Computing Variation Modes for Point Set Surfaces

Lanfang Miao   Jin Huang[†]   Xinguo Liu[2]   Hujun Bao   Qunsheng Peng   Baining Guo[2]

State Key Lab of CAD&CG, Zhejiang University   [2]Microsoft Research Asia

Abstract

Point sets have become a popular shape representation. In this paper, we present a novel approach to computing variation modes for point set surfaces, and represent the point set surface as a linear combination of the variation modes, called a generative representation for the point set surface. Given a point set, our approach consists of two steps: The first is to produce a set of new samples with increasing smoothness and less detailed features. We use a modified smoothing method based on moving least squares (MLS) surface to produce the samples. The second is to arrange the shape vectors of the new samples together with the original point set into a matrix, and then compute the singular value decomposition of the matrix, producing a set of variation modes (the eigen vectors). Using the variation modes and the generative representation, we can easily synthesize new shapes. Typical applications are low/high/band pass filtering as well as denoising and detail enhancement in multiple scales.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Curve, surface, solid, and object representations

1. Introduction

Acquiring 3D models from real-world objects by 3D digital photography and scanning systems is a relatively inexpensive process for an increasing number of applications. For example, it can help 3D designers to create detailed models at high resolution, which in turn improves visual realism in entertainment and film productions. It is also helpful in rapid prototyping, reverse engineering, and capture of cultural artifacts.

The acquired data is generally a dense set of points, where each point samples a 3D location and possible additional attributes such as normal information and material properties. In the earlier years, the point sets were converted into polygonal meshes for further editing and visualization.However, polygonal meshes are actually not suitable for representing large and complex data sets, since storing and maintaining the connectivity information in meshes is expensive both in terms of memory and computation.

Recently, considerable research efforts have been devoted to point-based modeling and rendering [MA04, KB04],

which regard the point set as a meshless representation for surfaces. As opposed to a polygonal mesh, such a purely point based representation is particularly useful when dealing with large and complex data sets, since there is no connectivity information to worry about during modeling. Point sets have become a popular shape representation as a result. There are many methods for defining and approximating point set surfaces. Moving least squares (MLS) based surface approximation has developed into one of the state-of-art methods [ABCO*01, Lev03, AK04].

Point sets obtained by acquisition devices are usually noisy and contain artifacts due to physical measurement errors. Before they can be used for further modeling purposes, substantial post-processing is usually required. We refer the readers to the paper of Weyrich et al. [WPH*04] for a review on the main sources of noise and artifacts. While noise removal algorithms have been extensively researched for a long time, most of them have been developed for mesh models, and are not applicable on point sets without explicit connectivity information. As point sets become simple and versatile representations for shapes, it is desirable and useful to directly perform noise removal and smoothing operations on points without converting them into meshes in advance.

In this paper, we present a method to compute a set of vari-

[†] This work was done while Jin Huang interned at Microsoft Research Asia
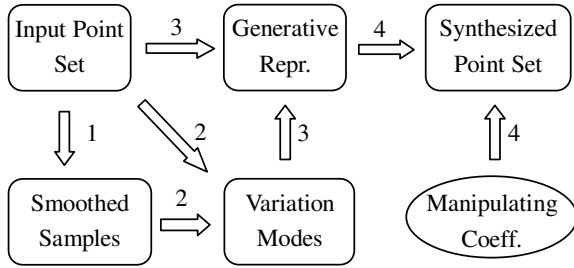
1. Generating smoothed samples. 2. Computing the variation modes. 3. Computing the generative representation. 4. Manipulating the generation coefficients to synthesize new shapes.

Figure 1: *Flowchart of computing variation modes for point set surfaces and synthesizing new shapes.*

ation modes for point set surfaces, and represent the point set surfaces as a linear combination of the variation modes, yielding a generative representation for point set surfaces. Then we use this representation to synthesize new shapes for denoising, smoothing and detail enhancement. The spectral representation presented by Pauly et al. [PG01] can be also regarded as a generative representation using a Fourier basis. Instead of using a Fourier basis, we compute some variation modes specialized for the point models. In this way, we need neither divide the point models into patches nor re-sample the points, which are non-trivial tasks for general complex geometries.

