Computing point set surfaces with controlled spatial variation of residuals

Yu Liu Xiaoping Qian *

Computational Design and Manufacturing Laboratory

Department of Mechanical, Materials and Aerospace Engineering

Illinois Institute of Technology

Chicago, IL 60616

Abstract

This paper presents an accurate method for computing point-set surfaces from input data that can suppress the noise effect in the resulting point-set surface. This is accomplished by controlling spatial variation of residual errors between the input data and the resulting point-set surface and offsetting any systematic bias. More specifically, this method first reduces random noise of input data based on spatial autocorrelation statistics: the statistics Z via Moran's I. The bandwidth of the surface is adjusted until the surface reaches desired value of the statistics Z corresponding to a given significance level. The method then compensates for potential systematic bias of the resultant surface by offsetting along computed normal vectors. Computational experiments on various point sets demonstrate that the method leads to an accurate surface with controlled spatial variation of residuals and reduced systematic bias.

*All correspondence should be addressed to qian@iit.edu.

1 INTRODUCTION

This paper presents an accurate method for computing point-set surfaces from input data that can suppress the noise effect in the resulting point-set surfaces. Point-set surfaces (PSSs) are continuous surfaces defined directly from point sets. Since its original inception [1, 2, 3], the PSS and its many variants have been widely used in various graphics, visualization, geometric modeling and engineering applications. For example, we have applied the PSS in the context of computer-aided design and manufacturing and developed a new approach termed *direct digital design and manufacturing* [4]. This approach can enable direct digital design and manufacturing from massive scanned data, by passing the usual CAD model reconstruction. This includes the use of PSSs for direct rapid prototyping [5], NC machining [6], and direct Boolean intersection between CAD geometry and acquired geometry[7]. The projection operation in defining PSSs can also be applied for drawing curves onto digital surfaces in points based modeling [8, 9, 10].

However, despite the broad usage of PSSs and its wide variants (for example, ψ-type M-Estimators [11] and the forward search algorithm [12] have been used to improve the shape quality of the resulting PSSs.), thus far there have been limited work on examining the spatial distribution of residual errors of the resulting PSSs.

Our work in this paper assume that the input points are points sampled from object surfaces contaminated with random noise. The premise of this work is the observation that, when the residual errors of input points with respect to the reconstructed PSS approach spatially random, the corresponding PSS approaches the true surface. Our approach is based on Levin's PSS [1]. We obtain the resulting PSS in a two-phase approach. We first vary the bandwidth of the PSS until the statistics Z and Moran's I (measures used for characterizing the spatial randomness of residual errors between the input data and the resultant PSS, which are introduced in Section 3) reach a specified random level. We then compensate for potential systematic bias by offsetting the points in the amount of mean error along the computed

corresponding normal vectors.

Moran's I and corresponding statistics Z have been used in three dimensional coordinate metrology to examine randomness of geometric errors of B-spline surfaces [13]. However, in this paper, we reveal that controlling the randomness alone does not preclude the systematic bias (as shown in Section 4.3). This observation has led to the second phase (the offsetting operation) in our method.

Various computational experiments demonstrate that, through controlled spatial variation of residual errors and the offset of systematic bias of the mean error, accurate PSSs can be obtained for input points sampled from various freeform shapes.

The remainder of the paper is organized as follows. Section 2 reviews related work on PSSs. Section 3 introduces Moran's I and corresponding statistics Z to measure the randomness of spatial patterns. Section 4 presents a method to compute the randomness of residuals in the reconstructed PSS and describes how to offset the potential bias in the surface. Section 5 details the proposed point-set surface reconstruction method. Experimental results are given in Section 6. Finally, conclusions are given in Section 7.

2 RELATED WORK

A point-set surface is a continuous surface defined directly from a point set. Given a point set $P = \{\boldsymbol{p}_i, i = 1, \ldots, n\}$, the original PSS is defined in two steps [1]: an MLS (moving least-squares) projection procedure and a polynomial fitting procedure. First a local plane $H = \{\boldsymbol{x} | \boldsymbol{n} \cdot \boldsymbol{x} - D = 0, x \in R^3\}$ is found by minimizing the weighted sum of squared distances

$$\sum_{i=1}^{n} \left(\boldsymbol{p}_i \cdot \boldsymbol{n} - D\right)^2 \theta\left(\boldsymbol{q}, \boldsymbol{p}_i\right),$$

where $\boldsymbol{q}$ is the projection of $\boldsymbol{x}$ onto H, $\theta\left(\boldsymbol{q}, \boldsymbol{p}_i\right)$ is usually a Gaussian weight function with bandwidth h:

$$\theta\left(\boldsymbol{x}, \boldsymbol{p}_i\right) = e^{-\frac{||\boldsymbol{x}-\boldsymbol{p}_i||^2}{h^2}}. \tag{1}$$

After H is found, the second step finds a local polynomial by taking the H as the reference plane and using a similar weighted least squares method.

