis of Mesh Saliency Detection Methods

In this section, we perform an analysis of three mesh saliency detection methods [23–25]. We compute a saliency map for the mesh with each of three methods, and analyze the similarities and differences between three saliency maps. We use the reference meshes of the Dinosaur model and the RockerArm model in the LIRIS/EPFL general-purpose database [1] for the experimental analysis. We generate a saliency map for the reference mesh of the Dinosaur model and the RockerArm model with each mesh saliency detection method. Fig. 5.2 illustrates the saliency maps computed by MS, MSSP and MSLCE for the reference mesh of Dinosaur model while Fig. 5.3 illustrates the saliency maps computed by MS, MSSP and MSLCE for the reference mesh of RockerArm model. The colormap is used to map the saliency value to the RGB color for each vertex in the mesh. As illustrated in Fig. 5.2e and Fig. 5.3e, for each vertex of the mesh, the red color represents a high saliency value, the green color represents a median saliency value, and the blue color represents a low saliency value. When the saliency value of a vertex is higher than the mean of the saliency values of the mesh, we regard the vertex as a salient vertex in the mesh. Otherwise, we regard the vertex as a non-salient vertex in the mesh.

From Fig. 5.2 and Fig. 5.3, we observe that, on the same mesh, the saliency map of the MSLCE method is overall warmer than the saliency map of the MSSP method while the saliency map of the MSSP method is overall warmer than the saliency map of the MS

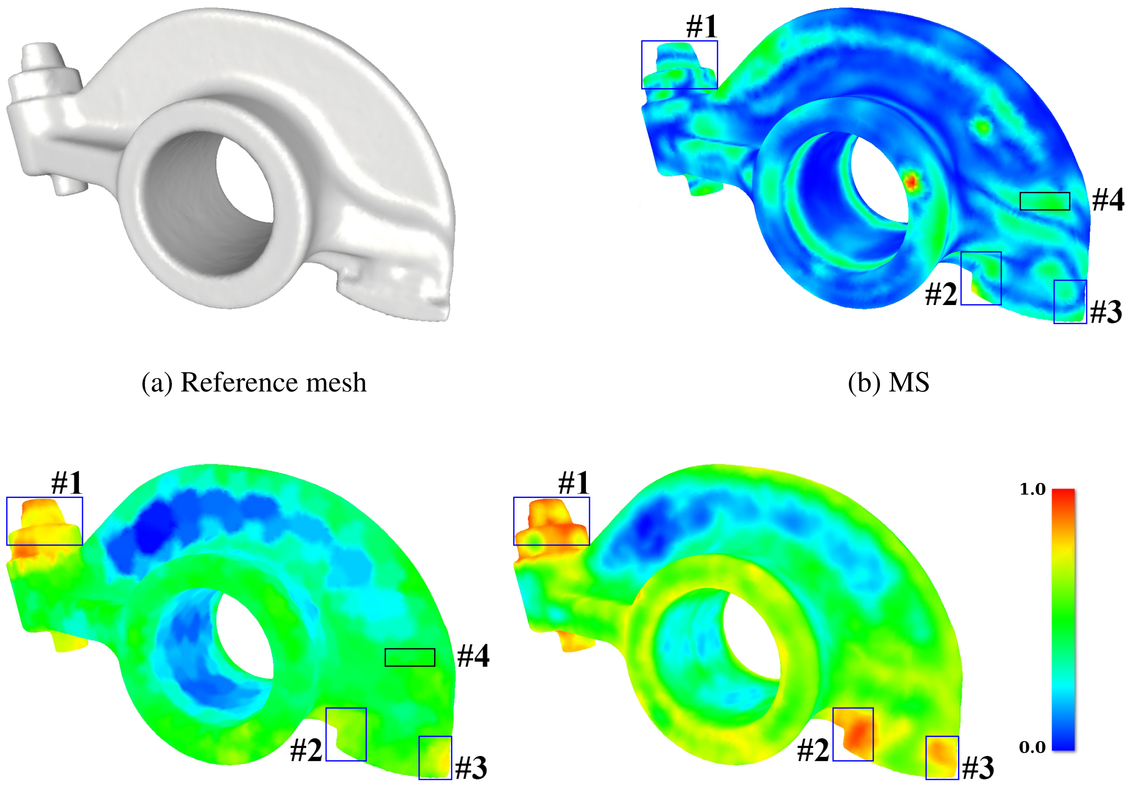


(a) Reference mesh (b) MS

(c) MSSP (d) MSLCE (e)

Figure 5.2 : The saliency maps computed by the MS, MSSP and MSLCE methods for the reference mesh of the Dinosaur model.

method. We also observe that three methods [23–25] detect some common vertices as the salient vertices at some regions though the salient vertices that each method detects are not exactly the same. Particularly, there is a relatively higher similarity between the saliency maps of the MSSP method and the MSLCE method since MSSP and MSLCE detect more common vertices as salient vertices among three saliency methods. On the reference mesh of Dinosaur model, all the three saliency methods detect the vertices at the #1 region (the left eye region) as salient vertices, as shown in the blue rectangles of



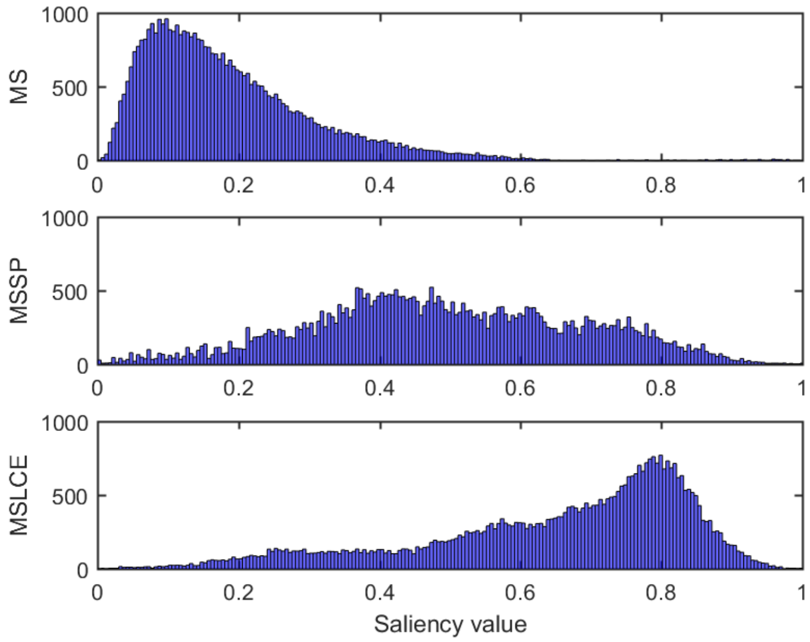
(a) Reference mesh (b) MS

(c) MSSP (d) MSLCE (e)

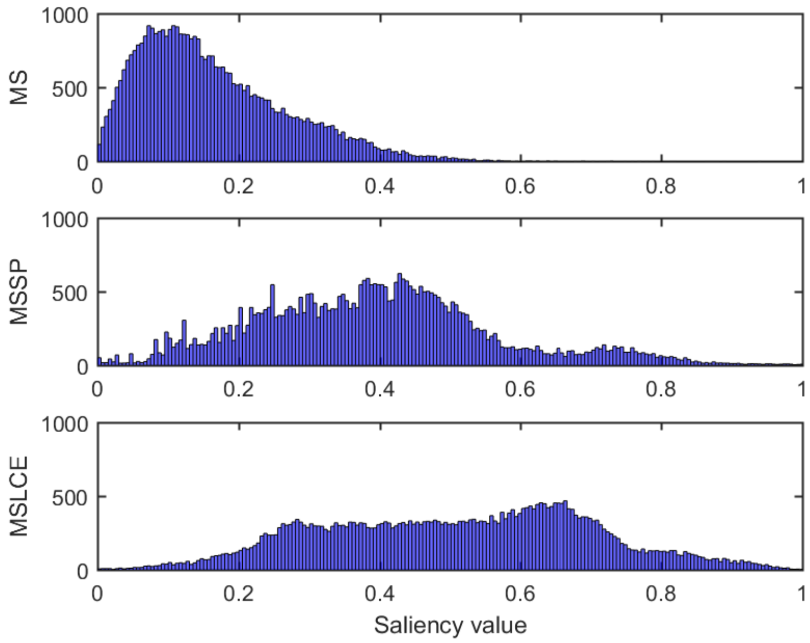
Figure 5.3 : The saliency maps computed by the methods MS, MSSP and MSLCE for the reference mesh of the RockerArm model.

Fig. 5.2b - Fig. 5.2d. Besides, at some other regions, such as the #2 region (the neck region) and the #3 region (the tail region) as shown in the red rectangles of Fig. 5.2b - Fig. 5.2d, both MSSP and MSLCE methods detect the vertices as salient vertices which however are detected as non-salient vertices by the MS method. On the reference mesh of the RockerArm model, at the #1, #2, and #3 regions as shown in the blue rectangles of Fig. 5.3b - Fig. 5.3d, both MSSP and MSLCE methods detect generally high saliency while the MS method detects high saliency only at some parts of these regions and low saliency at the remaining part of these regions.

In order to analyze the statistical distribution characteristics of each saliency map, we plot a histogram for each saliency map generated by three saliency methods on the



(a) Dinosaur



(b) RockerArm

Figure 5.4 : The histograms of the saliency maps generated by three mesh saliency detection methods on the reference meshes of the Dinosaur model and RockerArm model.

reference meshes of the Dinosaur model and RockerArm model in Fig. 5.4. We list the statistical characteristics of three individual saliency maps on the reference mesh of the Dinosaur model and RockerArm model respectively in Table 5.1 and Table 5.2, where $Mean$ and Std represent the mean and standard deviation of the saliency map respectively. We sort the saliency map in the ascending order. Then Q_1, Q_2 and Q_3 stand for the first quartile, the second quartile, and the third quartile of the sorted saliency map respectively. We observe that three saliency maps show different statistical distributions on the same mesh. When comparing the statistical characteristics of three saliency maps in terms of Q_1, Q_2, Q_3 and $Mean$, on the reference mesh of either the Dinosaur model or the RockerArm model, MSLCE always has a greater value than MSSP while MSSP always has a greater value than MS. Thus, the saliency map of MSLCE has overall greater values than the saliency map of MSSP while the saliency map of MSSP has overall greater values than the saliency map of MS. This conclusion is consistent with the visual illustration in Fig. 5.2 and Fig. 5.3.

Table 5.1 : The statistical characteristics of three individual saliency maps on the reference mesh of the Dinosaur model.

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

We use the Pearson linear correlation coefficient (PLCC) to measure the similarity between two saliency maps on the same mesh. The PLCC has been used to evaluate the similarity between two saliency maps in the image saliency detection [22, 100, 101]. We list the PLCC value between each pair of saliency maps on the reference meshes of Dinosaur model and RockerArm model in Table 5.3. The PLCC value lies within the range [-1, 1], and a greater PLCC value indicates a higher similarity between two saliency

Table 5.2 : The statistical characteristics of three individual saliency maps on the reference mesh of the RockerArm model.

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

Table 5.3 : The PLCC value (%) between each pair of saliency maps on the reference meshes of the Dinosaur model and RockerArm model.

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

maps. We observe that the rank of three PLCC values is the same on each model though there is a significant difference in the PLCC values between two models. On either the Dinosaur model or the RockerArm model, the PLCC value between the saliency maps of MS and MSLCE is smallest, the PLCC value between the saliency maps of MSSP and MSLCE is greatest, and the PLCC value between the saliency maps of MS and MSSP is median. This indicates that, relatively speaking, the similarity between the saliency maps of MSSP and MSLCE is greatest, the similarity between the saliency maps of MS and MSLCE is lowest, and the similarity between the saliency maps of MS and MSSP is median.

5.4 Experimental Results and Analysis

5.4.1 Experimental Protocol

In this section, we use the LIRIS/EPFL general-purpose database [1] as the test bed to evaluate the performance of the TPDMVS metric. We compute the quality scores for all the distorted meshes in the LIRIS/EPFL general-purpose database with the TPDMVS metric, and then compute the PLCC and SROCC values between the quality scores and the MOSs of the meshes. We validate the superiority of the TPDMVS metric by comparing the performance of the TPDMVS metric with the performance of state-of-the-art MVQ metrics. Before computing the correlation coefficients, we conduct a psychometric fitting between the quality scores and MOSs with the cumulative Gaussian function to remove the nonlinearity [84].

We use each of three mesh saliency detection methods [23–25] that are described in Section 5.3 to generate a saliency map s for the reference mesh of each model in the LIRIS/EPFL general-purpose database, and then compute a quality score for the distorted mesh by incorporating the saliency map s in the TPDMVS metric through Eq. (5.1). We calculate the correlation coefficients of the TPDMVS metric for each of three mesh saliency detection methods. Note that the MS method [23] takes a long time to compute the saliency map particularly for the high-resolution mesh. Thus, in the case of the MS method [23], we use the QSlim method [99] to simplify the high-resolution mesh M into a simplified mesh M', and then generate a saliency map s' for the simplified mesh M'. The saliency map s of the high-resolution mesh M is finally obtained by using a closest point matching strategy as in literature [24].

5.4.2 Performance Comparison

We compare the performance of the TPDMVS metric with the performance of state-of-the-art MVQ metrics, including Hausdorff Distance (HD) [49], Root Mean Square

Error (RMS) [49], Geometric Laplacian (GL1, GL2) [43, 44], Strain Field (SF) [61], 3DWPM1 [60], 3DWPM2 [60], MSDM [1], MSDM2 [2], FMPD [4], DAME [3], TPDM [5], SMQI [62], Dong [6], TPDMSP [20] and TPDMPW [21]. Table 5.4 lists the PLCC and SROCC performance values of the TPDMVS metric and state-of-the-art MVQ metrics on the LIRIS/EPFL general-purpose database. Since the PLCC performance value of the SMQI metric is not reported in literature [62], only the SROCC performance value is listed for the SMQI metric in Table 5.4. TPDMVS(MS) indicates the performance of the TPDMVS metric with the MS method [23], TPDMVS(MSSP) indicates the performance of the TPDMVS metric with the MSSP method [24], and TPDMVS(MSLCE) indicates the performance of the TPDMVS metric with the MSLCE method [25]. TPDMSP(m) indicates the median performance of the TPDMSP metric in the random sample testing experiment on the LIRIS/EPFL general-purpose database. TPDMPW(o) indicates the optimal performance of the TPDMPW metric on the LIRIS/EPFL general-purpose database.

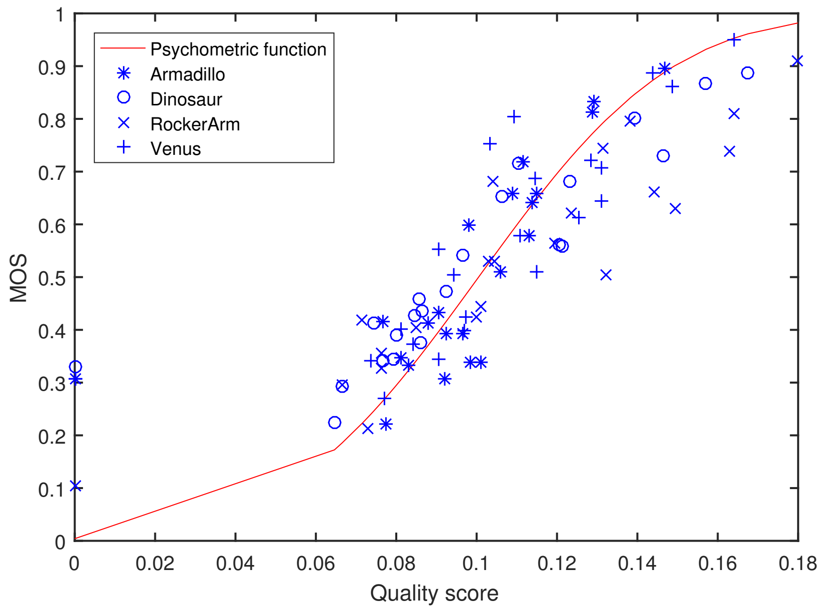


Figure 5.5 : The scatter plots of the MOSs versus the quality scores computed by the TPDMVS(MS) metric and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database.