Fig. 1 shows an overview of our approach. We rely on some smoothing operators to decompose the surface details from the input point model. We use MLS surface based smoothing methods [WPH*04] with some modifications for preserving sharp features.

2. Related Works

There are many smoothing algorithms for polygonal meshes. Most of them are based on a discrete approximation of Laplacian smoothing [Tau95] and mean curvature flow [DMSB99]. Later, some improved methods were proposed to avoid shape shrinkage and undesirable deformations [VMM99, LBSP02], and preserve desirable sharp features [FDCO03, JDD03, HP04] on the surfaces.

The MLS surface is a point-based surface definition that is continuous and smooth [ABCO*01, Lev03, AK04]. The MLS surface approximation method has been widely used in many point-based modeling and rendering algorithms [ABCO*03, PKKG03, AA03b, AA03a]. Based on the MLS surface approximation of point sets, Weyrich et al. [WPH*04] implemented a smoothing algorithm by shifting the points towards their projection on the MLS surface by certain distances. We adopt this MLS surface based smoothing method to produce new shape samples for computing a set of variation modes for point set surfaces.

The Fourier transform is a mutual spectral analysis tool and has been successfully applied to implement signal processing algorithms for regular sampled data, such as images. Pauly et al. [PG01] extended the Fourier transform based spectral processing algorithms to point models after pre-charting the point models into patches. Because they use the Fourier basis to represent the patches as height fields, they need to further re-sample each patch into a regular grid of points.

3. Variation Modes

In this section, we describe a method to compute a set of variation modes for a point set surface, and then represent the point model as a linear combination of the variation modes, yielding a generative representation.

Let $\mathcal{P}$ be a point set surface given by a set of 3D points $\mathbf{p}_k \in \mathcal{R}^3$, $k = 1, 2, \ldots, m$. We arrange the points into a vector form $\mathbf{P} = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_m)$, called the *shape vector* of $\mathcal{P}$. The shape vector $\mathbf{P}$ is also a function from a vertex set $\mathcal{V} = \{1, 2, \ldots, m\}$ to $\mathcal{R}^3$, i.e., $\mathbf{P}(k) = \mathbf{p}_k, \forall k \in \mathcal{V}$. Since the shape of a point model consists of many variation modes that vary at different scales, it can be decomposed into a linear combination of some basis shapes. Basis shapes with small-scale variations correspond to high frequency modes, and large-scale variations correspond to low frequency modes.

Representing 3D shapes as a linear combination of some basis shapes is not new, and several methods have been proposed to obtain a set of basis shapes. One category is the mesh based methods. Karni et al. proposed to use the eigen vectors of the Laplacian matrix as the basis shapes for polygonal meshes, and apply them for geometry compression [KG00]. Clarenz et al. introduced an algebraic multigrid based method to compute a multiscale of bases for manifold-2 meshes [CGR*04]. Hauser et al. use the eigen vectors of the stiffness matrix as the basis shapes for tetrahedra meshes, and apply them for interactive deformation [HSO03]. Another category is the example based methods. Cootes et al. collect a training set of labeled examples of objects, and then compute the basis shapes using principal component analysis [CTCG95]. Blanz et al. use a similar method to find the basis shapes for 3D face synthesis [BV99].

Finding suitable basis shapes for 3D shapes is difficult, and it becomes much harder on point set surfaces. The mesh based methods are not applicable to point set surfaces, since connectivity information is not available. So, we resort to the example based method. However, the example based method requires multiple samples of the shape with variations, which is usually also not available. Therefore, we need a method to generate the shape samples.

We will describe a sample generation method in Section 4.

