Visual Comput (2008) 24: 125–138
DOI 10.1007/s00371-007-0178-8

ORIGINAL ARTICLE

Yongwei Miao
Jieqing Feng
Chunxia Xiao
Qunsheng Peng

High frequency geometric detail manipulation and editing for point-sampled surfaces

Published online: 10 October 2007

Y. Miao (✉) · J. Feng · Q. Peng
State Key Lab. of CAD&CG, Zhejiang University, Hangzhou, P.R. China
ywmiao@zjut.edu.cn

Y. Miao
College of Science, Zhejiang University of Technology, Hangzhou, P.R. China

C. Xiao
Computer School, Wuhan University, Wuhan, P.R. China

Abstract In this paper, based on the new definition of high frequency geometric detail for point-sampled surfaces, a new approach for detail manipulation and a detail-preserving editing framework are proposed. Geometric detail scaling and enhancement can always produce fantastic effects by directly manipulating the geometric details of the underlying geometry. Detail-preserving editing is capable of preserving geometric details during the shape deformation of point-sampled model. For efficient editing, the point set of the model is first clustered by a mean shift scheme, according to its anisotropic geometric features and each cluster is abstracted as a simplification sample point (SSP). Our editing operation is implemented by manipulating the SSP first and then diffusing the deformation to all sample points on the underlying geometry. As a post-processing step, a new up-sampling and relaxation procedure is proposed to refine the deformed model. The effectiveness of the proposed method is demonstrated by several examples.

Keywords Point-sampled surfaces · Shape editing · Geometric detail · Simplification sample point · Deformation field · Mean shift

1 Introduction

Benefitting from the rapid developments of the 3D digital photographic and scanning devices, large-scale point-sampled surfaces are now becoming popular in computer graphics. Developing new algorithms for efficiently editing and deforming the shape of point-sampled surfaces is on demand in digital geometry processing.

For various shape editing operations, such as local editing, free-form deformation, shape interpolation, morphing etc, one requirement is to preserve the meso-geometric details of the model, including bumps, folds and wrinkles, etc. To cope with this requirement, some intrinsic surface representation schemes are proposed in computer graphics, which include multi-resolution modeling [10, 12, 33] and differential domain methods [1, 2, 11, 13, 14, 18, 24, 25, 30–32].

In differential domain methods, the surface details are encoded as local differences or derivatives. The differential representation benefits shape editing and deforming in a natural way, the deformed geometry is reconstructed by solving a linear system which minimizes the shape distortion. It supports both the local shape manipulation within a region of interest and large range free-form deformation of the given model.

However, most of the above detail-preserving editing schemes are applicable to polygonal meshes or triangular meshes only, relying heavily on globally consistent connectivity information between sampled vertices. In contrast, in this paper, our novel detail-preserving editing framework is purely point-based (point positions and associated normals) without reconstructing the triangular meshes during editing procedure, which makes it particularly convenient for the large models obtained by scanning devices.

Based on the high frequency geometric detail definition for point-sampled surfaces, a new tool for detail manipulation and a detail-preserving editing framework are investigated. We propose a novel approach of direct geometric detail scaling and enhancement which provide users with a new means of shape modeling.

For detail-preserving editing, our editing framework consists of the following steps. The input geometry is first clustered by mean shift scheme according to its anisotropic geometric features and each cluster is abstracted as a simplification sample point (SSP). Then the simplification points are edited, but the underlying high frequency geometric details are left unchanged. Our editing operations intrinsically define deformation fields around the simplification points, which are diffused to other sample points on the geometry through the invariant local coordinates at the final step. As a postprocessing step, a new up-sampling and relaxation procedure is proposed to refine the deformed model. Figure 1 gives a high-level overview of the framework for high frequency geometric details preservation.

This paper includes the following contributions:

- Based on the transfer functions defined in the frequency domain, a new definition of high frequency geometric detail for point-sampled surfaces is introduced.
- Owing to our new definition of geometric details, we propose a novel shape editing tool with detail scaling and detail enhancement.
- An efficient framework for detail-preserving editing of point-sampled surfaces is set up, which can achieve naturally appealing results in reasonable computing time.

The paper is organized as follows: The related works about shape editing and deformation for point-based surfaces are briefly reviewed in Sect. 2. In Sect. 3, using the transfer functions, the linear, quadric and cubic synthetical geometric details for point-sampled models are defined. Two manipulation methods for high frequency geometric details – detail scaling and detail enhancement are investigated in Sect. 4. In Sect. 5, we describe how to determine neighboring points on the geometry and abstract the whole geometry as a collection of simplification sample points (SSP). Section 6 presents our surface editing approach in detail. Experimental results for detail-preserving editing are shown in Sect. 7, conclusions and limits are given in the last section.

2 Related works

In recent years, point-sampled geometry has received increased attention in digital geometry processing. Considerable research has been devoted to the efficient modeling, shape editing and deformation of point-sampled geometry [3, 7–9, 15–17, 21–23, 29, 34].

By transferring several 2D image editing techniques to irregular 3D point setting, Zwicker et al. [34] presented a system called *Pointshop 3D* for interactive shape and appearance editing of point-sampled surfaces. The system is supported by a powerful point cloud parameterization and a dynamic re-sampling scheme. In [22], Pauly et al. proposed another point-based free-form shape modeling framework by using the hybrid geometry representation of point set surfaces. Pauly et al. [23] designed a powerful and versatile hierarchical representation for point-based modeling based on the multi-resolution techniques.

For discrete point clouds or meshless model, many physically-based simulation techniques including rigid body simulation, deformable modeling, thin shell, and fluid simulation were presented recently.

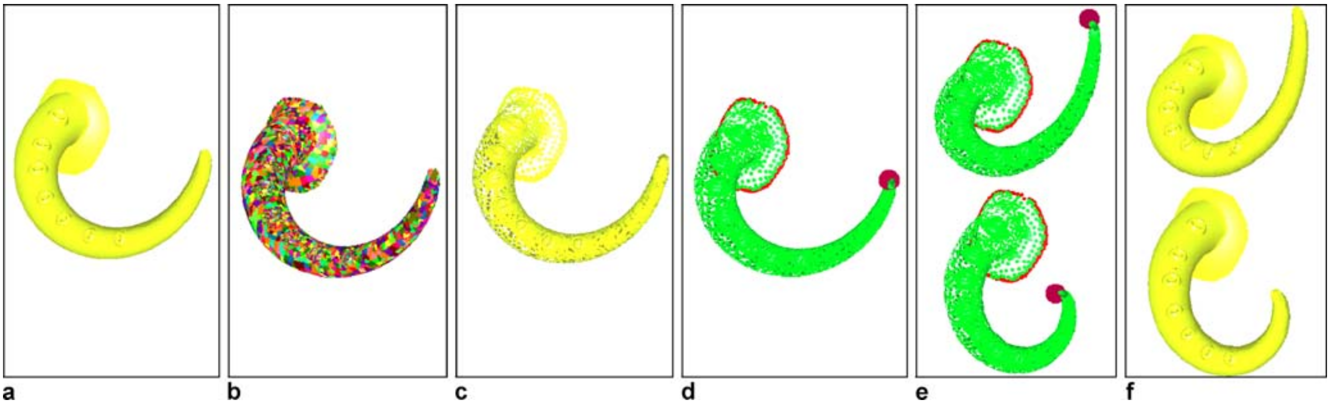

Fig. 1a–f. Flowchart of our high frequency geometric detail-preserving editing. **a** Original model, **b** mean shift clustering for original model, **c** simplification sample points abstraction, **d** preprocessing for editing simplification points, **e** the editing results for simplification points, **f** the final editing results for the point-sampled model

Muller et al. [17] presented a method for modeling and animating elastic, plastic, and melting volumetric objects based on the moving least squares (MLS) approximation of the gradient of the displacement vector field. In [16], Muller et al. proposed a meshless deformation approach, which replaces energies and forces by geometric constraints and distances of current positions to goal positions based on a generalized shape matching scheme. Combined the meshless method with a highly dynamic surface and volume sampling method, Pauly et al. [21] presented an animation framework for elastic and plastic materials that affords complex fracture patterns. Guo and Qin [8] proposed a physically based dynamic local sculpting paradigm for point set surfaces via volumetric implicit functions. Later, they [9] presented a real-time meshless simulation and animation framework for volumetric objects. Wicke et al. [27] simulated and animated discrete thin shells under some related physical principals, such as the Kirchhoff theory of thin plates and the Kirchhoff–Love theory of thin shells. Based on global conformal parameterization and meshless dynamics, Guo et al. [7] presented a physically based simulation approach to simulate thin-shell elastic deformation and fracture effect for point-sampled geometry.

However, all of the above physically based methods focus on the dynamic object behavior for physically realistic results, little attention was paid to the editing of surface details of the underlying geometry.

3 High frequency geometric detail

In spectral analysis [19, 26], high frequency signals contribute to the geometric detail, while low frequency signals account for the overall geometric shape. The high frequency information of the underlying model plays an essential role in its visual appearance.