Table 5.4 : The PLCC and SROCC values (%) of the TPDMVS metric and state-of-the-art MVQ metrics on the LIRIS/EPFL general-purpose database.

Metric	PLCC	SROCC
HD	11.4	13.8
RMS	28.1	26.8
GL1	35.5	33.1
GL2	42.4	39.3
SF	7.0	15.7
3DWPM1	61.8	69.3
3DWPM2	49.6	49.0
MSDM	75.0	73.9
MSDM2	81.4	80.4
FMPD	83.5	81.9
DAME	75.2	76.6
TPDM	84.1	84.3
SMQI	/	84.6
Dong	87.7	86.6
TPDMSP(m)	89.0	87.5
TPDMPW(o)	87.7	87.2
TPDMVS(MS)	89.0	89.3
TPDMVS(MSSP)	89.6	89.2
TPDMVS(MSLCE)	89.4	89.3

From Table 5.4, we observe that when using the saliency maps generated by any of three saliency methods [23–25], the TPDMVS metric achieves significant performance gain over the TPDM metric [5] and better performance than other state-of-the-art MVQ metrics, including the SMQI metric which also involves the mesh saliency in the metric design. This indicates that incorporating the mesh saliency into the MVQ metric can

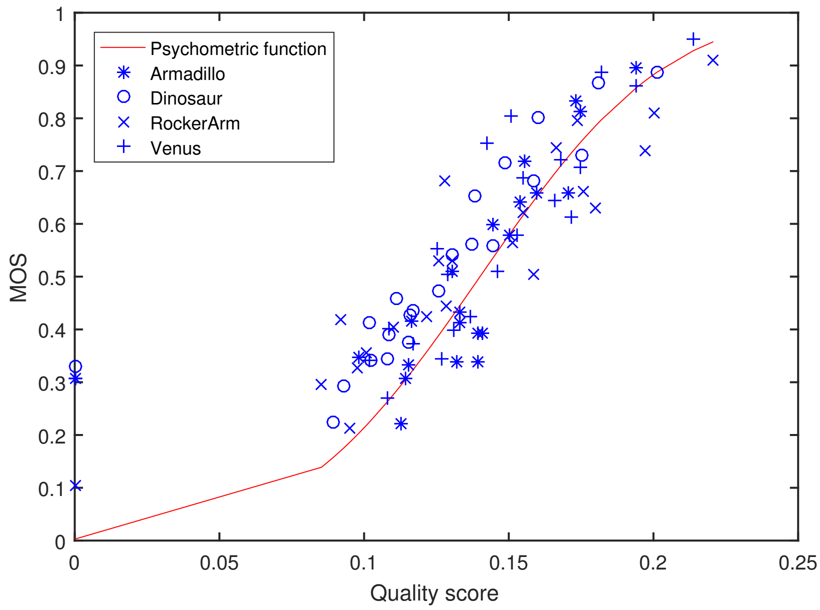


Figure 5.6 : The scatter plots of the MOSs versus the quality scores computed by the TPDMVS(MSSP) metric and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database.

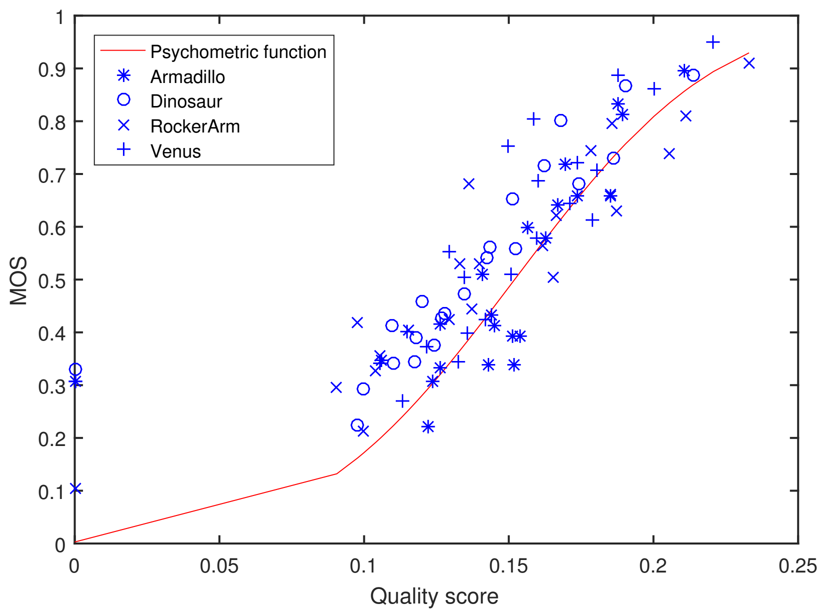


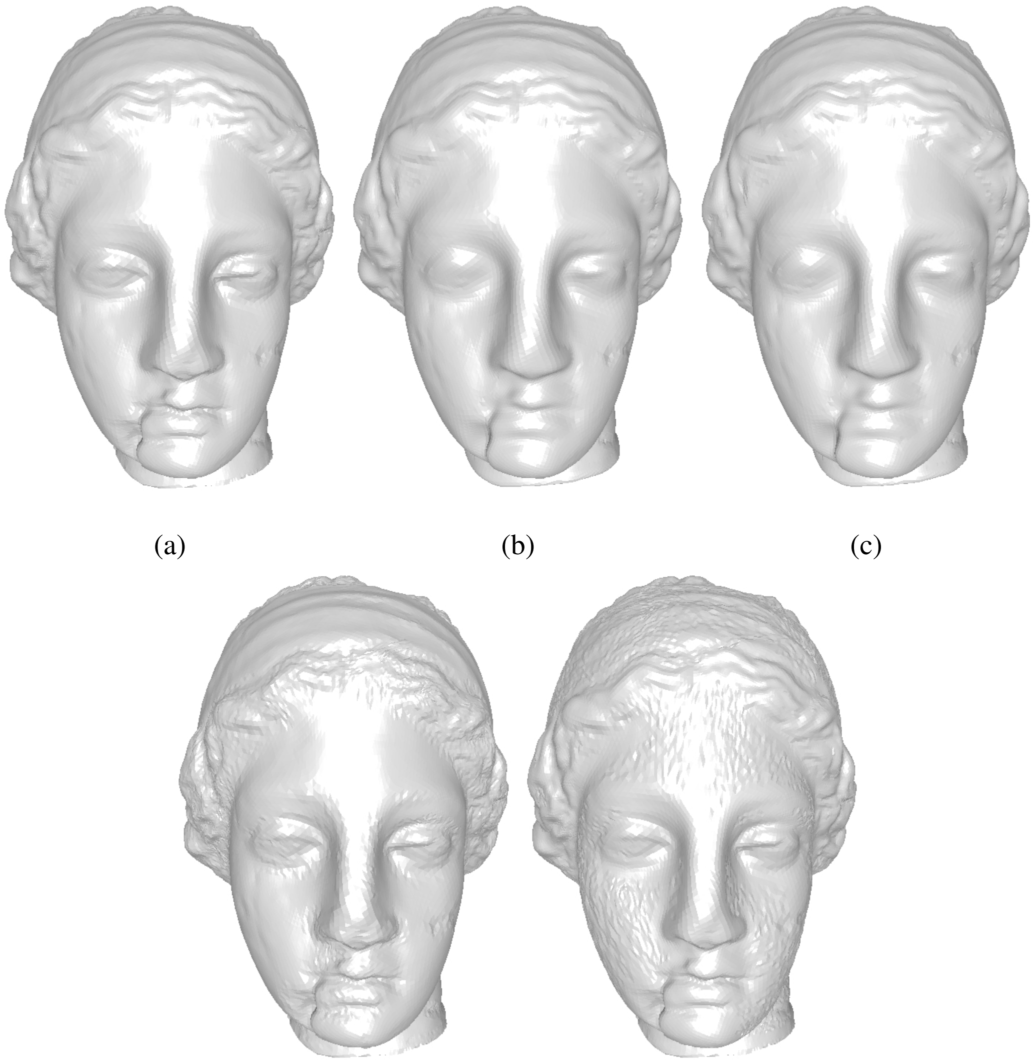
Figure 5.7 : The scatter plots of the MOSs versus the quality scores computed by the TPDMVS(MSLCE) metric and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database.

improve the performance of mesh quality assessment, and thus supports the assumption that we made in Section 5.1. Note that the TPDMVS metric achieves better performance than TPDMSP(m) and TPDMPW(o) when using the saliency maps generated by any of three saliency methods. This indicates that the saliency-based visual importance model proposed in this chapter achieves better performance for the metric than the local distortion-based visual importance model proposed in Chapter 4. The analysis in Section 5.3.4 shows that there are significant differences between the saliency maps generated by three mesh saliency detection methods. However, from Table 5.4, we observe that the TPDMVS metric achieves similar performance when using the saliency maps generated by any of three saliency methods. The reason may be that the performance of the TPDM metric [5] is already relatively high on the LIRIS/EPFL general-purpose database, and there is a performance bottleneck on the LIRIS/EPFL general-purpose database [1] which consists of a small number of meshes. In the research field of image quality assessment, any of the existing subjective image quality databases [7, 102–105] consists of hundreds or even thousands of image samples. But in the research field of mesh quality assessment, the existing subjective mesh quality databases [1, 3, 40] consist of a significantly smaller number of mesh samples. As the largest available subjective mesh quality database, the LIRIS/EPFL general-purpose database consists of only 88 mesh samples. Even though it is hard to achieve higher performance than the TPDM metric, the TPDMVS metric still achieves a performance gain by incorporating the mesh saliency map into the TPDM metric. As mentioned in literature [32], the issue of how the human attention affects the perception of visual quality has not been addressed and there is a lack of solid theoretical basis for the investigation on the relationship between human attention and visual quality. Thus, it is still difficult to explain in a theoretical way how much the performance gain will be when incorporating the visual attention or the mesh saliency map into a MVQ metric. In this section, we demonstrate the added value of mesh saliency in mesh quality assessment empirically by the performance gain of the TPDMVS metric over the TPDM metric, in a similar way as the scholars did in the image quality assessment [19, 28–32].

Table 5.5 : The process of generating the distorted meshes from the reference meshes.

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

For each of three saliency methods, we use the TPDMVS metric to compute the quality scores for all the meshes in the LIRIS/EPFL general-purpose database [1] and then perform a psychometric fitting between the quality scores and MOSs of the meshes using the cumulative Gaussian function in Eq. (2.5). Fig. 5.5, Fig. 5.6, and Fig. 5.7 illustrate the scatter plots of the MOSs versus the quality scores computed by the TPDMVS metric with each of three saliency methods and the fitted psychometric curve for the meshes



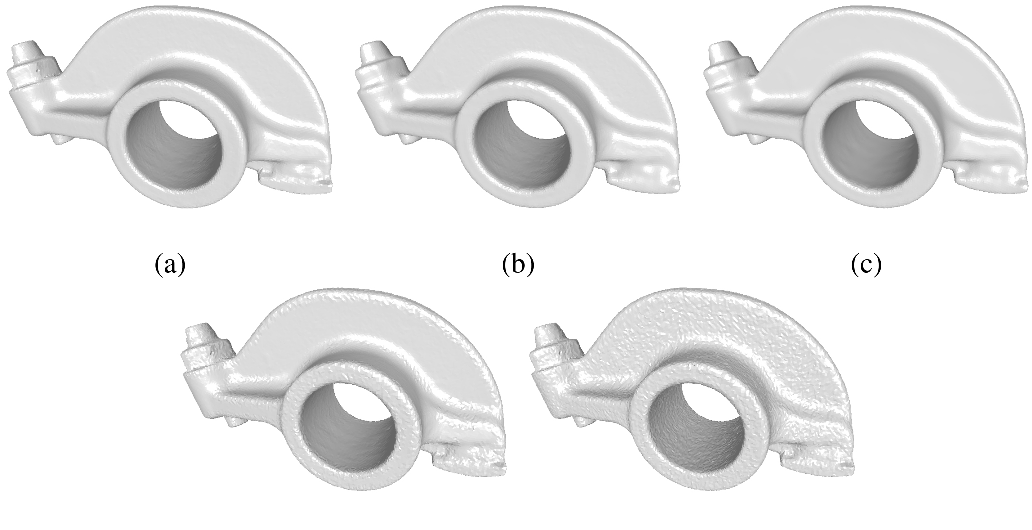
(a) (b) (c)

(d) (e)

Figure 5.8 : The reference mesh and four distorted meshes V_1, V_2, V_3, V_4 of the Venus mdoel. (a) Reference mesh. (b) V_1: MOS=3.722, QS=0.084. (c) V_2: MOS=5.530, QS=0.091. (d) V_3: MOS=5.774, QS=0.111. (e) V_4: MOS=8.867, QS=0.144.

in the LIRIS/EPFL general-purpose database. From Fig. 5.5, Fig. 5.6, and Fig. 5.7, we observe that the *QualityScore-MOS* pairs are fitted well by the psychometric fitting curve for each saliency method.