In the following, we suppose that the new shape samples have been generated. Let $\{\mathbf{P}_i = (\mathbf{p}_{i1}, \mathbf{p}_{i2}, \ldots, \mathbf{p}_{im}) \mid 1 \leq i \leq n\}$ be a set of shape vectors generated from the original point set $\mathcal{P}$ using the method in Section 4, where $\mathbf{P}_1 = \mathbf{P}$ is the original shape vector of $\mathcal{P}$ for convenience, and points $\mathbf{p}_{ik}$ and $\mathbf{p}_{jk}$ are assumed to be corresponding points of $\mathbf{P}_i$ and $\mathbf{P}_j$. We find the variation mode by singular value decomposition (SVD). First all points are assembled into a matrix as follows:

$$\mathbb{A} = \begin{pmatrix} \mathbf{p}_{11} & \mathbf{p}_{12} & \cdots & \mathbf{p}_{1m} \\ \mathbf{p}_{21} & \mathbf{p}_{22} & \cdots & \mathbf{p}_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{p}_{n1} & \mathbf{p}_{n2} & \cdots & \mathbf{p}_{nm} \end{pmatrix}$$

Note that the above matrix is $n \times 3m$, since each element of $\mathbb{A}$ is a 3D row vector. Second, we compute the singular value decomposition of $\mathbb{A}$ as follows:

$$\mathbb{A} = \mathbb{U}^t \Sigma \mathbb{V},$$

where $\Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_r)$, $r = min(n, 3m)$, with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r$, and $\mathbb{U}$ and $\mathbb{V}$ are orthogonal matrices of dimensions $n \times n$ and $3m \times 3m$ respectively.

Let $\mathbb{V}_i$, $1 \leq i \leq r$ be the first r row vectors of $\mathbb{V}$, i.e. the right eigen vectors of $\mathbb{A}$. Then each shape vector $\mathbf{P}_i$ in $\mathbb{A}$ can be expressed as a linear combination of $\mathbf{V}_i$. Note that the shape vector $\mathbf{P}$ of the original point model is $\mathbf{P}_1$. Then we have:

$$\mathbf{P} = \alpha_1 \mathbf{V}_1 + \alpha_2 \mathbf{V}_2 + \cdots + \alpha_r \mathbf{V}_r, \quad (1)$$

where $\alpha_i = \mathbf{P} \cdot \mathbf{V}_i$, the dot product of $\mathbf{P}$ and $\mathbf{V}_i$. We call this linear combination a generative representation for point model $\mathcal{P}$, and V_i and α_i are respectively called the variation modes and generation coefficients. For convenience, we adjust the direction of the eigen vectors such that the generation coefficients in Eq. (1) are all non-negative values, i.e., $\alpha_i \geq 0$.

4. Modified MLS Smoother

In the above method, the new shape samples serve as a training set for us to learn the variation modes of a point set surface. So, the variations in the shape samples fully determine their variability of the final generative representation. In order to capture as much as possible surface variations of a point set, we need some *good* shape samples. In this paper, we focus on applications of denoising, smoothing and detail enhancement, then, the new shape samples should have different smoothness and details at different scales.

Therefore, we propose to generate the shape samples by smoothing the original point model to various of degrees. We expect that the shape details at different scales can be separated as much as possible in this way.

We use a smoothing method based on MLS surface approximation [WPH*04]. The MLS surface of point set $\mathcal{P}$

(a)　　　　(b)　　　　(c)
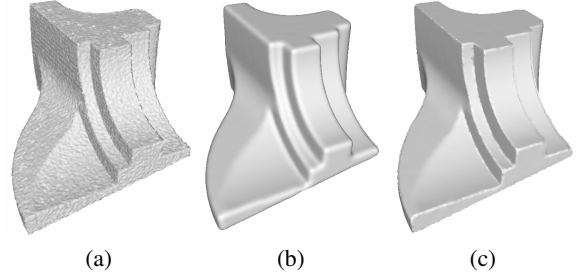
Figure 2: *Comparison between smoothing results. (a) A corrupted noisy Fandisk model. The noises are randomly added in 3D Studio Max with parameters of scale $= 0.03$ and strength $= 0.02$. (b) A smoothing result with the MLS smoother. Note that the sharp edge feature is smoothed out. (c) A smoothing result with our modified MLS smoother. Note that we preserve the sharp edge features well.*

is defined as the stationary set of a projection operator which takes a point $\mathbf{x}$ nearby the point set onto a polynomial that locally approximates the underlying surface in the vicinity of $\mathbf{x}$ [Lev03]. MLS surfaces have been widely used for point based modeling and rendering in the last few years, and there are several slightly different procedures for the MLS projection taking a point $\mathbf{x}$ onto the MLS surfaces [Lev03, ABCO*01, ZPKG02, AK04]. We based our approach on the MLS projection procedure implemented in PointShop3D[ZPKG02] for simplicity and efficiency. Briefly, this MLS projection procedure at point $\mathbf{x}$ takes two steps:

(i) First, a reference plane $\mathcal{H}$ is fitted by minimizing the following weighted least squares:

$$\sum_{i \in \mathcal{V}} (\mathbf{p}_i \cdot \mathbf{n} - D)^2 \theta(\mathbf{x}, \mathbf{p}_i), \quad (2)$$

where $\mathbf{n}$ is the normal vector of the reference plane $\mathcal{H}$, D is the distance of the origin to the reference plane $\mathcal{H}$, and $\theta(\mathbf{x}, \mathbf{p}_i)$ is a positive weighting function.

The reference plane $\mathcal{H}$ provides a local parametrization of the sample points. Let $\mathbf{q}$ be the projection of $\mathbf{x}$ on $\mathcal{H}$, (u_i, v_i) be the local parametrization of $\mathbf{p}_i$'s projection on $\mathcal{H}$, and $f_i = \mathbf{n} \cdot (\mathbf{p}_i - \mathbf{q})$.

(ii) Then a bivariate polynomial $g(u, v)$ in the reference plane $\mathcal{H}$ is fitted by minimizing the following weighted least squares:

$$\sum_{i \in \mathcal{V}} (g(u_i, v_i) - f_i)^2 \theta(\mathbf{x}, \mathbf{p}_i). \quad (3)$$

Finally, the projection of $\mathbf{x}$ on the polynomial is obtained as the MLS projection $\psi(\mathbf{x}) = \mathbf{q} + g(0, 0)\mathbf{n}$.

Based on the MLS projection, Weyrich et al. [WPH*04] implemented a smoothing method, called MLS smoother, by
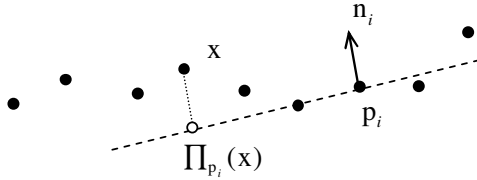
Figure 3: *Linear prediction for point* **x** *by the information of a nearby point* **p**$_i$.

shifting each point $\mathbf{p}_i \in \mathbf{P}$ toward $\psi(\mathbf{p}_i)$ by some distance:

$$\mathbf{p}_i \longleftarrow (1-\alpha)\,\mathbf{p}_i + \alpha\,\psi(\mathbf{p}_i), \qquad (4)$$

where $\alpha \in (0,1]$.

The weighting function in the above the MLS projection is given by a Gaussian $\theta(\mathbf{x},\mathbf{p}_i) = e^{-||\mathbf{x}-\mathbf{p}_i||^2/(2h^2)}$, where h is an user-specified constant. Generally, a larger h will give a smoother surface.

However, the MLS smoother does not preserve sharp features. An example is shown in Fig. 2(b).

Therefore, we introduce a modified MLS smoothing method by incorporating the idea of the bilateral denoising methods [FDCO03, JDD03, JDZ04]. Specifically, we introduce an additional influence weight $\phi(\mathbf{x},\mathbf{p}_i)$ and replace the weighting function $\theta(\mathbf{x},\mathbf{p}_i)$ in Eq. (2) and Eq. (3) by $w(\mathbf{x},\mathbf{p}_i) = \theta(\mathbf{x},\mathbf{p}_i)\phi(\mathbf{x},\mathbf{p}_i)$. The influence weight is defined by

$$\phi(\mathbf{x},\mathbf{p}_i) = \exp\left(-\frac{||\mathbf{x}-\Pi_{\mathbf{p}_i}(\mathbf{x})||^2}{2s^2}\right),$$

where s is the user specified constant parameter, and $\Pi_{\mathbf{p}_i}(\mathbf{x})$ is the linear prediction for **x** given the information at point $\mathbf{p}_i$.