As a powerful tool for spectral analysis [24, 26], discrete Laplacian transformation is widely used in shape representation, model compression, and digital watermarking, etc. The discrete Laplacian operator can be weighted averages over the neighborhoods:

$$\Delta \boldsymbol{p}_i = \boldsymbol{p}_i - \sum_{j \in N_i} \omega_{ij} \boldsymbol{p}_j = \boldsymbol{K} \boldsymbol{p}_i$$

where N_i denotes the adaptive neighborhoods of regular point $\boldsymbol{p}_i$, for example, the mean shift neighborhood (see Sect. 5.1), the operator $\boldsymbol{K} = \boldsymbol{I} - \boldsymbol{W}$, the weights ω_{ij} are positive numbers that add up to one, they can be chosen in many different ways taking into consideration the spatial distribution of neighborhood points. One particularly simple choice is to set ω_{ij} equal to the inverse of the valence of sample point $\boldsymbol{p}_i$, that is $\frac{1}{d_i}$. Another choice for the weight functions can be some power of the distance of neighbors $\omega_{ij} = \frac{\|\boldsymbol{p}_i - \boldsymbol{p}_j\|^{\mu}}{\sum_j \|\boldsymbol{p}_i - \boldsymbol{p}_j\|^{\mu}}$, for example, $\mu = -1$ always produces good results for our experiments.

Similar to the Fourier transformation on 2D image processing, the discrete Laplacian operator on irregular 3D discrete geometry builds an intrinsic bridge between the space domain and the frequency domain. As an extension of the Fourier analysis, we also define some special transfer functions for the Laplacian operator, focusing on some important geometric details of point-sampled surfaces.

By analyzing the frequency spectrum of the discrete Laplacian matrix $\boldsymbol{K} = \boldsymbol{I} - \boldsymbol{W}$, we get the eigenvalues

$$\lambda_1 \le \lambda_2 \le \lambda_3 \le \ldots \le \lambda_n$$

and corresponding eigenvectors

$$E = \{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3, \ldots, \boldsymbol{e}_n\}$$

which compose a set of orthogonal basis functions. They represent natural vibration modes of the surface. The corresponding eigenvalues are regarded as the associated natural frequencies. Low frequency information (for example, the first eigenvectors), represents the overall geometric shape, while high frequencies ones (for example, the last eigenvectors) accounts for mesoscopic geometric details.

Discrete geometric signal can be decomposed under the basis functions E:

$$\boldsymbol{p} = c_1 \boldsymbol{e}_1 + c_2 \boldsymbol{e}_2 + c_3 \boldsymbol{e}_3 + \ldots + c_n \boldsymbol{e}_n$$

and the Laplacian operator can be applied to the following:

$$\boldsymbol{K}\boldsymbol{p} = c_1 \lambda_1 \boldsymbol{e}_1 + c_2 \lambda_2 \boldsymbol{e}_2 + c_3 \lambda_3 \boldsymbol{e}_3 + \cdots + c_n \lambda_n \boldsymbol{e}_n .$$

So, under the transfer function defined in the frequency domain, the model geometry can be represented as the following:

$$f(\boldsymbol{K})\boldsymbol{p} = c_1 f(\lambda_1)\boldsymbol{e}_1 + c_2 f(\lambda_2)\boldsymbol{e}_2 + c_3 f(\lambda_3)\boldsymbol{e}_3 + \cdots + c_n f(\lambda_n)\boldsymbol{e}_n .$$

We can then define geometric details for point-sampled surfaces with some special transfer functions $f(\boldsymbol{K})$, which exaggerate important geometric details by enhancing high frequency components.

The first simple transfer function is linear function $f(\boldsymbol{K}) = \frac{1}{\lambda}\boldsymbol{K}$, which leads to simple linear Laplacian geometric details,

$$\xi(\boldsymbol{p}_i) = \frac{1}{\lambda} \Delta \boldsymbol{p}_i$$

i.e.

$$\frac{1}{\lambda}\Big(\boldsymbol{p}_i - \sum_{j \in N_i} \omega_{ij} \boldsymbol{p}_j\Big).$$

The linear transfer function contributing to linear geometric details can amplify high frequency components while restrain the low frequency components.

The second transfer function is quadric, for example

$$f(\boldsymbol{K}) = \left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\boldsymbol{K} - \frac{1}{\lambda\mu}\boldsymbol{K}^2, \quad \lambda, \mu > 0$$

which leads to synthetical quadric geometric details,

$$\eta(\boldsymbol{p}_i) = \left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Delta\boldsymbol{p}_i - \frac{1}{\lambda\mu}\Delta^2\boldsymbol{p}_i$$

i.e.,

$$\left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Big(\boldsymbol{p}_i - \sum_{j\in N_i}\omega_{ij}\boldsymbol{p}_j\Big) \\ - \frac{1}{\lambda\mu}\Big(\boldsymbol{p}_i - 2\sum_{j\in N_i}\omega_{ij}\boldsymbol{p}_j + \sum_{j\in N_i}\sum_{k\in N_j}\omega_{ij}\omega_{jk}\boldsymbol{p}_k\Big).$$

And the third is cubic, for example

$$f(\boldsymbol{K}) = \left(\frac{1}{\lambda^2} + \frac{1}{\lambda\mu} + \frac{1}{\mu^2}\right)\boldsymbol{K}^2 - \frac{1}{\lambda\mu}\left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\boldsymbol{K}^3, \\ \lambda, \mu > 0$$

which leads to synthetical cubic geometric details,

$$\zeta(\boldsymbol{p}_i) = \left(\frac{1}{\lambda^2} + \frac{1}{\lambda\mu} + \frac{1}{\mu^2}\right)\Delta^2\boldsymbol{p}_i - \frac{1}{\lambda\mu}\left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Delta^3\boldsymbol{p}_i$$

i.e.,

$$\left(\frac{1}{\lambda^2} + \frac{1}{\lambda\mu} + \frac{1}{\mu^2}\right)\Big(\boldsymbol{p}_i - 2\sum_{j\in N_i}\omega_{ij}\boldsymbol{p}_j + \sum_{j\in N_i}\sum_{k\in N_j}\omega_{ij}\omega_{jk}\boldsymbol{p}_k\Big) \\ - \frac{1}{\lambda\mu}\left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Big(\boldsymbol{p}_i - 3\sum_{j\in N_i}\omega_{ij}\boldsymbol{p}_j + 3\sum_{j\in N_i}\sum_{k\in N_j}\omega_{ij}\omega_{jk}\boldsymbol{p}_k \\ - \sum_{j\in N_i}\sum_{k\in N_j}\sum_{l\in N_k}\omega_{ij}\omega_{jk}\omega_{kl}\boldsymbol{p}_l\Big)$$

where λ means the pass-band frequency, and μ denotes the highest frequency. The high frequency details in the region of $[\lambda, \mu]$ can be enhanced (Fig. 2), yielding some fancy effects.

The color maps for high frequency geometric details are shown in Fig. 3, where different color indicates different oriented magnitude *ori_mag* of geometric details, i.e., it takes *ori_mag* $= \|\boldsymbol{v}\|$ if the high frequency detail vector $\boldsymbol{v}$ satisfy $\boldsymbol{v}\cdot\boldsymbol{n} > 0$, otherwise *ori_mag* $= -\|\boldsymbol{v}\|$. In order to reflect the high frequency details clearly, we first smooth the given model to obtain the low frequency base surface, then show the color maps of geometric details on the base surface. Our experimental results indicate that

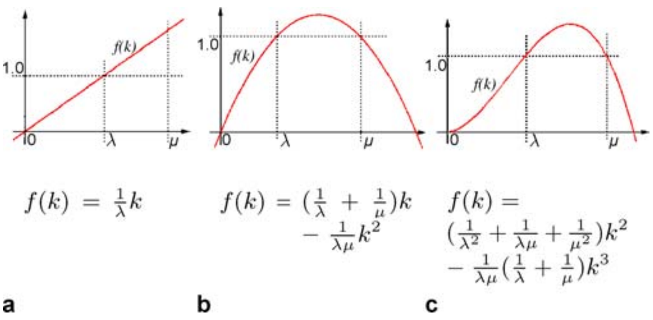

Fig. 2a–c. Different transfer functions for high frequency pass, where λ means the pass-band frequency, and μ denotes the highest frequency. **a** Linear transfer function $f(k) = \frac{1}{\lambda}k$, **b** quadric transfer function $f(k) = (\frac{1}{\lambda} + \frac{1}{\mu})k - \frac{1}{\lambda\mu}k^2$, **c** cubic transfer function $f(k) = (\frac{1}{\lambda^2} + \frac{1}{\lambda\mu} + \frac{1}{\mu^2})k^2 - \frac{1}{\lambda\mu}(\frac{1}{\lambda} + \frac{1}{\mu})k^3$