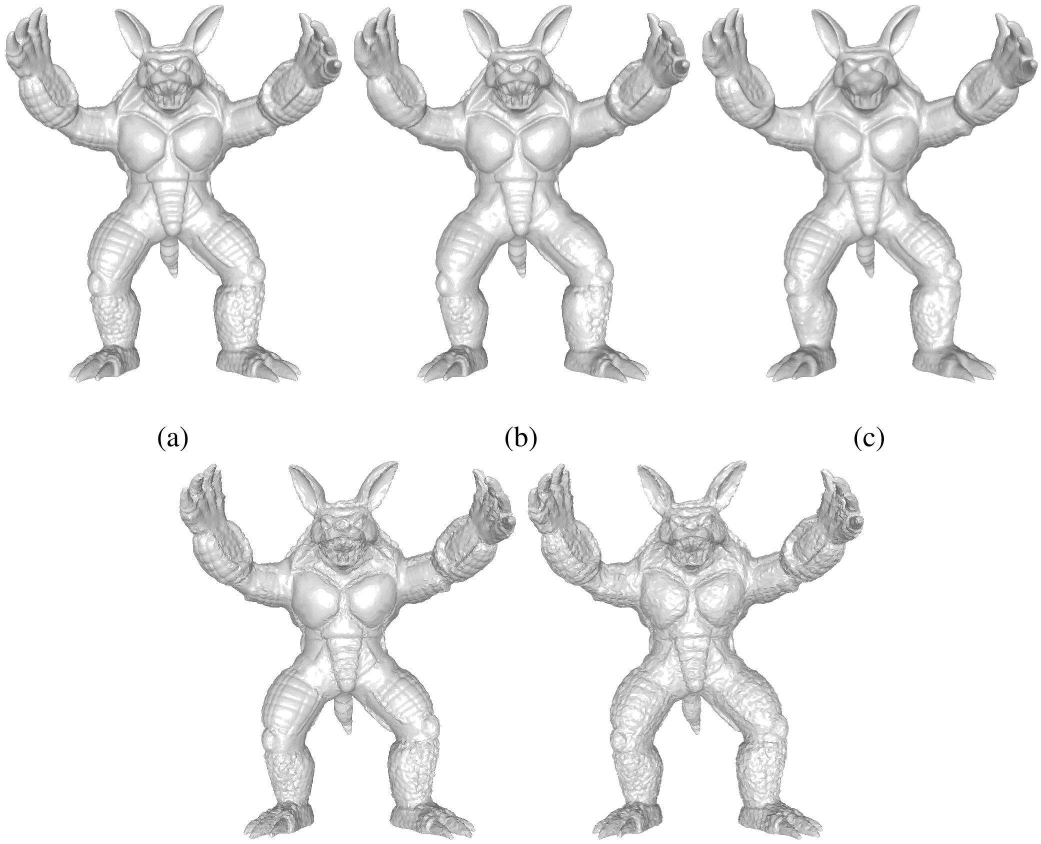
In order to demonstrate the applicability of the TPDMVS metric to different geometric models and distortion types, we choose some representative distorted meshes in the LIRIS/EPFL general-purpose database [1] for analysis. We employ the TPDMVS(MS)



(a) (b) (c) (d) (e)

Figure 5.9 : The reference mesh and four distorted meshes R_1, R_2, R_3, R_4 of the RockerArm mdoel. (a) Reference mesh. (b) R_1: MOS=4.044, QS=0.085. (c) R_2: MOS=5.288, QS=0.103. (d) R_3: MOS=6.206, QS=0.124. (e) R_4: MOS=8.106, QS=0.164.

metric to compute the quality scores of these meshes and evaluate the consistency between the quality scores and the MOSs. For each of the four models in the LIRIS/EPFL general-purpose database, we select four distorted meshes with different levels of distortions which are generated by applying the smoothing filter or adding noise with different strengths either locally or globally on the reference mesh. As stated in literature [1], these distortions reflect the distortions that appear in the common mesh processing operations, such as mesh simplification, mesh compression, and mesh watermarking. We illustrate the reference mesh and distorted meshes of each model in Fig. 5.8, Fig. 5.9, Fig. 5.10, and Fig. 5.11. The caption of each figure includes the MOSs and the quality scores of the distorted meshes that are computed by the TPDMVS(MS) metric. The MOS indicates the mean opinion score while the QS indicates the quality score. The process of generating the distorted meshes from the reference meshes is introduced in Table 5.5. We denote the distorted meshes of the Venus model as V_1, V_2, V_3, V_4, the distorted meshes of the RockerArm model as R_1, R_2, R_3, R_4, the distorted meshes of the Armadillo model as A_1, A_2, A_3, A_4, and the distorted meshes of the Dinosaur model as D_1, D_2, D_3, D_4, respectively.

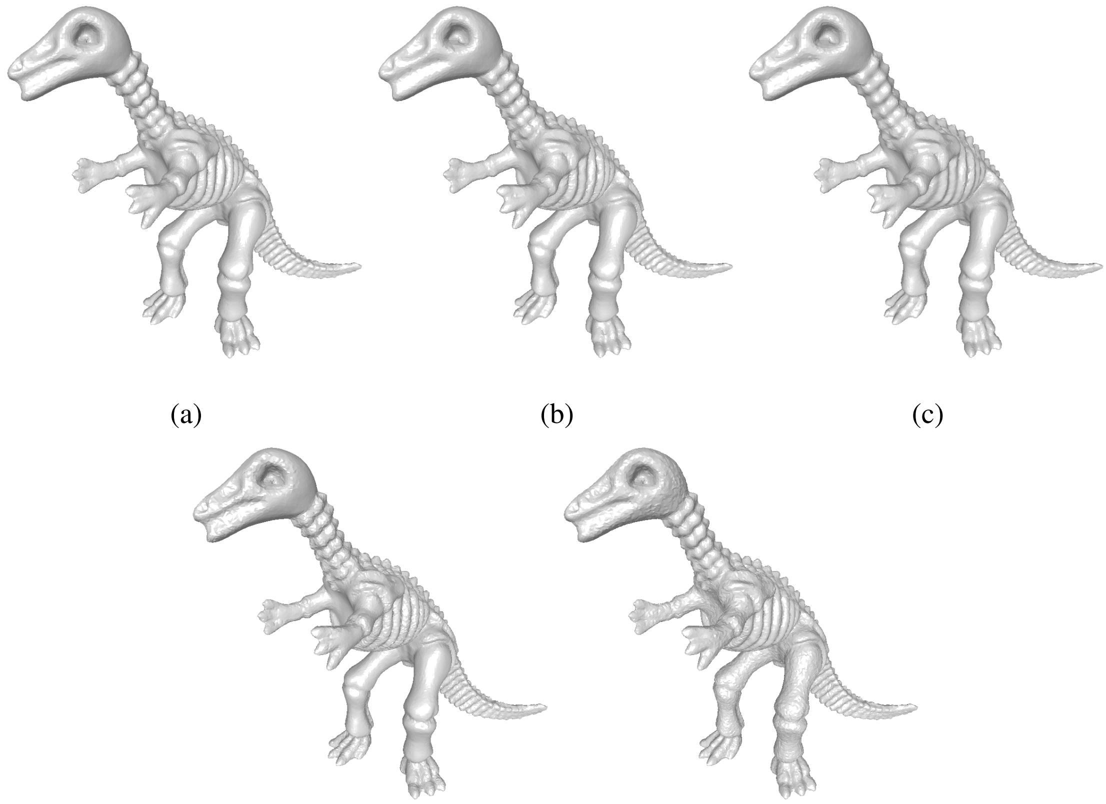


(a) (b) (c)

(d) (e)

Figure 5.10 : The reference mesh and four distorted meshes A_1, A_2, A_3, A_4 of the Armadillo mdoel. (a) Reference mesh. (b) A_1: MOS=4.134, QS=0.088. (c) A_2: MOS=5.978, QS=0.098. (d) A_3: MOS=6.412, QS=0.114. (e) A_4: MOS=8.335, QS=0.129.

From Fig. 5.8, Fig. 5.9, Fig. 5.10, and Fig. 5.11, we observe that the MOS values of four distorted meshes have exactly the same rankings with the QS values of four distorted meshes for each model despite the variations in the distortion type, distortion region and distortion strength between four models. This indicates that the TPDMVS metric has a good generalization capability of evaluating the visual quality of different models with various distortions. Though the MS saliency method [23] is used to demonstrate the applicability of the TPDMVS metric to different geometric models and distortion types, we can obtain a similar consistency between the MOS values and QS values of the distorted



(a) (b) (c)

(d) (e)

Figure 5.11 : The reference mesh and four distorted meshes D_1, D_2, D_3, D_4 of the Dinosaur mdoel. (a) Reference mesh. (b) D_1: MOS=3.429, QS=0.079. (c) D_2: MOS=4.278, QS=0.084 (d) D_3: MOS=6.540, QS=0.106 (e) D_4: MOS=8.011, QS=0.139.

meshes when using the other two saliency methods [24, 25] to generate the saliency maps in the TPDMVS metric.

5.4.3 The Influence of the Local Surface Area on the Performance

The TPDM metric [5] used the local surface area to weight the local distortion for each vertex at the stage of spatial pooling. However, the TPDMVS metric does not include the local surface area in the metric as shown in Eq. (5.1). The LIRIS/EPFL general-purpose database [1] involves two types of distortion: noise addition and smoothing. The smoothing operation usually introduces perceptually more significant distortions on the rough regions than on the smooth regions. The local surface areas on the rough regions are generally smaller than the local surface areas on the smooth regions because the rough regions generally need small-area triangles to characterize the highly curved shape while the smooth regions typically consist of large-area triangles to characterize the flat shape. Thus, in the case of the smoothing distortion, weighting the local distortion by the local surface area will lead to an overemphasis on the local distortions on the smooth regions and then result in an overestimation of the quality degradation of the mesh. Finally, the correlation between the quality scores and MOSs of the meshes in the entire database may decline to some extent. In order to validate the abovementioned analysis, we weight the local distortion by the local surface area for each vertex and develop a new MVQ metric TPDMVS-W. We compare the performance of the TPDMVS-W metric with the performance of the TPDMVS metric. The calculation formula of the TPDMVS-W metric is as follows:

$$TPDMVS\text{-}W = \left(\sum_{i=1}^{N} w_i s_i d_i^q \right)^{\frac{1}{q}}, \tag{5.7}$$

where $w_i = a_i / \sum_{i=1}^{N} a_i$ is the weighting coefficient for vertex v_i in the reference mesh, a_i is the local surface area of vertex v_i and its value is equal to one-third of the total areas of all the incident facets of vertex v_i.

Table 5.6 : The performance comparison among the TPDM, TPDMVS-W and TPDMVS metrics on the LIRIS/EPFL general-purpose database.

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

We use the TPDMVS-W metric with each of three saliency methods to compute the quality scores for the meshes in the LIRIS/EPFL general-purpose database, and compute the correlation coefficients between the quality scores and the MOSs of all the distorted meshes. Table 5.6 lists the performance values of the TPDM metric [5], the TPDMVS-W metric and the TPDMVS metric on the LIRIS/EPFL general-purpose database. From Table 5.6, we observe that, for each of three saliency methods, the TPDMVS metric always achieves better performance than the TPDMVS-W metric while the TPDMVS-W metric always achieves better performance than the TPDM metric. This indicates that there is a close relationship between the mesh saliency and mesh quality, and the performance of the MVQ metric can be improved via weighting the local distortion by the local saliency value at the stage of spatial pooling. The comparison also reveals that it is inappropriate to include the local surface area in the TPDMVS metric on the LIRIS/EPFL general-purpose database.

5.4.4 Synthetic Saliency Map

In Section 5.3.4, we analyzed the similarities and differences between the saliency maps generated by three mesh saliency detection methods [23–25]. The analysis shows that there are significant differences among the saliency maps generated by three methods. While some vertices of the mesh are detected as salient vertices by one method, they may be detected as non-salient vertices by the other two methods. However, the experimental results in Section 5.4.2 show that any of three methods leads to a performance gain when the saliency map is incorporated in the TPDMVS metric in spite of the differences among the saliency maps of three methods.

Therefore, we come up with a question naturally: Is it possible to further improve the performance if we assemble the salient regions from different saliency maps on the same mesh into a synthetic saliency map and then incorporate the synthetic saliency map in the TPDMVS metric ? We assume that the synthetic saliency map will lead to better performance than the individual saliency map. In order to verify this assumption, we firstly merge the individual saliency maps by selecting the relatively higher saliency value as the saliency value of the vertex in the synthetic saliency map and then observe if there will be a performance gain over each individual saliency map when incorporating the synthetic saliency map in the TPDMVS metric. Since the saliency maps generated by three methods [23–25] have different statistical distributions, we standardize each saliency map s by transforming it to have a mean of zero and a standard deviation of one:

$$s_i' = (s_i - s_{mean})/s_{std}, \tag{5.8}$$

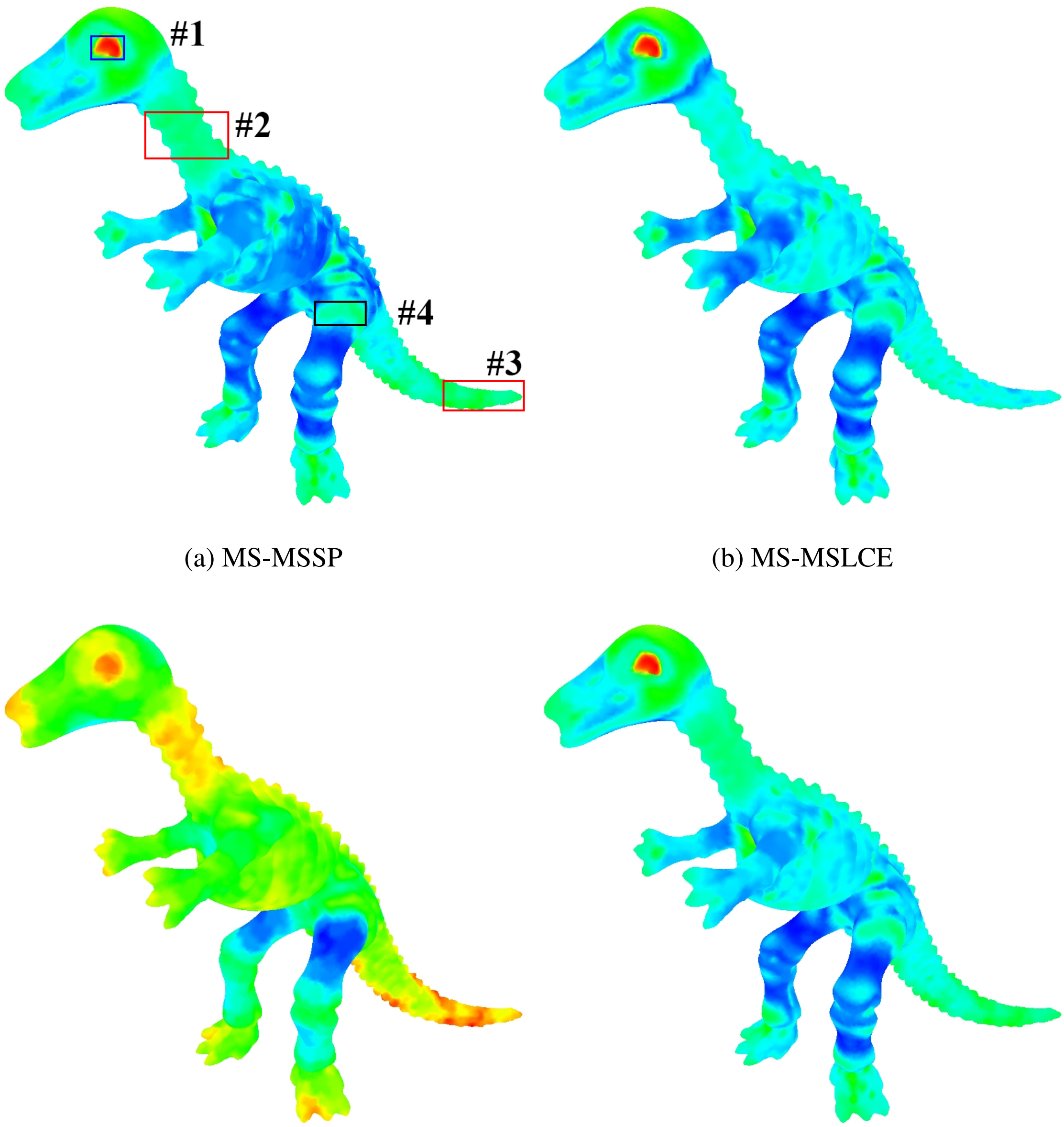
where s_i is the saliency value of vertex v_i before the standardization, s_i' is the saliency value of vertex v_i after the standardization, s_{mean} and s_{std} are the mean and standard deviation of the saliency map s respectively.