Let $\mathbf{n}_i$ be the normal at point $\mathbf{p}_i$, as shown in Fig. 3, the linear prediction $\Pi_{\mathbf{p}_i}(\mathbf{x})$ for **x** is defined as its projection on the tangent plane of $\mathbf{p}_i$, i.e,

$$\Pi_{\mathbf{p}_i}(\mathbf{x}) = \mathbf{x} + ((\mathbf{p}_i - \mathbf{x}) \cdot \mathbf{n}_i)\mathbf{n}_i.$$

Fig. 2(c) is a smoothing result by our modified MLS smoother, showing that the noises are smoothed out and the sharp edges are well preserved.

The modified MLS smoothing method requires normal information. If it is not available in the input model, we will compute it by the conventional MLS surface approximation method, followed by a procedure that adjusts the normal orientation to be consistent. The normal can be further improved using the method of Jones et al. [JDZ04].

5. Implementation and Results

We implemented the above algorithms of computing variation modes and generative representations for point set sur-

faces, together with a modified MLS smoothing method. In the following we will describe the implementation details and present some smoothing and enhancement results as applications. All the images are rendered in Pointshop3D using its *GPRender* plug-in.

Given a point model, we produce $n = 100$ smoothed samples using the MLS smoother. The process is as follows: We first smooth the original model, producing a smoothed sample. Then we iteratively apply the smoothing method on the output model in the previous step to produce a more smooth sample. During this procedure, we fix the value of h in the weighting function $\theta(\mathbf{x},\mathbf{p}_i)$. We found that $h = 0.02R$ (R is the radius size of the model's bounding sphere) works well in our experiments. The value of s in the influence weighting function is also fixed with $s = 0.005R$. Generating 100 smoothed samples takes about 30 minutes for the Venus model, and up to 90 minutes for the Armadillo model on a PC with *P4 2.7GHZ* CPU and 512*M* RAM.

	Fandisk	Armadillo	Venus head	Max-Planck
$|\mathcal{P}|$	103570	172974	50002	52809

Table 1: *Point number of 4 models in experiments.*

The point number (m) of the models used in experiments is usually much larger than the number (n) of the smoothed samples as shown in Table 1, which indicates $r = min(n,3m) = n$. So, we obtain n eigen vectors after computing the SVD of matrix of the samples. As shown in Table 2 and Fig. 8, the eigen values and generation coefficients quickly decrease to zero, and only a small number of them are significant. It is easy to verify that $\sum_{i=1}^{10}\sigma_i^2 > 0.9999\sum_{i=1}^{r}\sigma_i^2$ and $\sum_{i=1}^{10}\alpha_i^2 > 0.9999\sum_{i=1}^{r}\alpha_i^2$ for all the models in Table 2. Therefore, we can safely truncate the generative representation in Eq. (1) with 10 terms, without introducing any visual artifacts.

We did some smoothing and detail enhancement experiments on the Venus head model by manipulating its generation coefficients (α_i in Eq. (1)). Fig. 4 shows some results of manipulating α_3 and α_5. Bottom left and right are two smoothing results by decreasing α_3 and α_5 respectively, and top left and right are two enhancement results by increasing α_3 and α_5 respectively. The above experiment results show that manipulating α_i with small index i corresponds to manipulating the large scale of surface details, and vice versa. Comparing the results with the original model, we can find that the small-scale details remain when we manipulating the large-scale details, and vice versa, which agree with the orthogonality property of the variation modes.