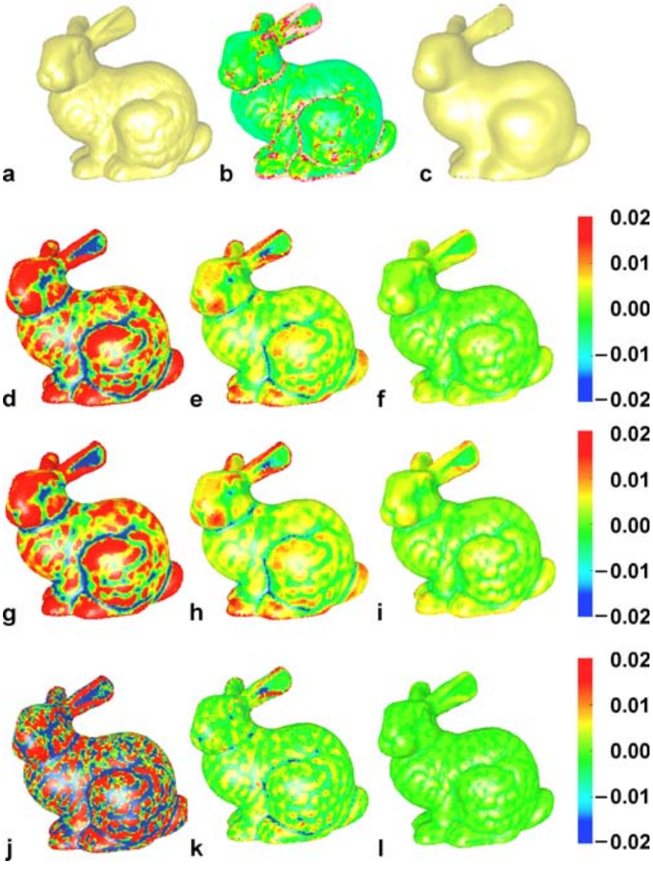

Fig. 3a–l. High frequency geometric details for Stanford bunny model. **a** Original model, **b** color map for curvature of bunny model, **c** the base surface of model, **d–f** color map for linear Laplacian geometric details, corresponding to $\lambda = 0.1$, $\lambda = 0.3$, $\lambda = 0.8$, respectively, **g–i** color map for quadric synthetical geometric details, corresponding to $\lambda = 0.1$, $\mu = 1.8$, $\lambda = 0.3$, $\mu = 1.8$ and $\lambda = 0.8$, $\mu = 1.6$, respectively, **j–l** color map for cubic synthetical geometric details, corresponding to $\lambda = 0.1$, $\mu = 1.8$, $\lambda = 0.3$, $\mu = 1.8$ and $\lambda = 0.8$, $\mu = 1.6$, respectively

quadric and cubic synthetical geometric details can reflect the subtle detail variations more faithfully than linear geometric details. On the other hand, the wide frequency band $[\lambda, \mu]$ can cover the variations of more geometric details. So, for highly detailed geometric models, editing by preserving quadric and cubic synthetical geometric details is a good choice.

4 High frequency geometric detail manipulation

Here, geometric detail manipulation includes two aspects. One is high frequency geometric detail scaling, which can shrink or expand the important surface details for point-sampled geometry. The other is high frequency geometric detail enhancement, which can enhance the model geometric details and is similar to image enhancement in digital image processing. Geometric detail manipulation can always produce fantastic effect in the industry of digital entertainment, while providing us with a new tool of shape modeling.

4.1 Geometric detail scaling

Geometric detail scaling is the process of scaling the high frequency details directly for point-sampled surfaces. For a given point-sampled model, the intrinsic geometric details can be defined as the linear Laplacian geometric detail,

$$\xi(\boldsymbol{p}_i) = \frac{1}{\lambda}\Delta \boldsymbol{p}_i$$

or synthetical quadric and cubic geometric detail, such as

$$\eta(\boldsymbol{p}_i) = \left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Delta \boldsymbol{p}_i - \frac{1}{\lambda\mu}\Delta^2 \boldsymbol{p}_i,$$

$$\zeta(\boldsymbol{p}_i) = \left(\frac{1}{\lambda^2} + \frac{1}{\lambda\mu} + \frac{1}{\mu^2}\right)\Delta^2 \boldsymbol{p}_i - \frac{1}{\lambda\mu}\left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Delta^3 \boldsymbol{p}_i.$$

For the sake of simplicity, we denote them as δ_i uniformly.

For the sample points located in the region of interest (ROI), we set the scaled geometric details as $\delta_i' = s\delta(\boldsymbol{p}_i)$, where s is the scale factor. The resultant surface can then accommodate these scaled details. More precisely, first the system determines the ROI of model surrounded by the user specified anchor points, then applies scaling operation to the sample points in the ROI domain. New positions of the sample points in ROI $\{\boldsymbol{p}_i'\}$, $i \in \{1, 2, \ldots, K\}$ can be obtained by solving the following quadratic minimization problem:

$$\min_{\boldsymbol{p}'}\left(\alpha\sum_{i=1}^{K}\|\delta(\boldsymbol{p}_i') - \delta_i'\|^2 + \sum_{i=1}^{m}\|\boldsymbol{p}_i' - \boldsymbol{p}_i^0\|^2\right).$$

The first term implies the varied geometric details of sample points, the second one implies the position constraints for sample points. The parameter α balances between two kinds of constraints. In our experiments, we set $\alpha = 1.0$. The above quadratic energy minimization can be converted into the following linear system $\boldsymbol{Ax} = \boldsymbol{b}$, that is,

$$\alpha\delta(\boldsymbol{p}_i') = \alpha\delta_i', \quad i \in \{1, 2, \ldots, K\}$$

$$\boldsymbol{p}_i' = \boldsymbol{p}_i^0, \quad i \in \{1, 2, \ldots, m\}$$

which can be solved by applying the conjugate gradient method to the associated normal equations $(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})\boldsymbol{x} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{b}$.

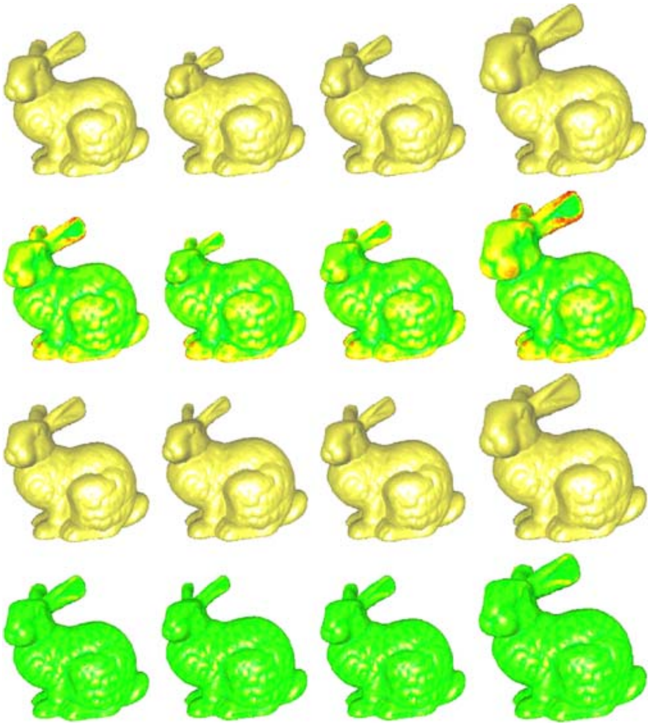

Fig. 4. The detail scaling results for bunny model via the ROI of bunny head. *First row*: original model and quadric synthetical geometric detail scaling ($\lambda = 0.8$, $\mu = 1.6$), the scale factor s is 0.6, 0.8 and 1.2, respectively; *Second row*: color map for quadric synthetical geometric detail scaling; *Third row*: original model and cubic synthetical geometric detail scaling ($\lambda = 0.8$, $\mu = 1.6$), the scale factor s is 0.6, 0.8 and 1.2, respectively. *Fourth row*: color map for cubic synthetical geometric detail scaling

Figure 4 shows the experiment results of scaling the geometric details of Stanford bunny model. The scale factor $s < 1$ means to shrink the geometric details, while $s > 1$ means to expand the geometric details, and $s = 1$ corresponds to the original model. The color maps of geometric details show the detail variations for scaling operation.

4.2 Geometric detail enhancement

Similar to image enhancement in digital image processing, geometric detail enhancement is the process of enhancing the visual important model details. Here, model detail is defined as the detail difference between the original surface and its smoothed base surface. Specifically, for the point-sampled surface S, we generate its smoothed base surface $\tilde{S}$ by balanced curvature flow filtering (see Xiao et al. [28]), then encode the model detail σ_i at sample point $\boldsymbol{p}_i$ based on the high frequency geometric details. Let δ_i and $\tilde{\delta}_i$ be the high frequency geometric detail of S and $\tilde{S}$ at sample point $\boldsymbol{p}_i$ respectively. We define the model detail $\sigma_i = \delta_i - \tilde{\delta}_i$ and enhance the detail as $\delta_i' = \tilde{\delta}_i + s\sigma_i$, in which s is the enhance factor specified by the user.