We use the *max* function to assign the higher saliency value from the standardized saliency maps as the new saliency value for each vertex. Let $s^{a'}$ and $s^{b'}$ denote two standardized saliency maps obtained via Eq. (5.8), the synthetic saliency map is generated by applying the *max* function to the element values of the saliency maps $s^{a'}$ and $s^{b'}$ for each vertex:

$$s_i^{m'} = max(s_i^{a'}, s_i^{b'}), \tag{5.9}$$

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

In Section 5.3.4, we analyzed the saliency maps of three mesh saliency detection methods [23–25] on the reference meshes of the Dinosaur model and the RockerArm model in the LIRIS/EPFL general-purpose database [1]. In this section, we analyze the relationship between the synthetic saliency map and the individual saliency maps on the reference meshes of the Dinosaur model and the RockerArm model. Fig. 5.12 and Fig. 5.13 illus-

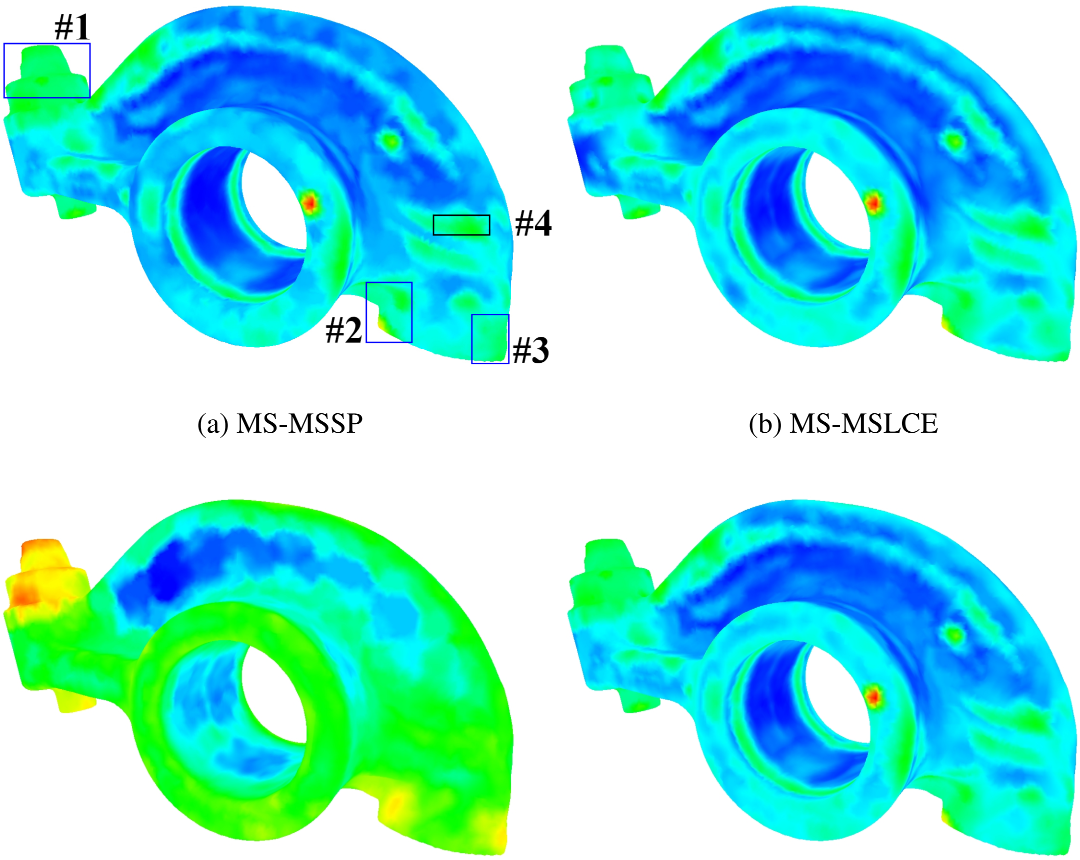


(a) MS-MSSP (b) MS-MSLCE

(c) MSSP-MSLCE (d) MS-MSSP-MSLCE

Figure 5.12 : The colormap of the synthetic saliency maps on the reference mesh of the Dinosaur model.

trate the colormap of the synthetic saliency maps on the reference meshes of the Dinosaur model and the RockerArm model respectively. MS-MSSP indicates the synthetic saliency map by merging the saliency maps of MS and MSSP, MS-MSLCE indicates the synthetic saliency map by merging the saliency maps of MS and MSLCE, MSSP-MSLCE indicates the synthetic saliency map by merging the saliency maps of MSSP and MSLCE, and MS-



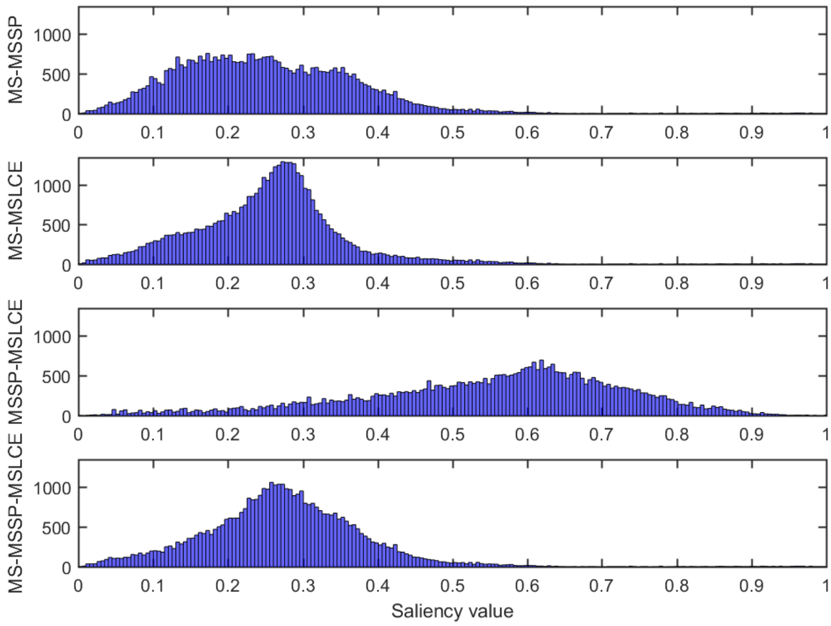
(a) MS-MSSP (b) MS-MSLCE

(c) MSSP-MSLCE (d) MS-MSSP-MSLCE

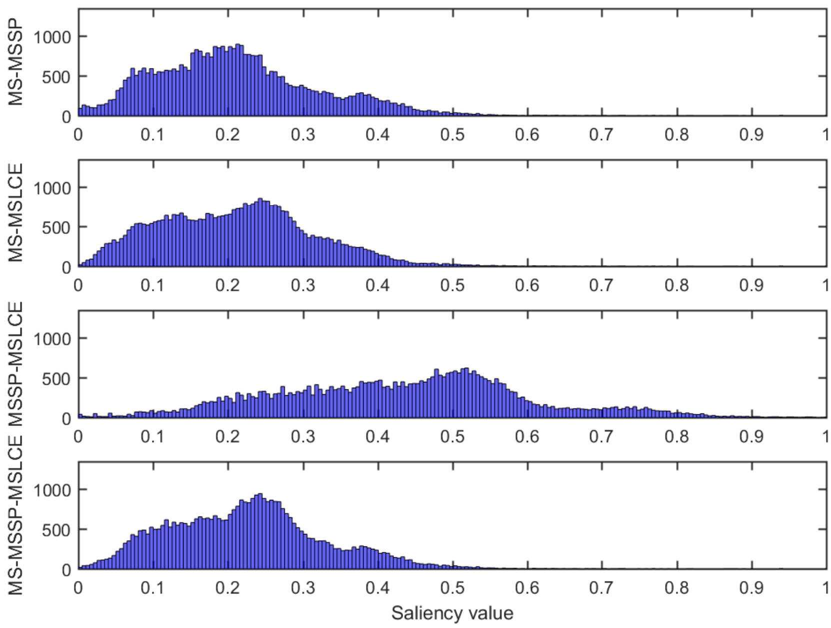
Figure 5.13 : The colormap of the synthetic saliency maps on the reference mesh of the RockerArm model.

Table 5.7 : The statistical characteristics of the synthetic saliency maps on the Dinosaur model.

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



(a) Dinosaur model



(b) RockerArm model

Figure 5.14 : The histograms of the synthetic saliency maps on the Dinosaur model and RockerArm model.

Table 5.8 : The statistical characteristics of the synthetic saliency maps on the RockerArm model.

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

Table 5.9 : The performance values of the TPDMVS metric on the LIRIS/EPFL general-purpose database when incorporating either the individual saliency map or the synthetic saliency map into the metric.

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

MSSP-MSLCE indicates the synthetic saliency map by merging the saliency maps of MS, MSSP, and MSLCE. In order to determine whether a vertex is a salient vertex on the mesh for each synthetic saliency map in Fig. 5.12 and Fig. 5.13, we plot a histogram for each synthetic saliency map on two models in Fig. 5.14 and list the statistical characteristics of the synthetic saliency maps on the Dinosaur model and the RockerArm model respectively in Table 5.7 and Table 5.8. From Fig. 5.12 and Fig. 5.13, we observe that the synthetic saliency map MSSP-MSLCE is overall warmer than the other three synthetic saliency maps on either the Dinosaur model or the RockerArm model. This observation is consistent with the comparison result obtained from the histograms of the synthetic saliency maps in Fig. 5.14, where the saliency values of MSSP-MSLCE are generally greater than the saliency values of the other three synthetic saliency maps on either the Dinosaur model or the RockerArm model. When comparing the statistical characteristics of the synthetic saliency maps in terms of Q_1, Q_2, Q_3 and $Mean$ in Table 5.7 and Table 5.8, we also observe that the synthetic saliency map MSSP-MSLCE always has a significantly greater value than the other three synthetic saliency maps on either the Dinosaur model or the RockerArm model.

By comparing Fig. 5.12 with Fig. 5.2, and Fig. 5.13 with Fig. 5.3, we observe that the salient regions on each individual saliency map are preserved well on the synthetic saliency maps. We use the synthetic saliency map MS-MSSP to elaborate the preservation of the salient regions on the synthetic saliency map on two models, and a similar phenomenon can also be observed for both MS-MSLCE and MSSP-MSLCE.

- On the Dinosaur model, the MS method detects high saliency at the #1 region (in the blue rectangle) and the #4 region (in the black rectangle), and low saliency at the #2 and #3 regions (in the red rectangles) as shown in Fig. 5.2b. The MSSP method detects high saliency at the #1, #2 and #3 regions, and low saliency at the #4 region as shown in Fig. 5.2c. Finally, the synthetic saliency map MS-MSSP shows high saliency at the #1, #2, #3 and #4 regions in Fig. 5.12a.

- On the RockerArm model, the MS method detects high saliency at the #4 region (in the black rectangle) and low saliency at some parts of the #1, #2, and #3 regions (in the blue rectangles) as shown in Fig. 5.3b. The MSSP method detects generally high saliency at the #1, #2, and #3 regions and median saliency at the #4 region as shown in Fig. 5.3c. Finally, the synthetic saliency map MS-MSSP shows high saliency at the #1, #2, #3, and #4 regions as shown in Fig. 5.13a.

We list the performance values of the TPDMVS metric on the LIRIS/EPFL general-purpose database when incorporating either the individual saliency map or the synthetic saliency map into the metric in Table 5.9. From Table 5.9, we observe that any of the synthetic saliency maps achieves a performance gain over each individual saliency map, and the synthetic saliency map MS-MSLCE has the best performance among all the synthetic saliency maps. Among the three synthetic saliency maps that merge only two individual saliency maps, the performance gain achieved by the synthetic saliency map MS-MSLCE over the corresponding individual saliency maps (MS and MSLCE) is the greatest while the performance gain achieved by the synthetic saliency map MSSP-MSLCE over the corresponding individual saliency maps (MSSP and MSLCE) is the least. The analysis in Section 5.3.4 reveals that the similarity between the saliency maps of MS and MSLCE is the lowest while the similarity between the saliency maps of MSSP and MSLCE is the highest. So we conclude that there is a close correlation between the performance gain of the synthetic saliency map over individual saliency maps and the similarity between the individual saliency maps. Specifically, our abovementioned analysis with three saliency methods indicates that when the similarity between two individual saliency maps is lower, the performance gain of the synthetic saliency map over the individual saliency maps will be greater. From Table 5.9, we also observe that the synthetic saliency map MS-MSSP-MSLCE does not achieve better performance than the synthetic saliency map MS-MSLCE. The reason is that there is already a high similarity between the saliency maps of MSSP and MSLCE, and it is hard to achieve a performance gain over the synthetic saliency map MS-MSLCE by further merging MS-MSLCE with the saliency map of MSSP. Due to a lack of sufficient knowledge of human visual system [19, 28–32], it is still difficult to explain theoretically why the synthetic saliency map achieves better performance than individual saliency maps for mesh quality assessment. However, the experimental results and analysis in this section will facilitate the investigation on how visual attention or mesh saliency affects the perception of mesh quality and on the correlation analysis among different mesh saliency detection methods.

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

5.5 Summary

In this chapter, we have investigated the incorporation of the mesh saliency map into the mesh visual quality metric. We have built a saliency-based visual importance model to emphasize the impact of the salient regions on the mesh quality, and proposed the TPDMVS metric by applying the visual importance model in the Minkowski method. The saliency value is used as the weighting coefficient of the local distortion for each vertex at the stage of spatial pooling. We have demonstrated the effectiveness of the visual importance model and the performance superiority of the TPDMVS metric over state-of-the-art metrics with three well-known mesh saliency detection methods. The saliency-based visual importance model proposed in this chapter achieves better performance for the metric than the local distortion-based visual importance model proposed in Chapter 4. We propose a saliency map synthesis method by assembling the salient regions of different individual saliency maps, and evaluate the performance of the TPDMVS metric when incorporating the synthetic saliency map in the metric. The experimental results demonstrate that the synthetic saliency map achieves performance gain over the individual saliency maps for the TPDMVS metric, and the performance gain will be greater if the similarity between the individual saliency maps is lower.

We concluded that it is inappropriate to include the local surface area in the TPDMPW metric on the LIRIS/EPFL general-purpose database in Chapter 4. In this chapter, we draw a similar conclusion for the TPDMVS metric, that is, it is inappropriate to include the local surface area in the TPDMVS metric on the LIRIS/EPFL general-purpose database. These conclusions indicate that it is inappropriate to include the local surface area in the metric on the LIRIS/EPFL general-purpose database. This finding may facilitate the investigation on how the local surface area influences the overall quality of the distorted meshes with various types of distortions.

The investigations in Chapter 3 and Chapter 4 reveal that the severely distorted regions have a great impact on the mesh quality while the investigation in this chapter reveals that the salient regions also have a great impact on the mesh quality. Both the Minkowski method used in this chapter and the percentile weighting method proposed in Chapter 4 have the capability to emphasize the impact of the severely distorted regions on the mesh quality, as demonstrated in Chapter 4. In Chapter 6, we will further investigate the perceptual importance of both the severely distorted regions and the salient regions to the mesh quality with both the Minkowski method and the percentile weighting method.

Chapter 6

Mesh Visual Quality Metric Based on the Saliency-Aware Distortion

6.1 Introduction

Most state-of-the-art mesh visual quality metrics [1–6] evaluated the local distortions of the mesh by utilizing the geometric features, such as local curvature, local roughness, and dihedral angle. These metrics show good performance for mesh quality assessment although the vertex saliency is not involved in the evaluation of local distortions. Nouri et al. [62] proposed the Saliency-based Mesh Quality Index (SMQI), which computes the local structural distortions from the saliency map generated by the mesh saliency detection method [27]. The SMQI metric evaluates the local distortions without utilizing geometric features. In Chapter 5, we used the saliency value of each vertex as the weighting coefficient of the local distortion, and demonstrated the benefit of incorporating the saliency map in the TPDM metric [5] with three well-known mesh saliency detection methods [23–25]. Based on the research work in Chapter 5, we further investigate the incorporation of the saliency map in the metric and propose the saliency weighting strategy to combine the contributions of the severely distorted regions and the salient regions to the mesh quality in this chapter. We assume that the human eyes pay more attention to both the severely distorted regions and the salient regions on the mesh in the task of mesh quality assessment. We generate the saliency-aware distortion for each vertex through weighting the local distortion by the exponential result of the saliency value. We apply the Minkowski method and the percentile weighting method respectively to the saliency-aware distortions to emphasize both the impact of the severely distorted regions and the impact of salient regions on the mesh quality, and propose two metrics: TPDMSW and TPDMPWSW accordingly.