Fig. 5 shows a comparison of two enhancement results of the Max-Planck head model. In Fig. 5(b), we use our modified MLS smoothing method for computing the variation

	σ_1	σ_2	σ_3	σ_4	σ_5	σ_6	σ_7	σ_8	σ_9	σ_{10}	$\sum_{i=1}^{r}\|\sigma_i\|$	$\sum_{i=1}^{r}\sigma_i^2$
Venus	226107	4013.92	785.609	212.499	88.3272	37.5345	15.5140	6.52406	2.87426	1.50100	231278	5.11410e+10
Max-Planck	424653	1534.97	213.254	54.2995	21.2885	10.9010	6.46497	4.36732	3.09503	2.31973	426520	1.80333e+11
Armadillo	393942	6160.79	1393.93	604.447	343.470	202.847	150.024	107.461	86.1291	68.9264	403661	1.55231e+11

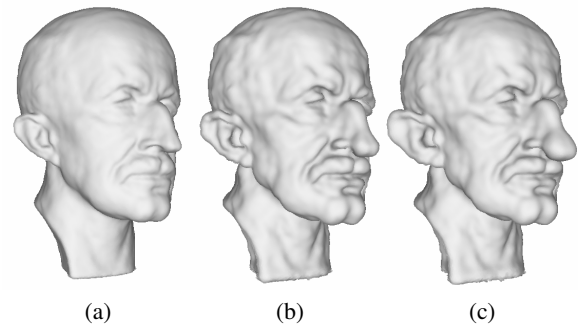|	α_1	α_2	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}	$\sum_{i=1}^{r}\|\alpha_i\|$	$\sum_{i=1}^{r}\alpha_i^2$
Venus	23970.3	863.831	298.706	117.786	50.6907	13.3632	2.61843	0.47769	0.07389	0.00576	25317.8	5.75427e+8
Max-Planck	42596.3	269.644	56.5077	19.2270	7.30376	3.36520	2.00344	1.27550	0.84492	0.63296	42958.7	1.81453e+9
Armadillo	39743.5	1079.38	320.275	184.755	103.707	64.4691	40.7653	34.4288	24.7074	20.5865	41675.1	1.58086e+9

Table 2: *The first 10 eigen values and generation coefficients of 3 point models. The total absolute sums and squared sums of the eigen values and generation coefficients are also shown in the last two columns.*

Figure 4: *Smoothing and enhancement results of Venus head model. Middle: the original model. Bottom: smoothing. Top: enhancement. Left: large scale. Right: small scale.*

(a)	(b)	(c)

Figure 5: *Enhancement results comparison of using two smoothing methods to generate samples for computing the variation modes. (a) the original Max-Planck model. (b) using our modified MLS smoothing method. (c) using the original MLS smoothing method of Weyrich et al. [WPH*04].*

modes, while in Fig. 5(c), we use the original MLS smoothing method of Weyrich et al. [WPH*04]. Though, it is not obvious what the proper way is to compare the enhancement results between two different sets of variation modes, we manipulate the generation coefficients to generate two enhancement results such that overall details on the face are similar. Fig. 5(b) and (c) show that there are noticeable differences on the ears and nose.

Fig. 6 demonstrates an application to smooth and enhance a selected region of a model. The region of interest is selected in Pointshop3D using its brush tool, which gives each point a scalar value. we normalize the values to $[0, 1]$, such that it is 1 in the selected region, and continuously decreases

in the vicinity of the boundary until it finally reaches to 0 in the unselected region. The scalar values serve as blending weights that blend the edit models with the original one. Fig. 6(a) shows the selected region for editing, which is zoomed in Fig. 6(d). Fig. 6(b) and (c) are two enhancement results by respectively increasing α_3 and α_7, and Fig. 6(e) and (f) are two smoothing result by respectively decreasing α_3 and α_7.