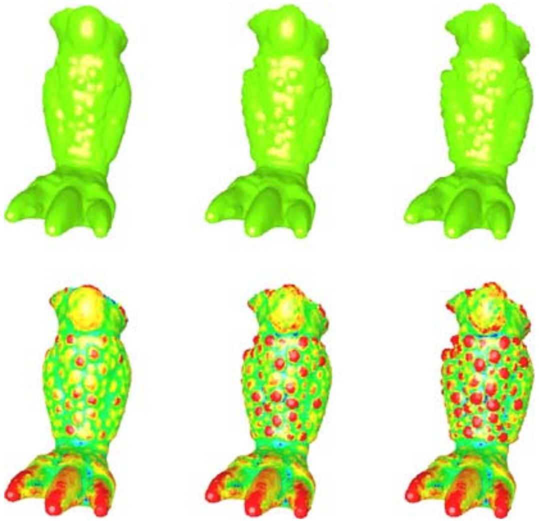

Fig. 5. Linear Laplacian geometric detail enhancement results for the armadillo leg. *First row*: the linear geometric detail enhancement results ($\lambda = 0.2$), the enhancement factor s is 1.0, 2.0 and 3.0, respectively. *Second row*: color map for linear geometric detail enhancement

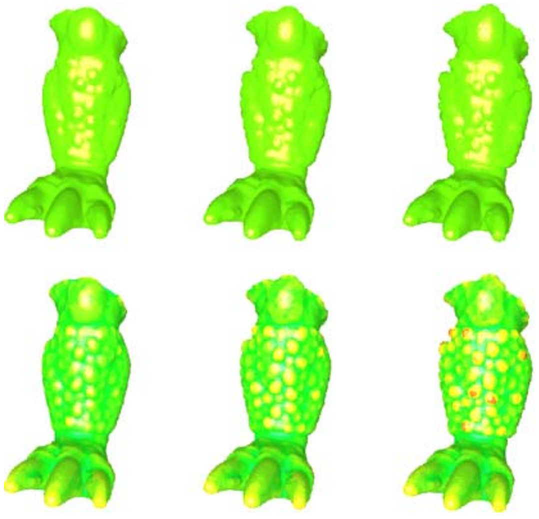

Fig. 6. Quadric synthetical geometric detail enhancement results for the armadillo leg. *First row*: the quadric synthetical detail enhancement results ($\lambda = 0.8$, $\mu = 1.6$), the enhancement factor s is 1.0, 2.0 and 3.0, respectively. *Second row*: color map for quadric synthetical geometric detail enhancement

Similar to geometric detail scaling, the reconstruction of the new surface can be converted to a corresponding quadratic minimization problem and solved by applying the conjugate gradient scheme to the associated normal equations.

Figures 5 and 6 show the experiment results for enhancing the geometric details for the Stanford armadillo leg. The enhancement factor s reflects the degree of the enhancement operation, $s > 1$ means to enhance the geometric details, and $s = 1$ means the original model. The color maps of geometric details show the detail variations by enhancement operation.

5 Generate simplification point set by anisotropic geometric property

5.1 Determine mean shift neighbors

Due to the complexity of point-sampled surfaces, the intrinsic geometric property is always anisotropic, and the neighboring points which share similar geometric attributes with the sample point cannot be determined by isotropic scheme, such as the uniform k-nearest neighbors [20, 34].

The mean shift approach [4–6] is a powerful nonparametric clustering technique for scattered data based on the analysis of multi-modal feature space. Mean shift scheme is essentially an iterative procedure which moves sample point along the direction of the maximum increase in the density gradient. Specially, for the given sampled-point geometry, the sample points equipped with normals and mean curvatures are considered as a scattered data set in seven-dimensional space. We apply the mean shift to the 7D point set to determine the anisotropic neighbor as follows:

1. For each sample point $\boldsymbol{p}_i$ on the sampled-point set, a $7D$ spatial and feature vector

   $$\bar{\boldsymbol{p}}_i = (x_i, y_i, z_i, n_{xi}, n_{yi}, n_{zi}, H_i)$$

   is defined, which contains its position coordinate (x_i, y_i, z_i), normal (n_{xi}, n_{yi}, n_{zi}) and mean curvature H_i;
2. Before mean shift iteration, the k-nearest neighboring points of generalized point $\boldsymbol{p}_i$ are found first, namely $N^S(\boldsymbol{p}_i) = \{\boldsymbol{q}_{i1}, \boldsymbol{q}_{i2}, \dots, \boldsymbol{q}_{ik}\}$;
3. For each sample point $\boldsymbol{p}_i$, the mean shift local mode vector $M^*(\boldsymbol{p}_i)$ in the joint spatial-range domain can be

generated by the following iteration procedure:

$$M_v(\boldsymbol{p}_i) := \frac{\sum_{j=1}^{k} \boldsymbol{q}_{ij}\, g\left(\left\|\boldsymbol{p}_i^r - \boldsymbol{q}_{ij}^r\right\|\right)}{\sum_{j=1}^{k} g\left(\left\|\boldsymbol{p}_i^r - \boldsymbol{q}_{ij}^r\right\|\right)} - M(\boldsymbol{p}_i),$$

$$M(\boldsymbol{p}_i) := M(\boldsymbol{p}_i) + M_v(\boldsymbol{p}_i)$$

where $g(\cdot)$ could be either Gaussian kernel or Epanechnikov kernel, $\boldsymbol{p}_i^r = (\boldsymbol{n}_i, H_i)$ is the range part of $\boldsymbol{p}_i$, $M(\boldsymbol{p}_i)$ is called the mean shift point, which could be initialized to coincide with $\boldsymbol{p}_i$, $M_v(\boldsymbol{p}_i)$ is the mean shift vector associated with $M(\boldsymbol{p}_i)$;

4. Finally, the sample points with close mean shift local modes are considered as the mean shift neighboring points.

We apply the mean shift neighbors to subsequent operations for editing and deformation, and also cluster the original sample points into small regions hierarchically based on the mean-shift local modes of sample points.

5.2 Generate simplification sample points

To edit the shape of large-scale point-sampled model efficiently, reducing the complexity of the data sets while maintaining their geometric features is one of the key preprocessing steps. To satisfy this requirement, some representative sample points, called simplification sample points (SSP), are derived from the underlying model.

We employ a mean shift operator to classify neighboring point, a hierarchical clustering scheme to generate the simplification sample points. It recursively splits the point clouds into a set of clusters by binary space partition based on the following criteria:

– the size is larger than the user specified maximum cluster size (typically 50 sample points, depending on model size) or
– the variation for the mean-shift local modes of sample points exceeds a given threshold.

Then, a binary tree can be built, and each leaf node corresponds to a simplification sample point. We apply the covariance analysis to each cluster to estimate normal vector of the node. The centroid point and estimated normal of each cluster are chosen as the two elements of the *Surfel* representation for the simplification sample points (see Fig. 7).

The mean shift clustering is performed in both the spatial domain and the range domain of sampled points, which helps to generate the simplification sample points for reflecting the intrinsic geometric feature (Fig. 7). For example, the simplification points are dense when representing the subtle detail (usually the high curvature zones), and sparse when representing the planar regions.

6 Geometric detail-preserving editing for point-sampled surfaces

The detail-preserving editing for the given model means to modify the overall geometric shape whilst keeping the meso-geometric details unchanged. It can be formulated as a quadratic energy optimization problem. By minimizing the energy, we can preserve intrinsic geometry details

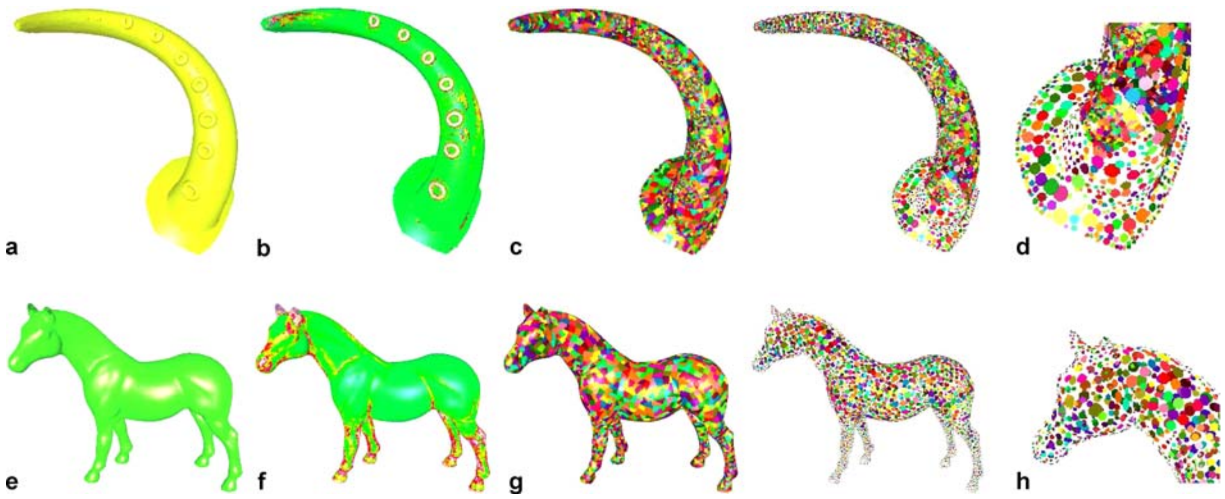

Fig. 7a–h. Generating simplification sample points for point-sampled surfaces. **a** Original tentacle model, **b** the curvature distribution for tentacle, **c** the clustering result by mean shift, **d** the simplification points generated by mean shift clustering and zoomed in for tentacle, **e** original horse model, **f** the curvature distribution for horse, **g** the clustering result by mean shift, **h** the simplification points generated by mean shift clustering and zoomed in for horse. The size of simplification sample point is determined by the number of sample points in the corresponding cluster

between original and deformed geometry in a least squares sense.

However, to fully represent the highly detailed surfaces of a physical 3D model, millions of discrete sample points may be involved. For editing the point-sampled geometry, it is very difficult to manipulate these large numbers of sample points directly by using the current computing resource. For example, the size of the normal equation system for determining the deformed positions of n sample points is $n \times n$ for each coordinate component x, y and z, which is usually a time-consuming procedure for large-scale high detailed point-sampled models. On contrast, the size of the linear system for editing simplification points becomes $K \times K$ for each component. It is always a high-level efficient operation due to relatively few simplification points ($K \ll n$). So, we perform the editing operation on two level representations – a coarse level (simplification points) and a detail level (original model). To speed up the deformation, the original model is first down-sampled and reduced to a simplification point set, the optimized editing operation is then performed on these simplification points. Finally, the deformation result is diffused to rest sample points of the underlying model.