The relationship between this chapter and Chapter 5 can be recognized in three points: (1) In Chapter 5, we proposed the TPDMVS metric [33] by using the saliency value to weight the exponential result of the local distortion for each vertex, as shown in Eq. (5.1). The contribution of vertex saliency is fixed in the TPDMVS metric. However, in this chapter, we weight the local distortion by the exponential result of the saliency value and generate the saliency-aware distortion for each vertex. The exponent is used to adjust the contribution of vertex saliency. (2) In this chapter, we develop the TPDMSW metric by applying the Minkowski method to the saliency-aware distortions. The TPDMSW metric is the generalized version of the TPDMVS metric that was proposed in Chapter 5. The TPDMSW metric will turn into the TPDMVS metric when the exponent is set to be a specific value. We investigate how the value of the exponent influences the performance of the TPDMSW metric in this chapter. (3) We proposed the percentile weighting method for the mesh visual quality metric and demonstrated the capability of the percentile weighting method to emphasize the impact of the severely distorted regions on the mesh quality in Chapter 4. In this chapter, we apply the percentile weighting method to the saliency-aware distortions and propose the TPDMPWSW metric. We compare the performance of the TPDMPWSW metric with the performance of the TPDMVS metric.

The remainder of this chapter is organized as follows: In Section 6.2, we introduce the saliency weighting strategy. We apply both the Minkowski method and the percentile weighting method to the saliency-aware distortions, and propose two metrics: TPDMSW and TPDMPWSW accordingly. We investigate how the value of the exponent influences the performance of the metrics, and suggest the appropriate range of value for the exponent in the metrics. In Section 6.3, we evaluate the performance of the TPDMPWSW metric and analyze the experimental results. We summarize this chapter in Section 6.4.

6.2 Saliency-Aware Distortion

6.2.1 Saliency Weighting Strategy

We propose the saliency weighting strategy to combine the contributions of both the severely distorted regions and the salient regions to the mesh quality. The contribution of each vertex to the quality score of the mesh is determined based on both the local distortion and the saliency value of the vertex. We generate the saliency-aware distortion for each vertex through weighting the local distortion by the exponential result of the saliency value:

$$d_i' = s_i^{\gamma} \cdot d_i, \tag{6.1}$$

where d_i ($0 \leq d_i \leq 1$) is the local distortion of vertex v_i in the reference mesh, s_i ($0 \leq s_i \leq 1$) is the saliency value of vertex v_i, d_i' is the saliency-aware distortion of vertex v_i. A large value for d_i indicates vertex v_i is severely distorted while a small value for d_i indicates vertex v_i is slightly or not distorted. A large value for s_i indicates vertex v_i is salient while a small value for s_i indicates vertex v_i is not salient. The exponent γ ($0 \leq \gamma \leq 1$) is used to adjust the contribution of vertex saliency to the saliency-aware distortion. By applying either the Minkowski method or the percentile weighting method to the saliency-aware distortions, we will be able to emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality. As the value of the exponent γ increases, the contribution of vertex saliency to the saliency-aware distortion will increase and thus more emphasis will be put on the impact of salient regions. $\gamma = 0$ indicates no emphasis on the impact of the salient regions while $\gamma = 1$ indicates full emphasis on the impact of the salient regions.

When we aggregate the local distortions with either the Minkowski method or the percentile weighting method, the severely distorted vertices will have greater contributions to the quality score than the moderately vertices regardless of the saliency values of the vertices. However, when we aggregate the saliency-aware distortions into the quality score, the contribution of the vertex depends on both the local distortion and the saliency value of the vertex, which indicates that the moderately distorted vertices with high saliency values may have greater contributions to the quality score than the severely distorted vertices with low saliency values.

In this chapter, we compute the local distortion map between the reference mesh and the distorted mesh with the TPDM metric [5], and generate the saliency maps for the reference meshes with the mesh saliency detection methods [23–25]. We apply the Minkowski method to the saliency-aware distortions and propose the TPDMSW metric in Section 6.2.2. We apply the percentile weighting method to the saliency-aware distortions and propose the TPDMPWSW metric in Section 6.2.3. We investigate how the value of the exponent γ influences the performance of each metric on the LIRIS/EPFL general-purpose database respectively in each section.

6.2.2 Applying the Minkowski Method to the Saliency-Aware Distortions

In Chapter 5, we proposed the TPDMVS metric by incorporating the saliency map of the mesh in the TPDM metric [5]. As shown in Eq. (5.1), the Minkowski method was employed by the TPDMVS metric at the stage of spatial pooling to emphasize the impact of the severely distorted regions on the mesh quality. In this section, we apply the Minkowski method to the saliency-aware distortions to emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality. Accordingly, we develop the TPDMSW metric as follows:

$$TPDMSW = \left(\frac{1}{N} \sum_{i=1}^{N} (s_i{}^{\gamma} \cdot d_i)^q \right)^{\frac{1}{q}}, \tag{6.2}$$

where N is the number of vertices in the reference mesh, and q is the Minkowski exponent which is set to be 4, the same value as in the TPDMVS metric. d_i is the local distortion of vertex v_i in the reference mesh, and s_i is the saliency value of vertex v_i. The TPDMSW metric is the generalized version of the TPDMVS metric [33], and will turn into the TPDMVS metric in the case of $\gamma = 0.25$.

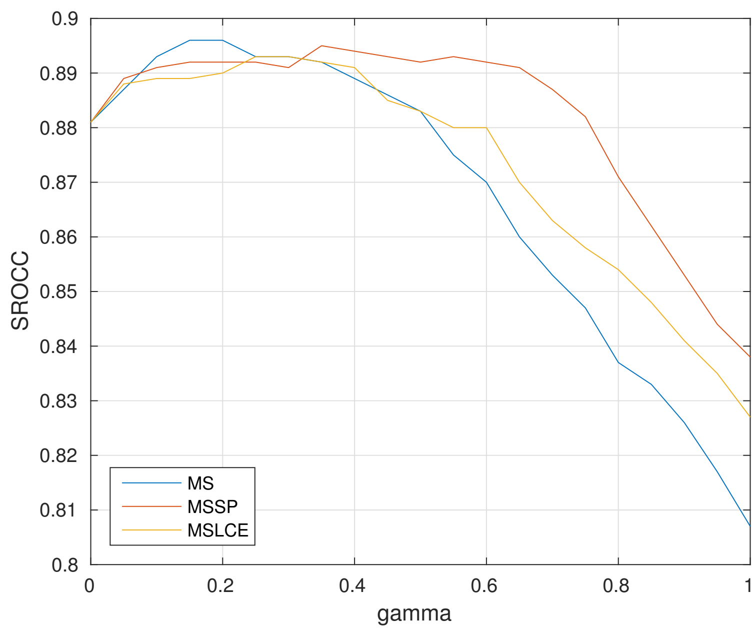


Figure 6.1 : The plot of the SROCC performance value of the TPDMSW metric with each of three mesh saliency detection methods as a function of the exponent γ on the LIRIS/EPFL general-purpose database.

We investigate how the value of the exponent γ influences the performance of the TPDMSW metric through the experimental tests. We assign different values to the exponent γ, and compute the corresponding SROCC performance values of the TPDMSW metric on the LIRIS/EPFL general-purpose database [1] with each of three mesh saliency detection methods [23–25]. Fig. 6.1 illustrates how the SROCC performance value of the TPDMSW metric changes as the value of the exponent γ varies within the range [0, 1] with a step-size of 0.05. From Fig. 6.1, we observe that when the value of the exponent γ starts to increase from 0, the SROCC performance value of the TPDMSW metric increases for any of three mesh saliency detection methods. This indicates that the emphasis on the impact of the salient regions improves the performance of the metric. For any of three mesh saliency detection methods, the SROCC performance value of the TPDMSW metric reaches to a relatively stable level when the value of the exponent γ lies within the range [0.25, 0.4], and declines in overall when the value of the exponent γ increases from 0.45 to 1 gradually. This indicates that an overemphasis on the impact of the salient regions may cause a performance degradation to the metric. It is critical to choose an appropriate value for the exponent γ, and a good value for the exponent γ lies within the range [0.25, 0.4].

6.2.3 Applying the Percentile Weighting Method to the Saliency-Aware Distortions

We proposed the percentile weighting method and demonstrated its capability to emphasize the impact of the severely distorted regions on the mesh quality in Chapter 4. When the percentile weighting method is applied in the TPDMPW metric [21], as we described in Section 4.2.1, all the local distortion elements of the vertices are partitioned into two distortion sets: A and B. The local distortion elements in the distortion set A are overall greater than the local distortion elements in the distortion set B. Since the severely distorted region is not necessarily the salient region on the mesh, the vertex with a large local distortion value may not have a large saliency value. The saliency-aware distortion value d_i' of vertex v_i in the distortion set A may not necessarily be greater than the saliency-aware distortion value d_j' of vertex v_j in the set B. If we simply assign a greater weight to the saliency-aware distortions of the vertices in the distortion set A, the severely distorted regions with low saliency would be emphasized while the moderately distorted regions with high saliency would not be emphasized. This contradicts with the saliency weighting strategy. Thus, in this section, we make an adaptation to the partitioning process of the percentile weighting method and propose the TPDMPWSW metric by applying the percentile weighting method to the saliency-aware distortions. The flowchart of the TPDMPWSW metric is illustrated in Fig. 6.2.

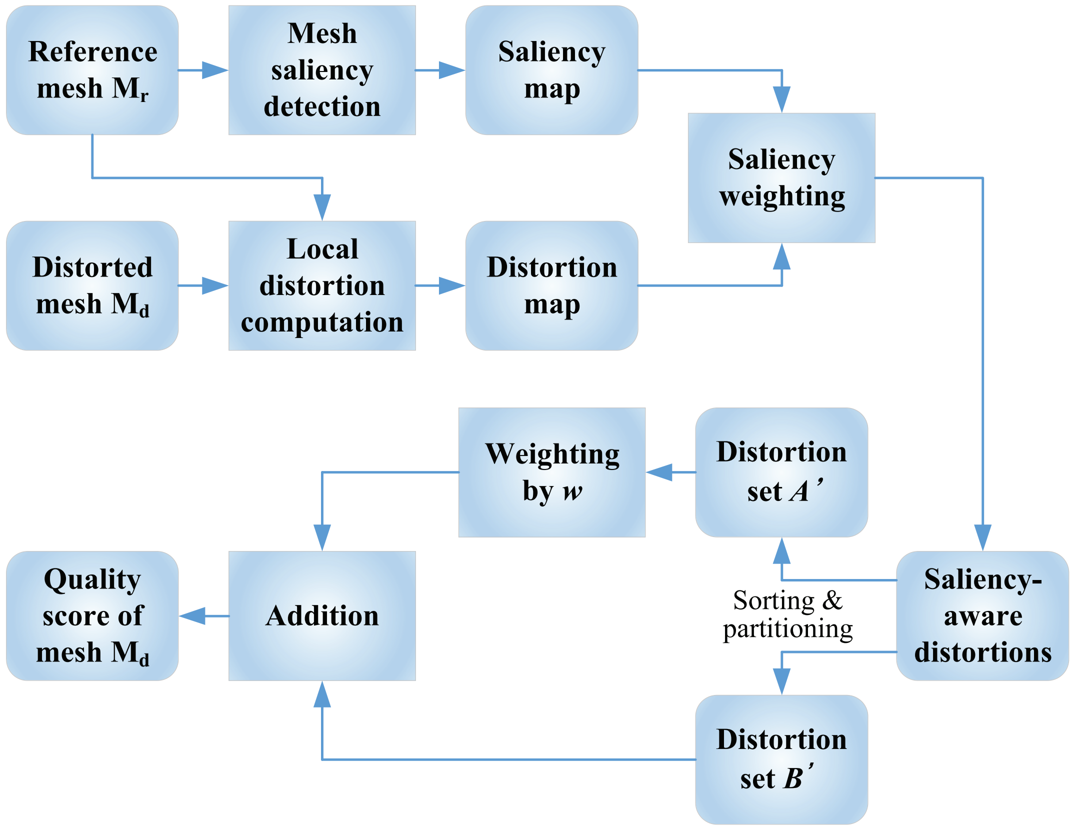


Figure 6.2 : The flowchart of the TPDMPWSW metric.

Given a reference mesh and a distorted mesh, we firstly compute the local distortion map between two meshes with the TPDM metric [5], and generate a saliency map for the reference mesh with a mesh saliency detection method [23–25]. Secondly, we use the saliency weighting strategy defined in Eq. (6.1) to compute the saliency-aware distortion for each vertex in the reference mesh, and construct the saliency-aware distortion vector D' by sorting all the saliency-aware distortion elements in the descending order. Thirdly, we partition the saliency-aware distortion vector D' into two distortion sets: A' and B'. The saliency-aware distortion elements in the distortion set A' rank within the top p percentile of the saliency-aware distortion vector D' while the saliency-aware distortion elements in the distortion set B' rank beyond the p percentile of the saliency-aware distortion vector D'. We assign a greater weight to the saliency-aware distortion elements in the distortion set A' to emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality. Finally, we aggregate all the saliency-aware distortion elements through the addition operation, and take the normalized addition result as the quality score of the distorted mesh. The TPDMPWSW metric is formulated as follows:

$$TPDMPWSW = \frac{1}{Npw}\left(\sum_{d'_i \in A'} w \cdot d'_i + \sum_{d'_j \in B'} d'_j\right), \tag{6.3}$$

where $d'_i = s_i^{\gamma} \cdot d_i$ is the saliency-aware distortion of vertex v_i in the distortion set A', and $d'_j = s_j^{\gamma} \cdot d_j$ is the saliency-aware distortion of vertex v_j in the distortion set B'. d_i and d_j are the local distortion values of the vertices v_i and v_j respectively. s_i and s_j are the saliency values of the vertices v_i and v_j respectively. N is the number of vertices in the reference mesh, p is the percentile, and w is the weight assigned to the saliency-aware distortion elements in the distortion set A'. The values of the parameters p and w are set to be the unified values: $w = 60$, $p = 0.06$, as recommended in Section 4.3.3. Since it is inappropriate to include the local surface area in the metric on the LIRIS/EPFL general-purpose database as demonstrated in Chapter 4 and Chapter 5, we do not include the local surface area in the TPDMPWSW metric. In the case of $\gamma = 0$, the TPDMPWSW metric will turn into the TPDMPW metric [21] with $\alpha = 0$, which was introduced in Chapter 4.