Fig. 7(a) shows an exaggeration result using our method. Weyrich et al. showed that shape enhancement can be achieved using Eq. (4) by setting α to negative values. Fig. 7(b) shows an result we produced using the method of Weyrich et al. implemented in Pointshop3D (with parameters k-nearest neighborhood = 300, gaussian smoothing factor = 300, alpha = -1.8 and use only plane fit for smoothing). Our method is advantageous in that it can selectively enhance the desirable scale of details, and/or smooth out undesirable scale of details, to produce pleasing results.
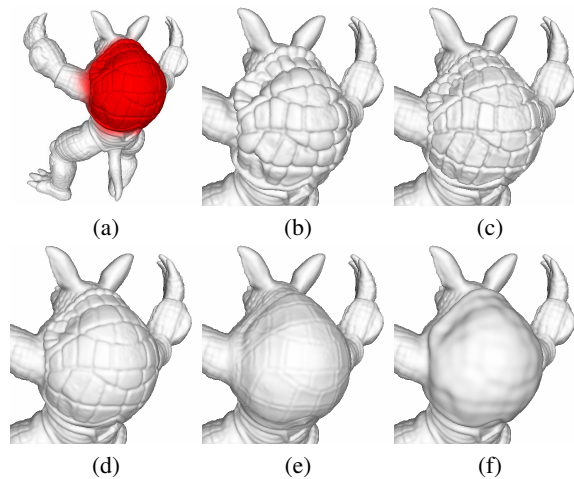
Figure 6: *Smoothing and enhancement results on a selected part of the Armadillo Model. (a) The original model with the selected region painted in red. (d) A zoomed view of the selected region. (b) and (c) two enhancement results. (e) and (f) two smoothing results.*
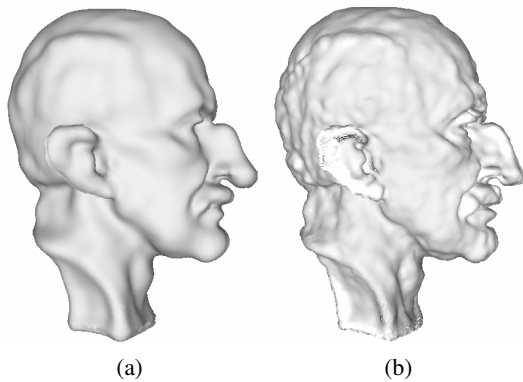

Figure 7: *Comparison between enhancement results of our method (a) and the method in Pointshop3D [WPH*04] (b).*

6. Conclusion and Discussion

Our main contribution in this paper is a novel approach to computing variation modes for point set surfaces, which provides a way to directly process point set surface details in multiple scales.

The results in Fig. 4 and Fig. 6 show that the first a few modes control the large-scale, low-frequency shape variations, while the consequent ones control the small-scale, high-frequency shape variations. We have shown that the variation modes is very useful in smoothing and detail enhancement for point set surface.

In our current implementation, a large number of smoothed samples is generated for computing the variation
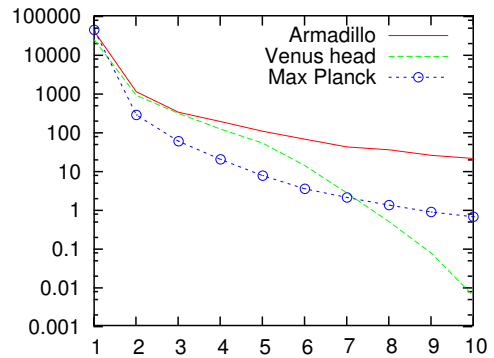
Figure 8: *Plots of the first 10 generation coefficients of 3 point sets.*

modes. It takes some time, as a limitation. A future work would be to analyze how many samples are necessary and adequate for computing a non-trivial set of variation modes.

We use a MLS based smoothing method to filter the point set surfaces for generating new samples. It is worthwhile to try other filters or a combination of several filters.

References

[AA03a] ADAMSON A., ALEXA M.: Approximating and intersecting surfaces from points. In *Eurographics/ACM SIGGRAPH symposium on Geometry processing* (2003), pp. 230–239.

[AA03b] ADAMSON A., ALEXA M.: Ray tracing point set surfaces. In *SMI '03: Proceedings of the Shape Modeling International 2003* (2003), pp. 272–279.

[ABCO*01] ALEXA M., BEHR J., COHEN-OR D., FLEISHMAN S., LEVIN D., SILVA C. T.: Point set surfaces. In *Proceedings of the conference on Visualization* (2001), pp. 21–28.

[ABCO*03] ALEXA M., BEHR J., COHEN-OR D., FLEISHMAN S., LEVIN D., SILVA C. T.: Computing and rendering point set surfaces. *IEEE TVCG 9*, 1 (2003), 3–15.