Some editing operations, e.g., stretching or twisting, may cause distortions in the distribution of sample points, leading insufficient local sampling in some local area. To refine the edit results, it is necessary to up-sample the deformed model.

6.1 Editing the simplification point set for geometric detail preservation

Note that the simplification point set of a model is only a partition of the original point-sampled data set at low frequency level.

Given all simplification sample points $\mathbf{SSP} = \{\mathbf{ssp}_i = (x_i, y_i, z_i), i = 1, 2, \ldots, K\}$ of the point-sampled model, the intrinsic geometry can be described by linear Laplacian geometric details,

$$\xi(\boldsymbol{p}_i) = \frac{1}{\lambda}\Delta \boldsymbol{p}_i$$

or synthetical geometric details,

$$\eta(\boldsymbol{p}_i) = \left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Delta \boldsymbol{p}_i - \frac{1}{\lambda\mu}\Delta^2 \boldsymbol{p}_i$$

$$\zeta(\boldsymbol{p}_i) = \left(\frac{1}{\lambda^2} + \frac{1}{\lambda\mu} + \frac{1}{\mu^2}\right)\Delta^2 \boldsymbol{p}_i - \frac{1}{\lambda\mu}\left(\frac{1}{\lambda} + \frac{1}{\mu}\right)\Delta^3 \boldsymbol{p}_i.$$

For convenience, we denote them all as δ_i.

To launch detail-preserving editing over simplification points, user need only specify the deformed absolute positions $\mathbf{ssp}_i^0, i \in \{1, 2, \ldots, m\}$ for a few simplification points called handles, i.e.,

$$\mathbf{ssp}_i' \to \mathbf{ssp}_i^0, i \in \{1, 2, \ldots, m\}$$

and the system will solve the deformed position $\{\mathbf{ssp}_i'\}$, $i \in \{m+1, m+2, \ldots, K\}$ of the remaining points in the region of interest (ROI) and then fit $\mathbf{SSP}' = \{\mathbf{ssp}_i', i = 1, 2, \ldots, K\}$ with given geometric details of original simplification points $\mathbf{SSP}$.

The deformed positions of the simplification points $\{\mathbf{ssp}_i'\}, i \in \{1, 2, \ldots, K\}$ can be obtained by solving the following quadratic minimization problem:

$$\min_{\mathbf{ssp}'}\left(\alpha \sum_{i=1}^{K} \|\delta(\mathbf{ssp}_i') - \delta_i'\|^2 + \sum_{i=1}^{m} \left\|\mathbf{ssp}_i' - \mathbf{ssp}_i^0\right\|^2\right)$$

where δ_i' means the transformed high frequency geometric details in the deformed coordinate frame.

The first term implies preserving high frequency geometric details of deformed simplification points, and the second term implies the position constraints. The parameter α makes balance between the detail-preserving requirement and the position constraints for simplification points. In our experiments, we take $\alpha = 1.0$.

The minimization procedure can be converted into a linear system $\boldsymbol{Ax} = \boldsymbol{b}$ as follows:

$$\alpha\delta(\mathbf{ssp}_i') = \alpha\delta_i', \quad \textit{for } i \in 1, 2, 3, \ldots, K,$$

$$\mathbf{ssp}_i' = \mathbf{ssp}_i^0, \quad \textit{for } i \in 1, 2, 3, \ldots, m.$$

The linear system can be solved by applying the conjugate gradient method to the associated normal equations $(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})\boldsymbol{x} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{b}$.

Propagation of local transforms. Since the high frequency geometric detail definitions are sensitive to linear transformation, especially to rotation of the sample points, it is important to define the deformed position of each simplification point in its local frame. A distance-dependent propagation scheme is then adopted to derive the local transformation $\boldsymbol{T}_i$ for other simplification point $\boldsymbol{p}_i$ besides the handle points, which converts the geometric details δ_i in the global coordinates to new transformed geometric details $\delta_i' = \boldsymbol{T}_i\delta_i$ in the local frame.

After the user defines a region of interest (ROI) on the model and specifies some points as handles, the transforms of the editing handles can be propagated to all simplification points in the region of interest via a deformation strength field

$$f(\boldsymbol{p}) = \beta\left(\frac{d_0(\boldsymbol{p})}{d_0(\boldsymbol{p}) + d_1(\boldsymbol{p})}\right)$$

where $d_0(\boldsymbol{p})$ and $d_1(\boldsymbol{p})$ measure the relative distances of simplification sample point $\boldsymbol{p}$ from the SSP outside the interest region and from the handles, and $\beta(\cdot)$ is a continuous blending function with $\beta(0) = 0$ and $\beta(1) = 1$. So, the closer a simplification point is to the handles, the stronger the deformation will be for that point, and the deformation will be weak if the simplification point approaches to border of the influence region.

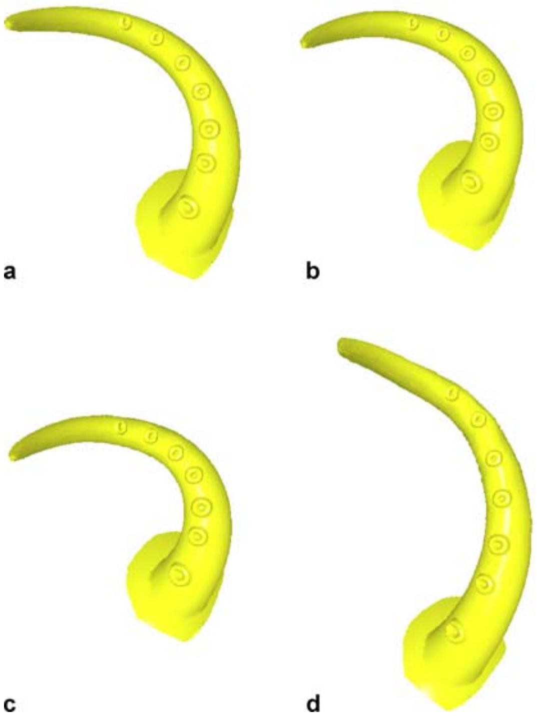

Fig. 8a–d. Global editing tentacle model with our framework. **a** Original tentacle model, **b** the editing result for tentacle by preserving linear Laplacian geometric details, **c** and **d** the editing results for tentacle according to different manipulations for handle points on the tip of tentacle

The local transform $\boldsymbol{T}_i$ can then be represented as a quaternion $\boldsymbol{Q}_i$, and the final transform quaternion at simplification sample point $\boldsymbol{p}_i$ can be interpolated as:

$$\boldsymbol{Q}_i = f(\boldsymbol{p}_i)\boldsymbol{Q}_{\text{handle}} + (1 - f(\boldsymbol{p}_i))\boldsymbol{Q}_I$$

where $\boldsymbol{Q}_I$ denotes a quaternion for the identity transform. This formula simply blends the transform of the handles with the identity using the distance-dependent strength field. The transform is then propagated smoothly to all simplification points in the region of interest.

The propagation of local transforms for tentacle of Octopus are shown in Fig. 8b–d. Note that the orientation of suckers on the tentacle are well-preserved after some rotations of handle point.

6.2 Diffusing the deformation field

The detail-preserving editing operation for simplification points defines deformation fields only at the simplification points. The deformation fields should be diffused to other sample points on the geometry. To satisfy the intrinsic shape features, the local coordinates for the transformed surface pieces should be invariant after deforming operations [27].

Let $\boldsymbol{p}$ be the sample point that needs to be deformed, called an alive point, $\mathbf{SSP} = \{\mathbf{ssp}_i, i = 0, 1, 2, \ldots, m\}$ be some neighboring simplification points in their original state, sorted by the distances to the alive point, and simplification point $\mathbf{ssp}_0$ is the nearest. The relative displacement $\boldsymbol{p} - \mathbf{ssp}_0$ can be encoded in m coordinate systems $\{\boldsymbol{E}_i^1, \boldsymbol{E}_i^2, \boldsymbol{E}_i^3\}$ with respect to the local frames of $\mathbf{ssp}_i$ yielding m relative coordinates $\{(c_i^1, c_i^2, c_i^3), i = 1, 2, \ldots, m\}$, such as,

$$\boldsymbol{p} - \mathbf{ssp}_0 = c_i^1\boldsymbol{E}_i^1 + c_i^2\boldsymbol{E}_i^2 + c_i^3\boldsymbol{E}_i^3$$

and they are stored with the alive point $\boldsymbol{p}$.

After deforming the geometry, the coordinate system $\{\boldsymbol{E}_i^{'1}, \boldsymbol{E}_i^{'2}, \boldsymbol{E}_i^{'3}\}$ for each simplification point is reconstructed according to its deformed position $\{\mathbf{ssp}_i', i = 1, 2, \ldots, m\}$. The local coordinates for the alive point at the local frame of $\mathbf{ssp}_i$ can be transformed back into world coordinates, yielding m positions for the alive point:

$$\boldsymbol{p}_i' = \mathbf{ssp}_0' + c_i^1\boldsymbol{E}_i^{'1} + c_i^2\boldsymbol{E}_i^{'2} + c_i^3\boldsymbol{E}_i^{'3}.$$

And the final deformed position for the alive point $\boldsymbol{p}$ is computed as a weighted sum of the m deformed positions $\boldsymbol{p}_i'$.