We compute the SROCC performance value of the TPDMPWSW metric with each of three mesh saliency detection methods [23–25] on the LIRIS/EPFL general-purpose database when we change the value of the exponent γ. Fig. 6.3 illustrates how the SROCC performance value of the TPDMPWSW metric changes as the value of the exponent γ varies within the range [0, 1] with a step-size of 0.05. The impact of the salient regions

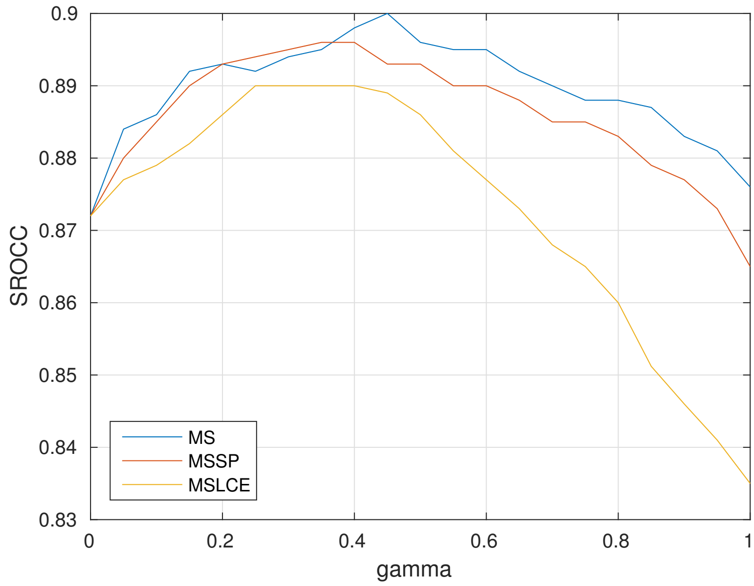


Figure 6.3 : The plot of the SROCC performance value of the TPDMPWSW metric with each of three mesh saliency detection methods as a function of the exponent γ on the LIRIS/EPFL general-purpose database.

will not be emphasized in the case of $\gamma = 0$. From Fig. 6.3, we observe that the SROCC performance value of the TPDMPWSW metric increases for any of three mesh saliency detection methods when the value of the exponent γ starts to increase from 0. This observation is similar with the observation obtained from Fig. 6.1, where the SROCC performance value of the TPDMSW metric also increases when the value of the exponent γ starts to increase from 0. From Fig. 6.3, we also observe that, for any of three mesh saliency detection methods, the SROCC performance value of the TPDMPWSW metric reaches to a relatively stable level when the value of the exponent γ lies within the range [0.25, 0.4] and then declines in overall when the value of the exponent γ continues to increase from 0.45 to 1 gradually. This observation is similar with the observation obtained from Fig. 6.1, where the SROCC performance value of the TPDMSW metric reaches to an overall high value when the value of the exponent γ lies within the range [0.25, 0.4] and then declines in overall when the value of the exponent γ continues to increase from 0.45 to 1 gradually. Though the TPDMPWSW and TPDMSW metrics employ different methods to aggregate the saliency-aware local distortions, they show generally good performance when the value of the exponent γ lies within the range [0.25, 0.4]. Based on the aforementioned observations, we conclude that the emphasis on the impact of salient regions on the mesh quality contributes to improving the performance of the metric, but an overemphasis on the impact of salient regions may cause a performance degradation to the metric. A good value for the exponent γ in the saliency weighting strategy lies within the range [0.25, 0.4].

Since the vertex saliency and the local distortion are multiplied together to generate the saliency-aware distortion of each vertex as shown in Eq. (6.1), both vertex saliency and local distortion have contributions to the saliency-aware distortion. The exponent γ not only regulates the contribution of vertex saliency to the saliency-aware distortion, but also affects the contribution of local distortion relative to the vertex saliency. As the value of the exponent γ increases, the contribution of vertex saliency to the saliency-aware distortion will increase and the contribution of local distortion relative to the vertex saliency will decrease, and thus more emphasis will be put on the salient regions while relatively less emphasis will be put on the distorted regions. An extreme case is that the slightly distorted regions with high saliency may have a greater impact on the mesh quality than the severely distorted regions with low saliency, which may contradict with the visual perception of mesh quality. We believe that though the salient regions have a great impact on the overall mesh quality, the vertex saliency and the local distortion should be combined in an appropriate way in order to achieve optimal performance for the metric. Either an overemphasis or a neglect on the salient regions would lead to performance degradation to some extent.

6.3 Experimental Results and Analysis

6.3.1 Experimental Protocol

We have proposed two metrics: TPDMSW and TPDMPWSW in this chapter. As demonstrated in Section 6.2.2 and Section 6.2.3, both the TPDMSW metric and the TPDMPWSW metric show relatively stable and good performance when the value of the exponent γ lies within the range [0.25, 0.4]. In Section 5.4, we already validated the performance of the TPDMVS metric, which is a representative version of the TPDMSW metric at $\gamma = 0.25$. In this section, we evaluate the performance of the TPDMPWSW metric at $\gamma = 0.3$ on the LIRIS/EPFL general-purpose database [1]. We generate the saliency map for the reference mesh with each of three mesh saliency detection methods [23–25]. We compute the quality scores for the distorted meshes with the TPDMPWSW metric, and then calculate the PLCC and SROCC coefficients between the quality scores and the MOSs of the meshes. We evaluate the performance of the TPDMPWSW metric by comparing the performance values of the TPDMPWSW metric with state-of-the-art mesh visual quality metrics. Before calculating the correlation coefficients, we perform a psychometric fitting between the quality scores and MOSs with the cumulative Gaussian function defined in Eq. (2.5) to remove the nonlinearity [84].

6.3.2 Performance Comparison

We compare the performance of the TPDMPWSW metric with the performance of state-of-the-art MVQ metrics, including Hausdorff Distance (HD) [49], Root Mean Square Error (RMS) [49], Geometric Laplacian (GL1, GL2) [43, 44], Strain Field (SF) [61], 3DWPM1 [60], 3DWPM2 [60], MSDM [1], MSDM2 [2], FMPD [4], DAME [3], TPDM [5], Dong [6], and TPDMPW [21]. Table 6.1 lists the PLCC and SROCC performance values of the TPDMPWSW metric and state-of-the-art MVQ metrics on the LIRIS/EPFL general-purpose database. In Table 6.1, we list the optimal performance values of the

TPDMPW metric on the LIRIS/EPFL general-purpose database. TPDMPWSW(MS) indicates the performance of the TPDMPWSW metric with the MS method [23], TPDMPWSW(MSSP) indicates the performance of the TPDMPWSW metric with the MSSP method [24], and TPDMPWSW(MSLCE) indicates the performance of the TPDMPWSW metric with the MSLCE method [25].

Table 6.1 : The performance values (%) of the TPDMPWSW metric and state-of-the-art MVQ metrics on the LIRIS/EPFL general-purpose database.

Metric	PLCC	SROCC
HD	11.4	13.8
RMS	28.1	26.8
GL1	35.5	33.1
GL2	42.4	39.3
SF	7.0	15.7
3DWPM1	61.8	69.3
3DWPM2	49.6	49.0
MSDM	75.0	73.9
MSDM2	81.4	80.4
FMPD	83.5	81.9
DAME	75.2	76.6
TPDM	84.1	84.3
Dong	87.7	86.6
TPDMPW	87.7	87.2
TPDMPWSW(MS)	89.2	89.4
TPDMPWSW(MSSP)	89.2	89.5
TPDMPWSW(MSLCE)	88.9	89.0

From Table 6.1, we observe that when incorporating the saliency map generated by any of three mesh saliency detection methods [23–25] in the metric, the TPDMPWSW metric achieves significant performance gain over the TPDMPW metric [21] and better performance than other state-of-the-art MVQ metrics. The performance superiority of the TPDMPWSW metric demonstrates the effectiveness of the saliency weighting strategy and verifies the assumption that we made in Section 6.1. Note that the TPDMPW metric applies the percentile weighting method to the local distortions while the TPDMPWSW metric applies the percentile weighting method to the saliency-aware distortions. Both the TPDMPWSW metric and TPDMPW metric use the unified values for the parameters w and p of the percentile weighting method. The performance superiority of the TPDM-PWSW metric confirms the effectiveness of both the percentile weighting method and the unified values for the parameters w and p of the percentile weighting method. For each of three mesh saliency detection methods, we perform a psychometric fitting between the quality scores computed by the TPDMPWSW metric and the MOSs of the meshes with the cumulative Gaussian function. Fig. 6.4, Fig. 6.5, and Fig. 6.6 illustrate the scatter plots of the MOSs versus the quality scores computed by the TPDMPWSW metric with each of three mesh saliency detection methods and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database. From Fig. 6.4, Fig. 6.5, and Fig. 6.6, we observe that the $QualityScore\text{-}MOS$ pairs are fitted well by the psychometric curve for each of three mesh saliency detection methods.

Both the TPDMPWSW and TPDMVS [33] metrics apply the saliency weighting strategy to emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality, but they employ different methods to aggregate the saliency-aware distortions. We compare the performance of the TPDMPWSW and TPDMVS [33] metrics with the performance of the TPDMPW [21] and TPDM [5] metrics, which do not apply the saliency weighting strategy. Table 6.2 lists the performance values of the TPDM, TPDMSP, TPDMPW, TPDMVS, and TPDMPWSW metrics on the LIRIS/EPFL general-purpose database. From Table 6.2, we observe that, for any of three

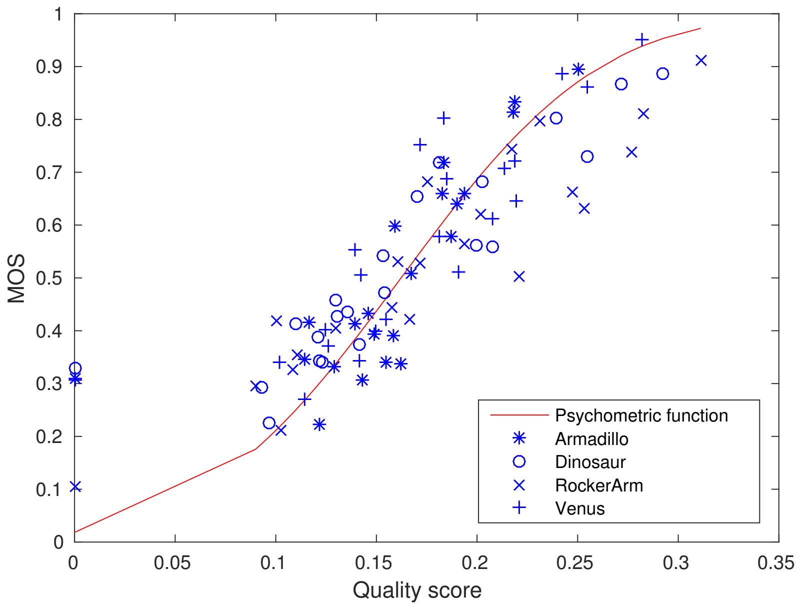


Figure 6.4 : The scatter plots of the MOSs versus the quality scores computed by the TPDMPWSW(MS) metric and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database.

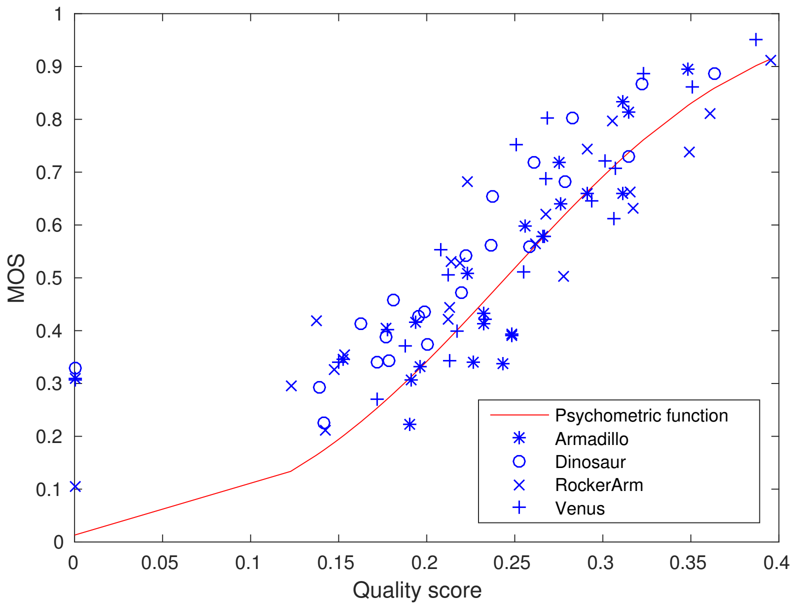


Figure 6.5 : The scatter plots of the MOSs versus the quality scores computed by the TPDMPWSW(MSSP) metric and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database.

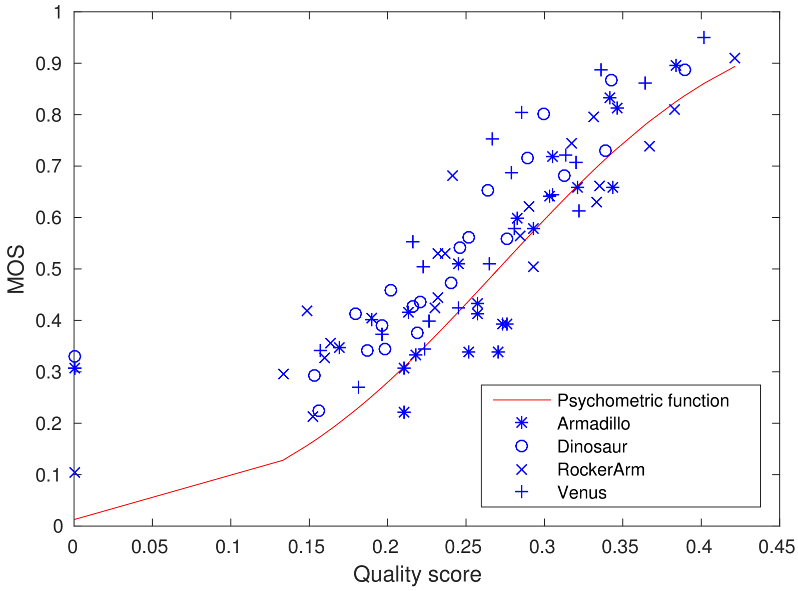


Figure 6.6 : The scatter plots of the MOSs versus the quality scores computed by the TPDMPWSW(MSLCE) metric and the fitted psychometric curve for the meshes in the LIRIS/EPFL general-purpose database.

Table 6.2 : The performance values (%) of the TPDM, TPDMSP, TPDMPW, TPDMVS, and TPDMPWSW metrics on the LIRIS/EPFL general-purpose database.

Metric		PLCC	SROCC
TPDM		84.1	84.3
TPDMSP		89.0	87.5
TPDMPW		87.7	87.2
MS	TPDMVS	89.0	89.3
	TPDMPWSW	89.2	89.4
MSSP	TPDMVS	89.6	89.2
	TPDMPWSW	89.2	89.5
MSLCE	TPDMVS	89.4	89.3
	TPDMPWSW	88.9	89.0

mesh saliency detection methods [23–25], the TPDMPWSW metric achieves significantly better performance than the TPDMPW metric while the TPDMVS metric achieves significantly better performance than the TPDM metric. The TPDMPWSW metric achieves comparable performance with the TPDMVS metric for three mesh saliency detection methods. This indicates that the saliency weighting strategy can improve the performance of the metric when either the Minkowski method or the percentile weighting method is applied at the stage of spatial pooling.

In Chapter 3, Chpater 4 and Chapter 5, we proposed the TPDMSP, TPDMPW, and TPDMVS metrics, respectively. Here we compare the performance of these three metrics and the TPDMPWSW metric. Among these four metrics, the TPDMSP metric follows the machine learning paradigm while the other three metrics are expressed in explicit formulas. In Table 6.2, the performance values of the TPDMSP metric represent the median performance of the TPDMSP metric on the LIRIS/EPFL general-purpose database, as reported in Section 3.5.2. From Table 6.2, we observe that the TPDMPWSW and TPDMVS metrics achieve generally better performance than the TPDMSP metric while the TPDMSP metric achieves better performance than the TPDMPW metric. In practical applications of mesh quality evaluation, we recommend using either the TPDMPWSW metric or the TPDMVS metric to pursue the highest accuracy.

6.3.3 Synthetic Saliency Map

In Section 5.4.4, we evaluated the performance of the TPDMVS metric when incorporating the synthetic saliency map in the metric. We concluded that the synthetic saliency map achieves performance gain for the TPDMVS metric over the individual saliency maps, and the performance gain will be greater when the similarity between two individual saliency maps is lower. In this section, we evaluate the performance of the TPDMPWSW metric when incorporating the synthetic saliency map in the metric, and analyze if the aforementioned conclusions drawn for the TPDMVS metric also hold for the TPDMPWSW metric.