[AK04]   AMENTA N., KIL Y. J.: Defining point-set surfaces. *ACM Trans. Graph. 23*, 3 (2004), 264–270.

[BV99]   BLANZ V., VETTER T.: A morphable model for the synthesis of 3d faces. In *SIGGRAPH* (1999), pp. 187–194.

[CGR*04]   CLARENZ U., GRIEBEL M., RUMPF M., SCHWEITZER M., TELEA A.: Feature sensitive multiscale editing on surfaces. *The Visual Computer*, 20 (2004), 329–343. Also as Preprint No. 89, SFB 611, Universität Bonn, Germany.

[CTCG95]   COOTES T. F., TAYLOR C. J., COOPER D. H., GRAHAM J.: Active shape models–their training and application. *Comput. Vis. Image Underst. 61*, 1 (1995), 38–59.

[DMSB99]   DESBRUN M., MEYER M., SCHRÖDER P., BARR A. H.: Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH* (1999), pp. 317–324.

[FDCO03]   FLEISHMAN S., DRORI I., COHEN-OR D.: Bilateral mesh denoising. *ACM Trans. Graph. 22*, 3 (2003), 950–953.

[HP04]   HILDEBRANDT K., POLTHIER K.: Anisotropic filtering of non-linear surface features. *Comput. Graph. Forum 23*, 3 (2004), 391–400.

[HSO03]   HAUSER K. K., SHEN C., O'BRIEN J. F.: Interactive deformation using modal analysis with constraints. In *Graphics Interface* (June 2003), pp. 247–256.

[JDD03]   JONES T. R., DURAND F., DESBRUN M.: Non-iterative, feature-preserving mesh smoothing. *ACM Trans. Graph. 22*, 3 (2003), 943–949.

[JDZ04]   JONES T. R., DURAND F., ZWICKER M.: Normal improvement for point rendering. *IEEE Computer Graphics and Applications 24*, 4 (2004), 53–56.

[KB04]   KOBBELT L., BOTSCH M.: A survey of point-based techniques in computer graphics. *Computers & Graphics 28*, 6 (2004), 801–814.

[KG00]   KARNI Z., GOTSMAN C.: Spectral compression of mesh geometry. In *SIGGRAPH* (2000), pp. 279–286.

[LBSP02]   LIU X., BAO H., SHUM H.-Y., PENG Q.: A novel volume constrained smoothing method for meshes. *Graph. Models 64*, 3/4 (2002), 169–182.

[Lev03]   LEVIN D.: Mesh-independent surface interpolation. In *Geometric Modeling for Scientific Visualization* (2003), pp. 37–49.

[MA04]   MARC ALEXA MARKUS GROSS M. P. H. P. M. S. M. Z.: Point-based computer graphics. In *SIGGRAPH 2004 Course Notes* (2004).

[PG01]   PAULY M., GROSS M.: Spectral processing of point-sampled geometry. In *SIGGRAPH* (2001), pp. 379–386.

[PKKG03]   PAULY M., KEISER R., KOBBELT L. P., GROSS M.: Shape modeling with point-sampled geometry. *ACM Trans. Graph. 22*, 3 (2003), 641–650.

[Tau95]   TAUBIN G.: A signal processing approach to fair surface design. In *SIGGRAPH* (1995), pp. 351–358.

[VMM99]   VOLLMER J., MENCL R., MÜLLER H.: Improved laplacian smoothing of noisy surface meshes. *Comput. Graph. Forum 18*, 3 (1999), 131–138.

[WPH*04]   WEYRICH T., PAULY M., HEINZLE S., KEISER R., SCANDELLA S., GROSS M.: Post-processing of scanned 3d surface data. In *Symposium on Point-Based Graphics* (2004), pp. 85–94.

[ZPKG02]   ZWICKER M., PAULY M., KNOLL O., GROSS M.: Pointshop 3d: an interactive system for point-based surface editing. *ACM Trans. Graph. 21*, 3 (2002), 322–329.