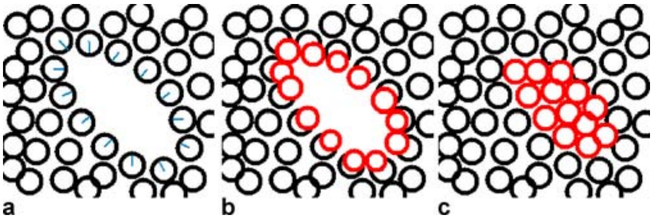

Fig. 9a–c. Up-sampling procedure. **a** Hole detection: the *blue segments* indicate the sparsest direction of boundary points, **b** Up-sampling: the *red circles* are new samples placed along the sparsest direction of boundary points, **c** Relaxation: all *red circles* are pushed by repulsion force and cover the hole

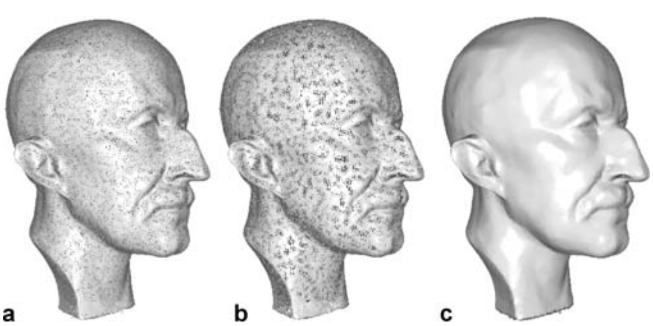

Fig. 10a–c. Up-sampling for point-sampled model. **a** Undersampled point-sampled model, **b** refined model after up-sampling scheme, **c** rendering result of the refined model

Table 1. Statistics and timings for editing different models. The timing is tested on a PC with a Pentium 4 2.0 GHz CPU, 512 M memory

Point-sampled models		Horse	Dog	Tentacle	Dinosaur
#Total sample points		48 484	95 063	44 383	56 194
#Total simplification points		3287	5275	3339	5954
#ROI simplification points		687	2135	3122	5584
Linear Laplacian geometric detail	For editing SSPs	0.1014 s	0.3180 s	1.6070 s	2.5838 s
	For diffusion	0.3223 s	0.7476 s	0.8278 s	1.1815 s
Quadric synthetical geometric detail	For editing SSPs	0.1897 s	0.9469 s	1.9777 s	4.8977 s
	For diffusion	0.3239 s	0.7497 s	0.8338 s	1.5482 s
Cubic synthetical geometric detail	For editing SSPs	1.1658 s	9.4520 s	20.7778 s	/
	For diffusion	0.3247 s	0.7499 s	0.8423 s	/

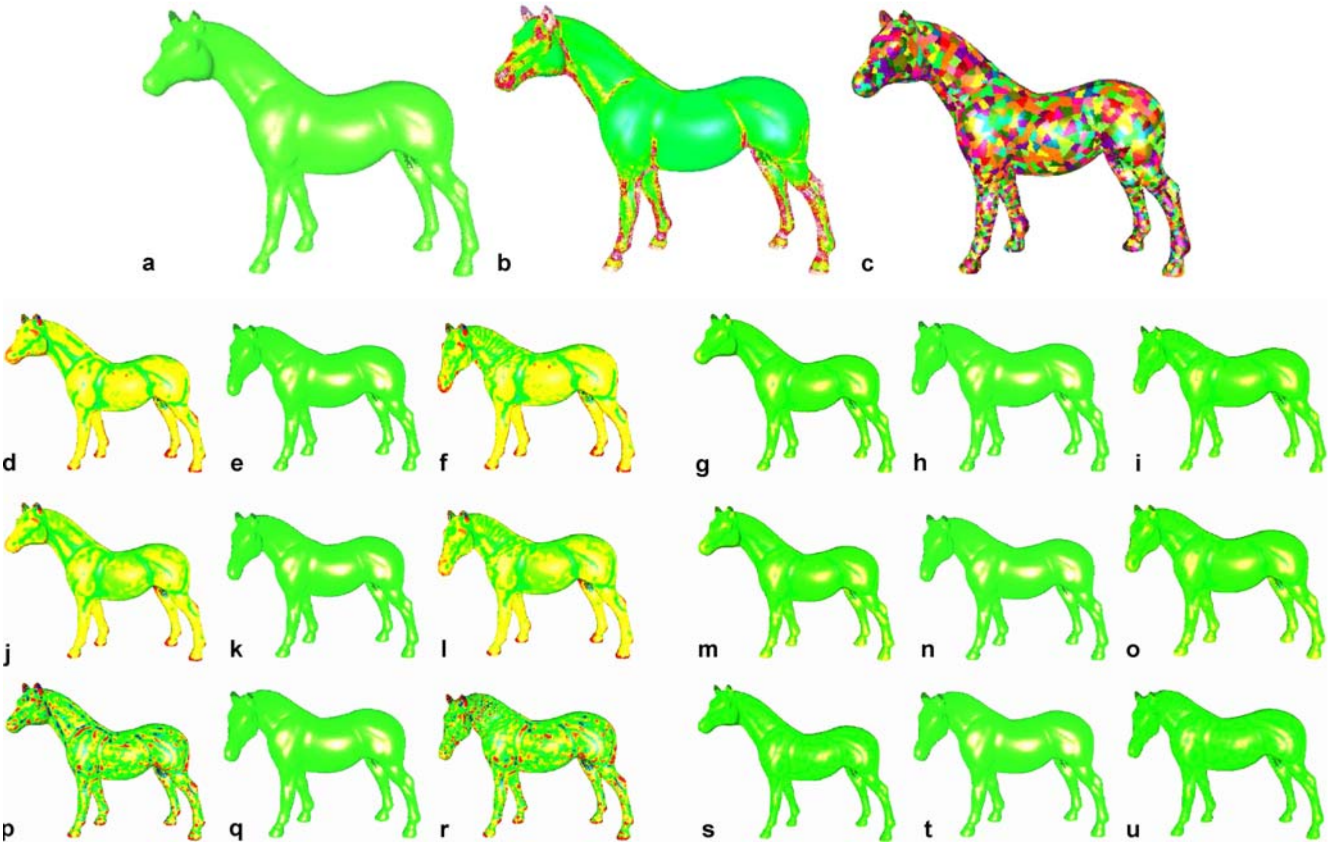

Fig. 11a–u. Local editing horse model with our framework. For every group of the editing results, the *second picture* is the editing result, the *first* and the *third* show the color map of the original model and the edited model, respectively. **a–c** Original horse model, the curvature and segmentation by mean shift, **d–f** the editing result for horse by preserving linear Laplacian geometric details ($\lambda = 0.2$), **g–i** the editing result for horse by preserving linear Laplacian geometric details ($\lambda = 1.0$), **j–l** the editing result for horse by preserving quadric synthetical geometric details ($\lambda = 0.2$, $\mu = 1.8$), **m–o** the editing result for horse by preserving quadric synthetical geometric details ($\lambda = 1.0$, $\mu = 1.6$), **p–r** the editing result for horse by preserving cubic synthetical geometric details ($\lambda = 0.2$, $\mu = 1.8$), **s–u** the editing result for horse by preserving cubic synthetical geometric details ($\lambda = 1.0$, $\mu = 1.6$), respectively

6.3 Postprocessing: Up-sampling for deformed geometry

The up-sampling procedure for deformed sampled-point geometry consists of three steps (see Fig. 9).

The first step is to detect holes on the deformed model. For sample point $\boldsymbol{p}_i$, its mean shift neighboring points $N_i = \{\boldsymbol{q}_{ij}\}$ are determined,and they are projected onto the fitting tangent plane at $\boldsymbol{p}_i$, whose projections are denoted as $\{\boldsymbol{q}^*_{ij}\}$. The corresponding covariance matrix can be constructed for $\{\boldsymbol{q}^*_{ij}\}$, and two eigenvectors $\boldsymbol{e}^1_i$ and $\boldsymbol{e}^2_i$ are computed which correspond to the two nonzero eigenval-