Table 6.3 : The performance values of the TPDMPWSW metric on the LIRIS/EPFL general-purpose database when incorporating either the individual saliency map or the synthetic saliency map in the metric.

Saliency map	PLCC (%)	SROCC (%)
MS	89.2	89.4
MSSP	89.2	89.5
MSLCE	88.9	89.0
MS-MSSP	89.7	90.5
MS-MSLCE	90.0	90.8
MSSP-MSLCE	89.4	89.6
MS-MSSP-MSLCE	89.8	91.1

We list the performance values of the TPDMPWSW metric on the LIRIS/EPFL general-purpose database when incorporating either the individual saliency map or the synthetic saliency map in the metric in Table 6.3. From Table 6.3, we observe that any of the synthetic saliency maps achieves a performance gain over the corresponding individual saliency maps. Among the three synthetic saliency maps that merge only two individual saliency maps, the performance gain achieved by the synthetic saliency map MS-MSLCE over the corresponding individual saliency maps (MS and MSLCE) is the greatest while the performance gain achieved by the synthetic saliency map MSSP-MSLCE over the corresponding individual saliency maps (MSSP and MSLCE) is the least. So the observations obtained from Table 6.3 for the TPDMPWSW metric are similar as the observations obtained in Section 5.4.4 for the TPDMVS metic. From Table 6.3, we also observe that the synthetic saliency map MS-MSSP-MSLCE does not achieve better performance than the synthetic saliency map MS-MSLCE. The reason is that, as we explained in Section 5.4.4, there is already a high similarity between the saliency maps of MSSP and MSLCE and it is hard to achieve a performance gain over the synthetic saliency map MS-MSLCE by further merging MS-MSLCE with the saliency map of MSSP. Based on the similar observations obtained for both the TPDMPWSW metric and TPDMVS metric, we confirm the conclusions that we drew in Section 5.4.4, that is, the synthetic saliency map achieves performance gain for the metric over the corresponding individual saliency maps, and the performance gain will be greater when the similarity between two individual saliency maps is lower.

6.4 Summary

In this chapter, we have proposed the saliency weighting strategy to combine the contributions of the severely distorted regions and the salient regions to the mesh quality. We generate the saliency-aware distortion for each vertex through weighting the local distortion by the exponential result of the saliency value. We have proposed the TPDMSW metric and the TPDMPWSW metric by applying the Minkowski method and the percentile weighting method respectively to emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality. The TPDMSW metric is the generalized version of the TPDMVS metric proposed in Chapter 5. The experimental results demonstrate that an emphasis on the impact of the salient regions contributes to improving the performance of the metric, but an overemphasis on the impact of salient regions may cause a performance degradation to the metric. We have suggested the range of value for the exponent in the saliency weighting strategy through the experimental tests. The experimental results demonstrate the effectiveness of the saliency weighting strategy and the performance superiority of the TPDMPWSW metric. We evaluate the performance of the TPDMPWSW metric when incorporating the synthetic saliency map in the metric. The experimental results verify the conclusions drawn in Chapter 5, that is, the synthetic saliency map achieves performance gain for the metric over the corresponding individual saliency maps, and the performance gain will be greater when the similarity between two individual saliency maps is lower. The study in this chapter and Chapter 5 will facilitate the investigation on how mesh saliency is associated with the subjective judgment of mesh quality in a theoretical way.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

We have investigated the perceptual quality metrics of 3D meshes in this thesis. We introduced the research background, research questions and research objectives in Chapter 1, and reviewed the related work and literature in Chapter 2. We investigated the perceptual importance of the severely distorted regions and the salient regions to the mesh quality in Chapter 3, Chapter 4, Chapter 5, and Chapter 6. We demonstrated that the severely distorted regions have a great impact on the mesh quality in Chapter 3, proposed the percentile weighting method to emphasize the impact of the severely distorted regions on the mesh quality in Chapter 4, built a saliency-based visual importance model to emphasize the impact of the salient regions on the mesh quality in Chapter 5, and proposed the saliency weighting strategy to emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality in Chapter 6. Based on the investigations, we conclude that an appropriate emphasis on both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality contributes to improving the performance of the mesh visual quality metric. The experimental results in Chapter 4 and Chapter 5 demonstrated that it is inappropriate to include the local surface area of the vertex in the metric on the LIRIS/EPFL general-purpose database. We have proposed five full-reference mesh visual quality metrics: TPDMSP, TPDMPW, TPDMVS, TPDMSW, and TPDMPWSW. The findings of this thesis are summarized as follows:

(1) Mesh quality prediction based on the statistical descriptors

We investigated the relationship between the statistical distribution of local distortions and the overall quality of the distorted mesh, and proposed the TPDMSP metric based on the statistical descriptors. This metric extracts the statistical descriptors from the local distortions as the feature vector to represent the mesh quality. We trained the support vector regression model to learn a function that maps from the feature vector to the quality score of the distorted mesh. The experiments on three databases demonstrated the performance superiority of the TPDMSP metric over state-of-the-art metrics, and the good generalization capability of the TPDMSP metric among different models and different databases. The TPDMSP metric shows good performance for mesh quality prediction even with a small amount of training samples. The experimental results demonstrated that the statistical descriptors of local distortions reflect the mesh quality well, and the severely distorted regions on the mesh have a greater impact on the mesh quality.

(2) Spatial pooling with the percentile weighting strategy

We proposed a spatial pooling method with the percentile weighting strategy to emphasize the impact of severely distorted regions on the mesh quality. We built a visual importance model by assigning a greater weight to the vertices with large local distortions at the stage of spatial pooling. The local surface area was used to weight the local distortion in order to reflect the influence of the local surface area on the mesh quality. We proposed the TPDMPW metric by applying the percentile weighting method to aggregate the local distortions. We investigated the influence of the parameters of the percentile weighting method on the performance of the metric, and determined the optimal values and unified values for the parameters through empirical tests on three databases. The experimental results demonstrated the effectiveness of the percentile weighting method, and the performance superiority of the TPDMPW metric over state-of-the-art metrics.

(3) Mesh quality assessment by incorporating mesh saliency

We incorporated the saliency map in the metric and built a saliency-based visual importance model to emphasize the impact of the salient regions on the mesh quality. We assigned the saliency value as the weighting coefficient of the local distortion at the stage of spatial pooling, and proposed the TPDMVS metric. The TPDM metric was used to compute the local distortions while three well-known mesh saliency detection methods were employed to generate the saliency maps. We proposed a saliency map synthesis method by assembling the salient regions of different individual saliency maps. The synthetic saliency map achieved performance gain for the metric over the individual saliency maps, and the performance gain depends on the similarity between the individual saliency maps. The experimental results demonstrated the effectiveness of the saliency-based visual importance model and the performance superiority of the TPDMVS metric.

(4) Mesh visual quality metric based on the saliency-aware distortion

We proposed the saliency weighting strategy to combine the contributions of the severely distorted regions and the salient regions to the mesh quality. We generated a saliency-aware distortion for each vertex through weighting the local distortion by the exponential result of the saliency value. We proposed the TPDMSW metric, which is the generalized version of the TPDMVS metric, and the TPDMPWSW metric by applying the Minkowski method and the percentile weighting method respectively to the saliency-aware distortions. These two metrics emphasize both the impact of the severely distorted regions and the impact of the salient regions on the mesh quality. The experimental results demonstrated that an emphasis on the impact of the salient regions improves the performance of the metric, but an overemphasis on the impact of salient regions may cause a performance degradation to the metric. The experimental results demonstrated the effectiveness of the saliency weighting strategy and the performance superiority of the TPDMPWSW metric.

7.2 Future Work

We extracted the statistical descriptors from the local distortions as the feature vector to represent the quality score of the mesh in this thesis. The experimental results revealed that the statistical descriptors of the local distortions reflect the mesh quality well, but the interplay between the seven statistical descriptors requires further investigation. We have proposed the percentile weighting method for the mesh visual quality metric in the spatial domain. The percentile weighting method can be generalized to the spatiotemporal domain, and facilitate the investigation of spatiotemporal pooling methods in the dynamic mesh quality metric. The percentile weighting strategy can be used to emphasize the impact of spatiotemporal regions with severe local distortions on the quality of the dynamic mesh. We have investigated the incorporation of mesh saliency in the mesh quality assessment, and the research works may promote the development of better perceptual mesh visual quality metrics. We have proposed five MVQ metrics for the mesh quality assessment. All the metrics proposed in this thesis employed the TPDM metric to compute the local distortions of the mesh, but the statistical descriptors, the percentile weighting method and the saliency weighting strategy proposed in this thesis can be generalized to other MVQ metrics that are based on the computation of the local quality map.

The future work of this thesis includes the following parts:

- Since the number of mesh samples in the existing subjective databases for mesh quality assessment is relatively small, we plan to build a larger subjective database that involves more 3D geometric models, various types of distortions and more distortion levels. We will further investigate the impact of local surface area on the overall quality of distorted meshes with various types of distortions.

- We plan to investigate the no-reference mesh visual quality metric by utilizing machine learning methods. When a larger subjective database with more mesh samples is available in the future, machine learning or deep learning methods may play a significant role in the design of no-reference mesh visual quality metrics.

- It will be interesting to investigate the incorporation of visual attention in the mesh visual quality metric. We will evaluate the performance difference between visual attention and mesh saliency by incorporating either the visual fixation map or the saliency map in the mesh visual quality metric.

- It will be interesting to investigate more comprehensive methods, apart from the PLCC and SROCC coefficients, for evaluating the performance of mesh visual quality metrics. We plan to evaluate the reliability and the distinguishing capability of the mesh visual quality metric by constructing the ambiguity intervals of the metric at different quality levels.

- It will be interesting to study the visual perception of colorimetric distortion in the colored meshes. We plan to investigate the collaborative impact of both geometric distortion and colorimetric distortion on the quality of colored meshes, and develop a quality metric for the colored meshes.

Bibliography

[1] Guillaume Lavoué, Elisa Drelie Gelasca, Florent Dupont, Atilla Baskurt, and Touradj Ebrahimi. Perceptually driven 3D distance metrics with application to watermarking. In *Proceedings of SPIE Electronic Imaging*, pages 63120L.1–63120L.12. International Society for Optics and Photonics, 2006.

[2] Guillaume Lavoué. A multiscale metric for 3D mesh visual quality assessment. *Computer Graphics Forum*, 30(5):1427–1437, 2011.

[3] Libor Váša and Jan Rus. Dihedral angle mesh error: a fast perception correlated distortion measure for fixed connectivity triangle meshes. *Computer Graphics Forum*, 31(5):1715–1724, 2012.

[4] Kai Wang, Fakhri Torkhani, and Annick Montanvert. A fast roughness-based approach to the assessment of 3D mesh visual quality. *Computers & Graphics*, 36(7):808–818, 2012.

[5] Fakhri Torkhani, Kai Wang, and Jean-Marc Chassery. A curvature-tensor-based perceptual quality metric for 3D triangular meshes. *Machine Graphics & Vision*, 23(1):59–82, 2014.

[6] Lu Dong, Yuming Fang, Weisi Lin, and Hock Soon Seah. Perceptual quality assessment for 3D triangle mesh based on curvature. *IEEE Transactions on Multimedia*, 17(12):2174–2184, 2015.

[7] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[8] Qiaohong Li, Yu-Ming Fang, and Jing-Tao Xu. A novel spatial pooling strategy for image quality assessment. *Journal of Computer Science and Technology*, 31(2):225–234, 2016.

[9] Guillaume Lavoué and Rafał Mantiuk. Quality assessment in computer graphics. In Chenwei Deng, Lin Ma, Weisi Lin, and King Ngi Ngan, editors, *Visual Signal Quality Assessment*, pages 243–286. Springer, 2015.

[10] Paolo Gastaldo, Rodolfo Zunino, and Judith Redi. Supporting visual quality assessment with machine learning. *EURASIP Journal on Image and Video Processing*, 2013(1):54:1–54:15, 2013.

[11] Manish Narwaria and Weisi Lin. Objective image quality assessment based on support vector regression. *IEEE Transactions on Neural Networks*, 21(3):515–519, 2010.

[12] Manish Narwaria and Weisi Lin. Svd-based quality metric for image and video using machine learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):347–364, 2012.

[13] Andrew Hines, Paul Kendrick, Adriaan Barri, Manish Narwaria, and Judith A Redi. Robustness and prediction accuracy of machine learning for objective visual quality assessment. In *Proceedings of the Twenty-Second European Signal Processing Conference (EUSIPCO)*, pages 2130–2134. IEEE, 2014.

[14] Paolo Gastaldo and Judith A Redi. Machine learning solutions for objective visual quality assessment. In *Proceedings of the Sixth International Workshop on Video Processing and Quality Metrics for Consumer Electronics, VPQM*, volume 12, pages 347–364, 2012.

[15] Long Xu, Weisi Lin, and C-C Jay Kuo. *Visual quality assessment by machine learning*. Springer, 2015.

[16] Guillaume Lavoué, Irene Cheng, and Anup Basu. Perceptual quality metrics for 3D meshes: towards an optimal multi-attribute computational model. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3271–3276. IEEE, 2013.

[17] Zhou Wang and Qiang Li. Information content weighting for perceptual image quality assessment. *IEEE Transactions on Image Processing*, 20(5):1185–1198, 2011.

[18] Zhou Wang and Xinli Shang. Spatial pooling strategies for perceptual image quality assessment. In *Proceedings of IEEE International Conference on Image Processing*, pages 2945–2948. IEEE, 2006.

[19] Anush Krishna Moorthy and Alan Conrad Bovik. Visual importance pooling for image quality assessment. *IEEE Journal of Selected Topics in Signal Processing*, 3(2):193–201, 2009.

[20] Xiang Feng, Wanggen Wan, Richard Yi Da Xu, Haoyu Chen, Pengfei Li, and J. Alfredo Sánchez. A perceptual quality metric for 3D triangle meshes based on spatial pooling. *Frontiers of Computer Science*, 12(4):798–812, August 2018.

[21] Xiang Feng, Wanggen Wan, Richard Yi Da Xu, Stuart Perry, Pengfei Li, and Song Zhu. A novel spatial pooling method for 3D mesh quality assessment based on percentile weighting strategy. *Computers & Graphics*, 74:12–22, 2018.

[22] Xianyong Liu, Ligang Liu, Weijie Song, Yanping Liu, and Lizhuang Ma. Shape context based mesh saliency detection and its applications: A survey. *Computers & Graphics*, 57:12–30, 2016.

[23] Chang Ha Lee, Amitabh Varshney, and David W Jacobs. Mesh saliency. *ACM Transactions on Graphics (TOG)*, 24(3):659–666, 2005.

[24] Ran Song, Yonghuai Liu, Ralph R Martin, and Paul L Rosin. Mesh saliency via spectral processing. *ACM Transactions on Graphics (TOG)*, 33(1):6:1–6:17, 2014.

[25] Max Limper, Arjan Kuijper, and Dieter W Fellner. Mesh saliency analysis via local curvature entropy. In *Proceedings of the Thirty-Seventh Annual Conference of the European Association for Computer Graphics: Short Papers*, pages 13–16. Eurographics Association, 2016.

[26] Anass Nouri, Christophe Charrier, and Olivier Lézoray. Multi-scale mesh saliency with local adaptive patches for viewpoint selection. *Signal Processing: Image Communication*, 38:151–166, 2015.

[27] Pingping Tao, Junjie Cao, Shuhua Li, Xiuping Liu, and Ligang Liu. Mesh saliency via ranking unsalient patches in a descriptor space. *Computers & Graphics*, 46:264–274, 2015.