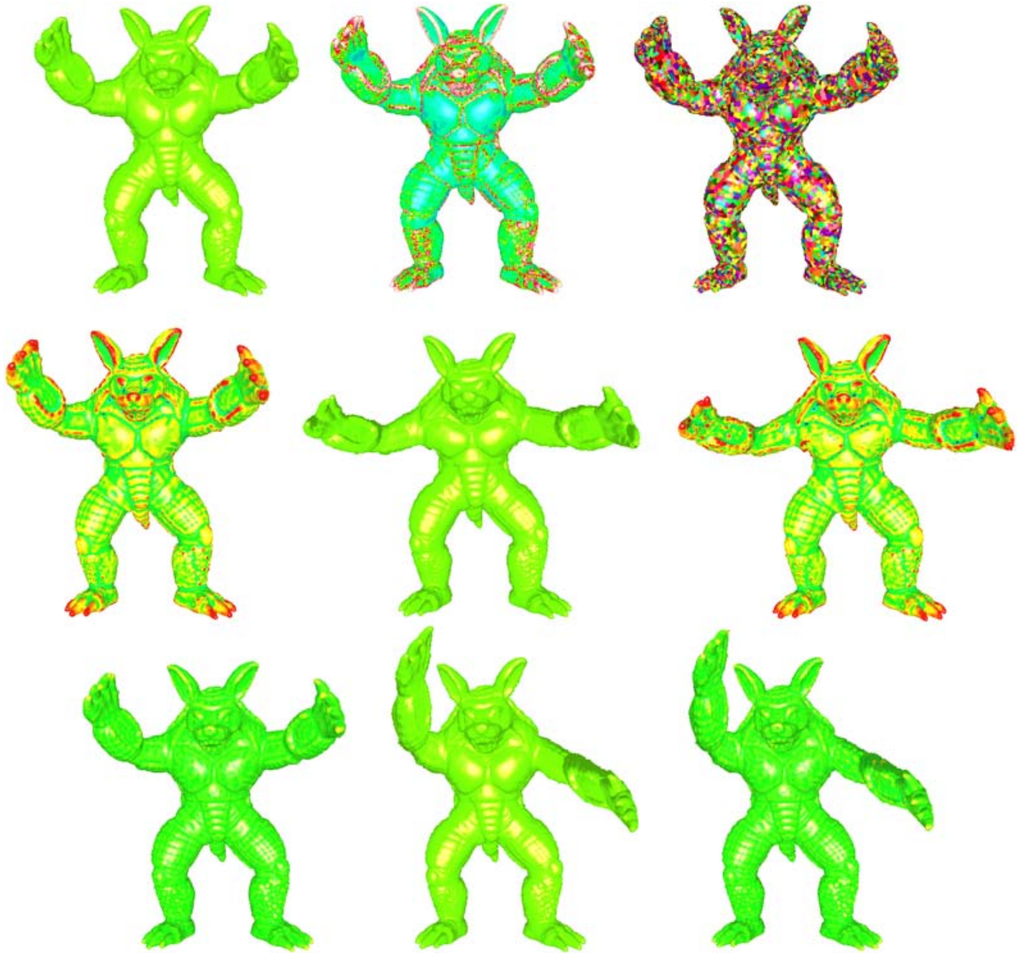

Fig. 12. Large range editing Stanford armadillo model with our framework. *First row*: original model, the curvature and segmentation by mean shift clustering, respectively; *Second row*: the editing result for armadillo by preserving quadric synthetical geometric details ($\lambda = 0.2$, $\mu = 1.8$), and *third row*: the editing result for the armadillo by preserving cubic synthetical geometric details ($\lambda = 0.6$, $\mu = 1.8$) according to different handle manipulations, the two handle points are specified at the end of each hand, respectively

ues. It is asserted that if the sample point $\boldsymbol{p}_i$ locates near a convex boundary of the sparse region, which is recognized as a hole, the direction $\boldsymbol{e}_i^1$(or $-\boldsymbol{e}_i^1$) will point to the hole. Otherwise, if $\boldsymbol{p}_i$ locates near a concave boundary of the hole, the directiony $\boldsymbol{e}_i^2$(or $-\boldsymbol{e}_i^2$) point to the hole. We then calculate the sampling densities along directions $\boldsymbol{e}_i^1$, $-\boldsymbol{e}_i^1$, $\boldsymbol{e}_i^2$, $-\boldsymbol{e}_i^2$. If the density along one direction is approximate zero, it shows that $\boldsymbol{p}_i$ is on the boundary of a hole, and the geometry should be up-sampled along the sparse direction.

The second step is up-sampling along the sparse direction near the hole according to pre-defined density threshold.

Finally, to obtain a more uniform sampling geometry, a relaxation operator is applied to the up-sampled surface, which rearranges the sample points on the surface similar to the relaxation approach proposed by Pauly et al. [20].

The example of Max-Planck model with up-sampling is shown in Fig. 10a–c.

7 Experimental results and discussion

All the algorithms presented in this paper are implemented and tested on a PC with a Pentium 4 2.0 GHz CPU, 512 M memory and Windows XP. Table 1 shows the data statistics and timings for editing different models presented in this paper. Other information of each model includes the number of sample points, the number of simplification points, and the number of simplification points in the region of interest, timings for editing simplification points and for diffusing the deformation via the linear, the quadric and cubic synthetical geometric detail, respectively. As can be seen from the performance data, the deformation for large scale detailed geometry is proved ef-

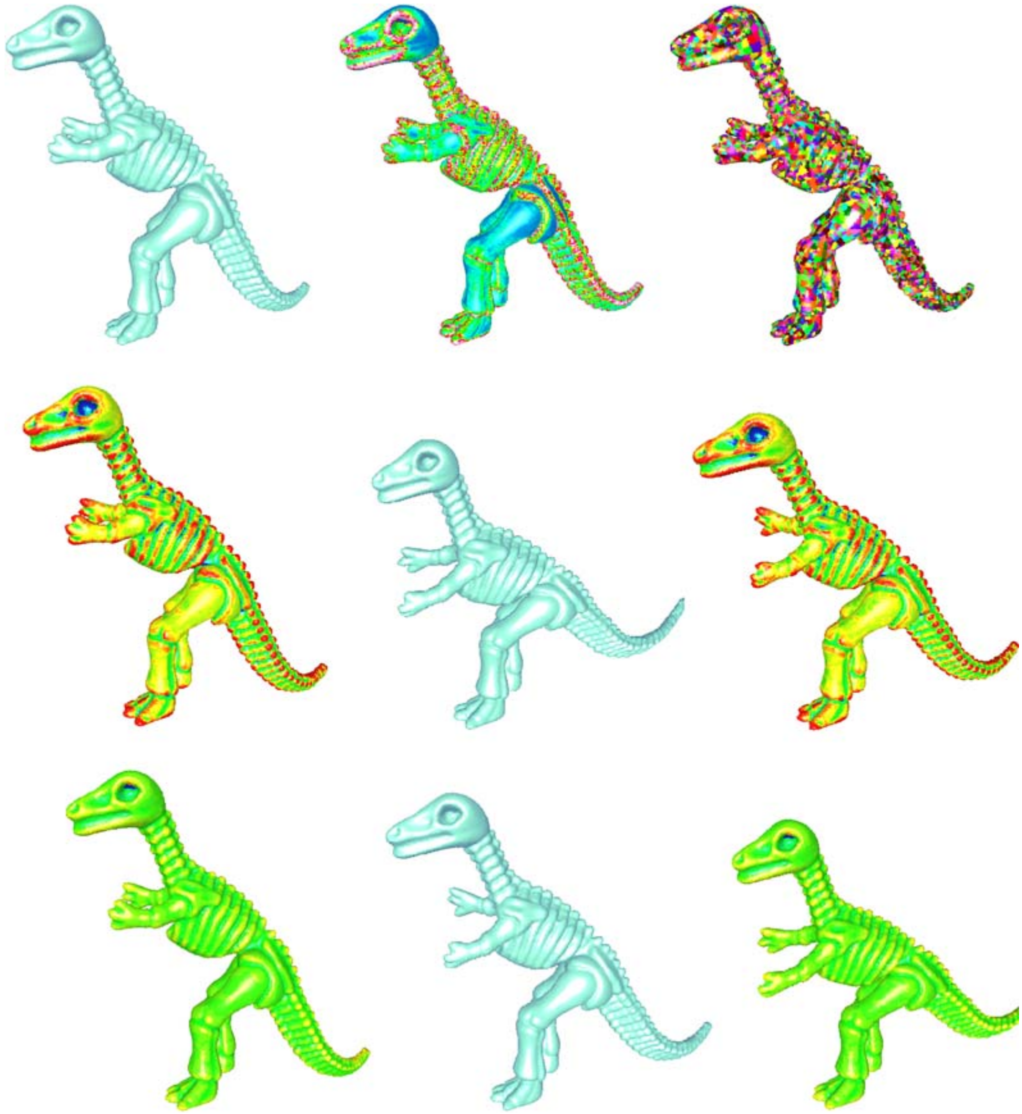

Fig. 13. Global editing dinosaur model with our framework. *First row*: original model, the curvature and segmentation by mean shift clustering, respectively; *Second row*: the global editing result for dinosaur by preserving linear Laplacian geometric details ($\lambda = 0.2$), and *third row*: the global editing result for dinosaur by preserving quadric synthetical geometric details ($\lambda = 0.6$, $\mu = 1.8$) according to different handle manipulations, the four handle points are specified at the mouth, the end of tail, and the front claws, respectively

fectively by our proposed methods, and the user can edit and deform the point-sampled geometry interactively or in reasonable computing time.

Figure 11 shows some local editing results for horse model, in which the horse head is selected as region of interest and is edited with our scheme. To compare the results by editing different geometric details, the handle points remain unchanged during user manipulations. Our experimental results show that preserving quadric and cubic synthetical geometric details during editing can always yield more natural deformation than linear geometric details. Moreover, to reflect the variation of details for different editing, for each group of the results, we illustrate our editing result of model (the second picture), the color map of original model (the first picture) and color map of edited model (the third picture). The comparisons between the color map of original model and those of edited model indicate that the high frequency details are preserved well.

Figure 12 shows some large range editing results for the Stanford armadillo model with our framework, the upper half body of the armadillo is defined as ROI and is edited with our geometric detail preservation scheme. Two handle points are specified at the end of each hand respectively.

Figures 8 and 13 show some global editing results under the user manipulation. For the tentacle model, one handle point on the tip of the tentacle is specified and manipulated, and all the other sample points are deformed systematically by our geometric detail preservation scheme. Four handles are specified for the dinosaur model: one is on the mouth, one is on the end of tail, and the other two is on the front claws.