[28] Hantao Liu and Ingrid Heynderickx. Visual attention in objective image quality assessment: Based on eye-tracking data. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(7):971–982, 2011.

[29] Mylène CQ Farias and Welington YL Akamine. On performance of image quality metrics enhanced with visual attention computational models. *Electronics Letters*, 48(11):631–633, 2012.

[30] Hantao Liu, Ulrich Engelke, Junle Wang, Patrick Le Callet, and Ingrid Heynderickx. How does image content affect the added value of visual attention in objective image quality assessment? *IEEE Signal Processing Letters*, 20(4):355–358, 2013.

[31] Lin Zhang, Ying Shen, and Hongyu Li. Vsi: A visual saliency-induced index for perceptual image quality assessment. *IEEE Transactions on Image Processing*, 23(10):4270–4281, 2014.

[32] Wei Zhang, Ali Borji, Zhou Wang, Patrick Le Callet, and Hantao Liu. The application of visual saliency models in objective image quality assessment: A statistical evaluation. *IEEE Transactions on Neural Networks and Learning Systems*, 27(6):1266–1278, 2016.

[33] Xiang Feng, Wanggen Wan, Richard Yi Da Xu, Stuart Perry, Song Zhu, and Zexin Liu. A new mesh visual quality metric using saliency weighting-based pooling strategy. *Graphical Models*, 99:1–12, 2018.

[34] Youngmin Kim, Amitabh Varshney, David W Jacobs, and François Guimbretière. Mesh saliency and human eye fixations. *ACM Transactions on Applied Perception (TAP)*, 7(2):12:1–12:13, 2010.

[35] Xiaobai Chen, Abulhair Saparov, Bill Pang, and Thomas Funkhouser. Schelling points on 3D surface meshes. *ACM Transactions on Graphics (TOG)*, 31(4):29:1–29:12, 2012.

[36] Thomas C Schelling. *The strategy of conflict*. Harvard University Press, 1980.

[37] Flora Ponjou Tasse, Jirí Kosinka, and Neil A Dodgson. Quantitative analysis of saliency models. In *Proceedings of SIGGRAPH ASIA 2016 Technical Briefs*, pages 19:1–19:4. ACM, 2016.

[38] Elizabeth Shtrom, George Leifman, and Ayellet Tal. Saliency detection in large point sets. In *Pro-*

ceedings of IEEE International Conference on Computer Vision (ICCV), pages 3591–3598. IEEE, 2013.

[39] Flora Ponjou Tasse, Jiri Kosinka, and Neil Dodgson. Cluster-based point set saliency. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 163–171, 2015.

[40] Guillaume Lavoué. A local roughness measure for 3D meshes and its application to visual masking. *ACM Transactions on Applied perception (TAP)*, 5(4):21:1–21:23, 2009.

[41] David Cohen-Steiner and Jean-Marie Morvan. Restricted Delaunay triangulations and normal cycle. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry*, pages 312–321. ACM, 2003.

[42] Haeyoung Lee, Pierre Alliez, and Mathieu Desbrun. Angle-analyzer: A triangle-quad mesh codec. *Computer Graphics Forum*, 21(3):383–392, 2002.

[43] Zachi Karni and Craig Gotsman. Spectral compression of mesh geometry. In *Proceedings of the Twenty-Seventh Annual Conference on Computer Graphics and Interactive Techniques*, pages 279–286. ACM Press/Addison-Wesley Publishing Co., 2000.

[44] Olga Sorkine, Daniel Cohen-Or, and Sivan Toledo. High-pass quantization for mesh encoding. In *Proceedings of Eurographics Symposium on Geometry Processing*, pages 1–11, 2003.

[45] Massimiliano Corsini, Mohamed-Chaker Larabi, Guillaume Lavoué, Oldřich Petřík, Libor Váša, and Kai Wang. Perceptual metrics for static and dynamic triangle meshes. *Computer Graphics Forum*, 32(1):101–125, 2013.

[46] Guillaume Lavoué and Massimiliano Corsini. A comparison of perceptually-based metrics for objective evaluation of geometry processing. *IEEE Transactions on Multimedia*, 12(7):636–649, 2010.

[47] A. A. M Muzahid, Wanggen Wan, and Xiang Feng. Perceptual quality evaluation of 3D triangle mesh: a technical review. In *Proceedings of 2018 International Conference on Audio, Language and Image Processing (ICALIP)*, pages 266–272. IEEE, July 2018.

[48] Paolo Cignoni, Claudio Montani, and Roberto Scopigno. A comparison of mesh simplification algorithms. *Computers & Graphics*, 22(1):37–54, 1998.

[49] Paolo Cignoni, Claudio Rocchini, and Roberto Scopigno. Metro: Measuring error on simplified surfaces. *Computer Graphics Forum*, 17(2):167–174, 1998.

[50] Peter Lindstrom and Greg Turk. Image-driven simplification. *ACM Transactions on Graphics (ToG)*, 19(3):204–241, 2000.

[51] Bernice E Rogowitz and Holly E Rushmeier. Are image quality metrics adequate to evaluate the quality of geometric objects? In *Human Vision and Electronic Imaging*, pages 340–348, 2001.

[52] Zhou Wang and Alan C Bovik. Modern image quality assessment. *Synthesis Lectures on Image,*

Video, and Multimedia Processing, 2(1):1–156, 2006.

[53] Ioan Cleju and Dietmar Saupe. Evaluation of supra-threshold perceptual metrics for 3D models. In *Proceedings of the Third Symposium on Applied Perception in Graphics and Visualization*, pages 41–44. ACM, 2006.

[54] Guillaume Lavoué, Mohamed Chaker Larabi, and Libor Váša. On the efficiency of image metrics for evaluating the visual quality of 3D models. *IEEE Transactions on Visualization and Computer Graphics*, 22(8):1987–1999, 2016.

[55] Quan Huynh-Thu and Mohammed Ghanbari. The accuracy of psnr in predicting video quality for different video scenes and frame rates. *Telecommunication Systems*, 49(1):35–48, 2012.

[56] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *Proceedings of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1398–1402. Ieee, 2003.

[57] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE Transactions on Image Processing*, 20(8):2378–2386, 2011.

[58] Hamid R Sheikh and Alan C Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, 2006.

[59] Rafat Mantiuk, Kil Joong Kim, Allan G Rempel, and Wolfgang Heidrich. Hdr-vdp-2: a calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Transactions on Graphics (TOG)*, 30(4):40:1–40:13, 2011.

[60] Massimiliano Corsini, Elisa Drelie Gelasca, Touradj Ebrahimi, and Mauro Barni. Watermarked 3-d mesh quality assessment. *IEEE Transactions on Multimedia*, 9(2):247–256, 2007.

[61] Zhe Bian, Shi-Min Hu, and Ralph R Martin. Evaluation for small visual difference between conforming meshes on strain field. *Journal of Computer Science and Technology*, 24(1):65–75, 2009.

[62] Anass Nouri, Christophe Charrier, and Olivier Lézoray. Full-reference saliency-based 3D mesh quality assessment index. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 1007–1011. IEEE, 2016.

[63] Ilyass Abouelaziz, Mohammed El Hassouni, and Hocine Cherifi. Blind 3D mesh visual quality assessment using support vector regression. *Multimedia Tools and Applications*, 77(18):24365–24386, 2018.

[64] Libor Vasa and Vaclav Skala. A perception correlated comparison method for dynamic meshes. *IEEE Transactions on Visualization and Computer Graphics*, 17(2):220–230, 2011.

[65] Fakhri Torkhani, Kai Wang, and Jean-Marc Chassery. Perceptual quality assessment of 3D dynamic meshes: subjective and objective studies. *Signal Processing: Image Communication*, 31:185–204,

2015.

[66] Zeynep Cipiloglu Yildiz and Tolga Capin. A perceptual quality metric for dynamic triangle meshes. *EURASIP Journal on Image and Video Processing*, 2017(1):12:1–12:18, 2017.

[67] Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C Bovik. Gradient magnitude similarity deviation: A highly efficient perceptual image quality index. *IEEE Transactions on Image Processing*, 23(2):684–695, 2014.

[68] Ilyass Abouelaziz, Mohammed El Hassouni, and Hocine Cherifi. A convolutional neural network framework for blind mesh visual quality assessment. In *Proceedings of 2017 IEEE International Conference on Image Processing*, pages 755–759. IEEE, 2017.

[69] Shengfa Wang, Nannan Li, Shuai Li, Zhongxuan Luo, Zhixun Su, and Hong Qin. Multi-scale mesh saliency based on low-rank and sparse analysis in shape feature space. *Computer Aided Geometric Design*, 35:206–214, 2015.

[70] Ran Song, Yonghuai Liu, Ralph R Martin, and Karina Rodriguez Echavarria. Local-to-global mesh saliency. *The Visual Computer*, 34(3):323–336, 2018.

[71] Jinliang Wu, Xiaoyong Shen, Wei Zhu, and Ligang Liu. Mesh saliency with global rarity. *Graphical Models*, 75(5):255–264, 2013.

[72] Xiuping Liu, Pingping Tao, Junjie Cao, He Chen, and Changqing Zou. Mesh saliency detection via double absorbing markov chain in feature space. *The Visual Computer*, 32(9):1121–1132, 2016.

[73] Yitian Zhao, Yonghuai Liu, Yongjun Wang, Baogang Wei, Jian Yang, Yifan Zhao, and Yongtian Wang. Region-based saliency estimation for 3D shape analysis and understanding. *Neurocomputing*, 197:1–13, 2016.

[74] Weiming Wang, Haiyuan Chao, Jing Tong, Zhouwang Yang, Xin Tong, Hang Li, Xiuping Liu, and Ligang Liu. Saliency-preserving slicing optimization for effective 3D printing. *Computer Graphics Forum*, 34(6):148–160, 2015.

[75] Shixiang Jia, Caiming Zhang, Xuemei Li, and Yuanfeng Zhou. Mesh resizing based on hierarchical saliency detection. *Graphical Models*, 76(5):355–362, 2014.

[76] Hung-Kuang Chen and Mu-Wei Li. A novel mesh saliency approximation for polygonal mesh segmentation. *Multimedia Tools and Applications*, 77(13):17223–17246, 2018.

[77] Xue Jiao, Tieru Wu, and Xuzhou Qin. Mesh segmentation by combining mesh saliency with spectral clustering. *Journal of Computational and Applied Mathematics*, 329:134–146, 2018.

[78] George Leifman, Elizabeth Shtrom, and Ayellet Tal. Surface regions of interest for viewpoint selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(12):2544–2556, 2016.

[79] Xi Wang, David Lindlbauer, Christian Lessig, Marianne Maertens, and Marc Alexa. Measuring

the visual salience of 3D printed objects. *IEEE Computer Graphics and Applications*, 36(4):46–55, 2016.

[80] Moritz Kassner, William Patera, and Andreas Bulling. Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pages 1151–1160. ACM, 2014.

[81] Guillaume Lavoué, Frédéric Cordier, Hyewon Seo, and Mohamed-Chaker Larabi. Visual attention for rendered 3D shapes. *Computer Graphics Forum*, 37(2):191–203, 2018.

[82] Ulrich Engelke, Hagen Kaprykowsky, Hans-Jürgen Zepernick, and Patrick Ndjiki-Nya. Visual attention in quality assessment. *IEEE Signal Processing Magazine*, 28(6):50–59, 2011.

[83] Zhou Wang and Alan C Bovik. Reduced-and no-reference image quality assessment. *IEEE Signal Processing Magazine*, 28(6):29–40, 2011.

[84] Video Quality Experts Group. Final report from the video quality experts group on the validation of objective models of video quality assessment, Phase II (FR_TV2). Website, 2003. `https://www.its.bldrdoc.gov/media/4150/vqegii_final_report.doc`.

[85] Ke Gu, Guangtao Zhai, Xiaokang Yang, and Wenjun Zhang. Using free energy principle for blind image quality assessment. *IEEE Transactions on Multimedia*, 17(1):50–63, 2015.

[86] Jinjian Wu, Weisi Lin, Guangming Shi, and Anmin Liu. Perceptual quality metric with internal generative mechanism. *IEEE Transactions on Image Processing*, 22(1):43–54, 2013.

[87] Wujie Zhou, Weiwei Qiu, and Ming-Wei Wu. Utilizing dictionary learning and machine learning for blind quality assessment of 3-d images. *IEEE Transactions on Broadcasting*, 63(2):404–415, 2017.

[88] Qingbo Wu, Hongliang Li, Fanman Meng, King N Ngan, Bing Luo, Chao Huang, and Bing Zeng. Blind image quality assessment based on multichannel feature fusion and label transfer. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(3):425–440, 2016.

[89] Lukáš Krasula, Karel Fliegel, Patrick Le Callet, and Miloš Klíma. On the accuracy of objective image and video quality models: New methodology for performance evaluation. In *Proceedings of Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6. IEEE, 2016.

[90] Robert C Streijl, Stefan Winkler, and David S Hands. Mean opinion score (mos) revisited: methods and applications, limitations and alternatives. *Multimedia Systems*, 22(2):213–227, 2016.

[91] M Cheon and J-S Lee. On ambiguity of objective image quality assessment. *Electronics Letters*, 52(1):34–35, 2015.

[92] Anmin Liu, Weisi Lin, Manoranjan Paul, Chenwei Deng, and Fan Zhang. Just noticeable difference

for images with decomposition model for separating edge and textured regions. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11):1648–1652, 2010.

[93] Manri Cheon and Jong-Seok Lee. Ambiguity-based evaluation of objective quality metrics for image compression. In *Proceedings of Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6. IEEE, 2016.

[94] Gabriel Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *Proceedings of Fifth International Conference on Computer Vision*, pages 902–907. IEEE, 1995.

[95] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[96] Hamid R Sheikh, Muhammad F Sabir, and Alan C Bovik. A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Transactions on Image Processing*, 15(11):3440–3451, 2006.

[97] PG Engeldrum. Psychometric scaling, a toolkit for imaging systems development, 2000.

[98] Guillaume Lavoué. Erratum of the results of mesh visual quality metrics. Website, June 2018. `http://liris.cnrs.fr/glavoue/travaux/Erratum.html`.

[99] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the Twenty-Fourth Annual Conference on Computer Graphics and Interactive Techniques*, pages 209–216. ACM Press/Addison-Wesley Publishing Co., 1997.

[100] Nabil Ouerhani, Roman Von Wartburg, Heinz Hugli, and René Müri. Empirical validation of the saliency-based model of visual attention. *Electronic Letters on Computer Vision and Image Analysis*, 3(1):13–24, 2004.

[101] Congyan Lang, Guangcan Liu, Jian Yu, and Shuicheng Yan. Saliency detection by multitask sparsity pursuit. *IEEE Transactions on Image Processing*, 21(3):1327–1338, 2012.

[102] Patrick Le Callet and Florent Autrusseau. Subjective quality assessment irccyn/ivc database. Website, 2005. http://www.irccyn.ec-nantes.fr/ivcdb/.

[103] Y Horita, Keiji Shibata, and ZM Parvez Saddad. Subjective quality assessment toyama database. Website, 2008. http://mict.eng.u-toyama.ac.jp/mict.

[104] Nikolay Ponomarenko, Vladimir Lukin, Alexander Zelensky, Karen Egiazarian, Marco Carli, and Federica Battisti. Tid2008-a database for evaluation of full-reference visual quality assessment metrics. *Advances of Modern Radioelectronics*, 10(4):30–45, 2009.

[105] Eric Cooper Larson and Damon Michael Chandler. Most apparent distortion: full-reference image quality assessment and the role of strategy. *Journal of Electronic Imaging*, 19(1):1–21, 2010.