In view of other detail-preserving editing approaches, we should point out that our high frequency detail-based editing scheme directly includes the Laplacian editing method [13, 25] – linear geometric details preservation. In addition, the existing gradient-domain techniques [11, 30] can hardly be generalized to point-sampled models because of their relying heavily on globally consistent topological information between sample points.

8 Conclusions and future work

In this paper, based on the high-pass transfer functions under the framework of spectral analysis, the linear, quadric and cubic synthetical geometric details for point-sampled geometry are defined. Following our new definition of high frequency geometric details, a new approach of geometric detail manipulation and a detail-preserving editing framework are investigated. The detail manipulation operations can scale and enhance directly the geometric details of the underlying geometry. The detail-preserving editing operation proceeds by adjusting the position of a few simplification sample points interactively, then diffusing the deformation field to other affected sample points on the underlying geometry. A new up-sampling and relaxation procedure is proposed for postprocessing the deformed model to achieve uniform sampling density.

However, in our editing framework, to compensate for the rotation-invariance of geometric details and to preserve the local details naturally, the local rotation transforms of the editing handles should be explicitly propagated to all simplification points. This step may lead to unnaturally and implausible results for large deformations, such as those found with characters performing nonrigid and highly exaggerated movements. To solve this problem, volumetric details for each sample point can be defined and should be preserved during the editing and deformation procedure.

In the future, more interactive editing and deformation operations based on high frequency geometric details should be investigated, for example, detail transfer, deformation transfer, shape interpolation, etc. Another challenging area for future work may be the investigation of other rigid motion invariant intrinsic representation of point-sampled surfaces.

Acknowledgement The authors would like to thank Prof. Mark Pauly and Dr. Richard Keiser for providing the Octopus model. This research work is supported by The National Basic Research Program of China (973 Program) under Grant No. 2002CB312101, National Natural Science Foundation of China (NSFC) under Grant Nos. 60503056, 60333010 and Natural Science Foundation of Zhejiang Province under Grant No. R106449.

References

1. Alexa, M.: Differential coordinates for local mesh morphing and deformation. Visual Comput. **19**(2), 105–114 (2003)
2. Au, O.-C., Tai, C.-L., Liu, L., Fu, H.: Dual Laplacian editing for meshes. IEEE Trans. Vis. Comput. Graph. **12**(3), 386–395 (2006)
3. Bao, Y., Guo, X., Qin, H.: Physically-based morphing of point sampled surfaces. Comput. Anim. Virtual Worlds **16**(3–4), 509–518 (2005)
4. Cheng, Y.: Mean shift, mode seeking, and clustering. IEEE Trans. Pattern Anal. Machine Intell. **17**, 790–799 (1995)
5. Comaniciu, D., Meer, P.: Mean shift analysis and applications. In: Proceedings of IEEE International Conference on Computer Vision, pp. 1197–1203. IEEE Computer Society, Washington (1999)
6. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Machine Intell. **24**(5), 603–619 (2002)
7. Guo, X., Li, X., Bao, Y., Gu, X., Qin, H.: Meshless thinshell simulation based on global conformal parameterization. IEEE Trans. Vis. Comput. Graph. **12**(3), 375–385 (2006)
8. Guo, X., Qin, H.: Dynamic sculpting and deformation of point set surfaces. In: Proceedings of Pacific Graphics, pp. 123–130. IEEE Computer Society, Washington (2003)
9. Guo, X., Qin, H.: Real-time meshless deformation. Comput. Anim. Virtual Worlds **16**(3–4), 189–200 (2005)
10. Guskov, I., Sweldens, W., Schroder, P.: Multiresolution signal processing for meshes. In: Proceedings of ACM SIGGRAPH, pp. 325–334. ACM Press, New York (1999)
11. Huang, J., Shi, X., Liu, X., Zhou, K., Wei, L.-Y., Teng, S.-H., Bao, H., Guo, B., Shum, H.-Y.: Subspace gradient domain mesh deformation. ACM Trans. Graph. **25**(3), 1126–1134 (2006)
12. Kobbelt, L., Campagna, S., Vorsatz, J., Seidel, H.-P.: Interactive multi-resolution modeling on arbitrary meshes. In: Proceedings of ACM SIGGRAPH, pp. 105–114. ACM Press, New York (1998)
13. Lipman, Y., Sorkine, O., Cohen-Or, D., Levin, D., Rossl, C., Seidel, H.-P.: Differential coordinates for interactive mesh editing. In: Proceedings of Shape Modeling International, pp. 181–190. IEEE Computer Society, Washington (2004)
14. Lipman, Y., Sorkine, O., Levin, D., Cohen-Or, D.: Linear rotation-invariant coordinates for meshes. ACM Trans. Graph. **24**(3), 479–487 (2005)
15. Miao, Y., Feng, J., Xiao, C., Li, H., Peng, Q.: Detail-preserving local editing for point-sampled geometry. In: Proceedings of Computer Graphics International Conference, Lecture Notes in Computer Science, vol. 4035, pp. 673–681. Springer, Germany (2006)
16. Muller, M., Heidelberger, B., Teschner, M., Gross, M.: Meshless deformations based on shape matching. ACM Trans. Graph. **24**(3), 471–478 (2005)
17. Muller, M., Keiser, R., Nealen, A., Pauly, M., Gross, M., Alexa, M.: Point-based animation of elastic, plastic, and melting objects. In: Eurographics/ACM SIGGRAPH Symposium on Computer Animation, pp. 141–151. Eurographics Association, Switzerland (2004)
18. Nealen, A., Sorkine, O., Alexa, M., Cohen-Or, D.: A sketch-based interface for detail-preserving mesh editing.

ACM Trans. Graph. **24**(3), 1142–1147 (2005)

19. Pauly, M., Gross, M.: Spectral processing of point-sampled geometry. In: Proceedings of SIGGRAPH, pp. 379–386. ACM Press, New York (2001)
20. Pauly, M., Gross, M., Kobbelt, L.: Efficient simplification of point-sampled surfaces. In: Proceedings of IEEE Visualization, pp. 163–170. IEEE Computer Society, Washington (2002)
21. Pauly, M., Keiser, R., Adams, B., Dutre, P., Gross, M., Guibas, L.: Meshless animation of fracturing solids. ACM Trans. Graph. **24**(3), 957–964 (2005)
22. Pauly, M., Keiser, R., Kobbelt, L., Gross, M.: Shape modeling with point-sampled geometry. ACM Trans. Graph., **22**(3), 641–650 (2003)
23. Pauly, M., Kobbelt, L., Gross, M.: Point-based multiscale surface representation. ACM Trans. Graph. **25**(2), 177–193 (2006)
24. Sorkine, O.: Laplacian mesh processing. State of The Art Report. In: Proceedings of the Eurographics, pp. 53–70. Eurographics Association, Switzerland (2005)
25. Sorkine, O., Lipman, Y., Cohen-Or, D., Alexa, M., Rossl, C., Seidel, H.-P.: Laplacian surface editing. In: Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, pp. 179–188. Eurographics Association, Switzerland (2004)
26. Taubin, G.: A signal processing approach to fair surface design. In: Proceedings of ACM SIGGRAPH, pp. 351–358. ACM Press, New York (1995)
27. Wicke, M., Steinemann, D., Gross, M.: Efficient animation of point-sampled thin shells. Comput. Graph. Forum **24**(3), 667–676 (2005)
28. Xiao, C., Miao, Y., Liu, S., Peng, Q.: A dynamic balanced flow for filtering point-sampled geometry. Visual Comput. **22**(3), 210–219 (2006)
29. Xiao, C., Zheng, W., Miao, Y., Zhao, Y., Peng, Q.: A unified method for appearance and geometry completion of point set surfaces. Visual Comput. **23**(6), 433–443 (2007)
30. Yu, Y., Zhou, K., Xu, D., Shi, X., Bao, H., Guo, B., Shum, H.-Y.: Mesh editing with poisson-based gradient field manipulation. ACM Trans. Graph. **23**(3), 641–648 (2004)
31. Zayer, R., Rossl, C., Karni, Z., Seidel, H.-P.: Harmonic guidance for surface deformation. Comput. Graph. Forum **24**(3), 601–609 (2005)
32. Zhou, K., Huang, J., Snyder, J., Liu, X., Bao, H., Guo, B., Shum, H.-Y.: Large mesh deformation using the volumetric graph Laplacian. ACM Trans. Graph. **24**(3), 496–503 (2005)
33. Zorin, D., Schroder, P., Sweldens, W.: Interactive multiresolution mesh editing. In: Proceedings of ACM SIGGRAPH, pp. 259–268. ACM Press, New York (1997)
34. Zwicker, M., Pauly, M., Knoll, O., Gross, M.: Pointshop 3D: An interactive system for point-based surface editing. ACM Trans. Graph. **21**(3), 322–329 (2002)

YONGWEI MIAO born in 1971, Ph.D., associate professor. His research interests include virtual reality, digital geometry processing and computer-aided geometric design.

JIEQING FENG born in 1970, Ph.D., professor. His research interests include space deformation, computer-aided geometric design, and computer animation.

CHUNXIA XIAO born in 1976, Ph.D. His research interests include virtual reality, digital geometry processing and point-based computer graphics.

QUNSHENG PENG born in 1947, Ph.D., professor. His research interests include virtual reality, realistic image synthesis, infrared image synthesis and computer animation, scientific visualization.