Amenta and Kil [3] generalized the MLS projection procedure via the concept of extremal surfaces. The resultant surface of the MLS projection is represented by an implicit function which is the product of a vector field $\boldsymbol{n}$ and the gradient of an energy function

$$e\left(\boldsymbol{x}, \boldsymbol{n}\left(\boldsymbol{x}\right)\right) = \sum_{i=1}^{n} \left(\left(\boldsymbol{x} - \boldsymbol{p}_i\right)^T \boldsymbol{n}\left(\boldsymbol{x}\right)\right)^2 \theta\left(\boldsymbol{x}, \boldsymbol{p}_i\right).$$

Such a surface definition for the MLS surface is conducive to calculate surface characteristics such as curvatures [14].

Since the original definitions, many variants of PSSs have been developed. They differ, for example, in the strategies used to specify bandwidths (sample density [15], local feature size [16], curvatures [17]), in the surface models used to fit (planes [3], spheres [18], polynomials [1]), in the weight functions (isotropic and anisotropic weight functions [19], singular weight functions [20]), and in fitting criterion (least squares criterion, ψ-type M-Estimators [11], the maximum residual criterion [12]), and so on.

Using singular weight functions and a proper centroid function, PSSs can interpolate locations and derivatives at these locations [20]. The interpolatory PSS is suitable for noiseless data. Algebraic PSSs [18] avoid the polynomial fitting procedure by fitting spheres in the projection procedure. They use spheres instead of general polynomials because spheres are easy to be fitted and there is a close form of closest points on spheres. However, not all surfaces can be accurately approximated locally by a sphere.

The bandwidth h in weight functions is an important parameter for PSSs because a PSS with bigger bandwidth is more smooth but may smooth out small or sharp features, while a PSS surface with smaller bandwidth is more faithful to the input data but may be rough. The bandwidth h is typically

selected according to sample density of points, local feature size, curvatures, and so on.

Pauly [15] defined a bandwidth by the function $h_{\boldsymbol{x}} = c/\rho(\boldsymbol{x})$, where c is a fixed scale factor, $\rho(\boldsymbol{x}) : R^3 \rightarrow R^+$ is a continuous, smooth function approximating the local sampling density. The ρ is computed by first estimating the local sample density for each point $\boldsymbol{p}_i \in P$ by $\rho_i = k/r_i^2$ and then interpolating by standard scattered data approximation techniques, e.g. radial basis functions, where r_i is the minimum radius of the sphere centered at $\boldsymbol{p}_i$ and containing k nearest neighbors to $\boldsymbol{p}_i$.

Dey and Sun [16] take the bandwidth to be a fraction of the local feature size and define their PSS by the weight function

$$\theta(\boldsymbol{x}, \boldsymbol{p}) = e^{\frac{-\sqrt{2}\|x-p\|^2}{\rho^2 l_{fs}(\hat{x}) l_{fs}(\hat{p})}},$$

where $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{p}}$ are closest points on the sampled surface S from points $\boldsymbol{x}$ and $\boldsymbol{p}$ respectively, and $|\rho| < 1$ is a scale factor. The local feature size $l_{fs}(\boldsymbol{x})$ at a point $\boldsymbol{x} \in S$ is defined as the distance from the point to the nearest point of the medial axis of S.

Wang et al. [17] used an optimal bandwidth in the second step of the definition of Levin's PSSs. They formulated the weighted least squares polynomial fitting by the kernel regression and found the optimal bandwidth by minimizing an approximated error evaluating the kernel regression performance. In their formula, the bandwidth is selected by combining noise level, sample density, and curvatures.

In addition to the the isotropic weight function given in Eq. (1), an individual ellipsoidal weight function to each sample point is used to define PSS in [19], which is given by

$$\omega_i(\boldsymbol{x}) = \theta\left(\left\|H_i^{-1}(\boldsymbol{x} - \boldsymbol{p}_i)\right\|\right).$$

where θ is a smooth monotonically decreasing function, H_i is an ellipsoid oriented so that one of its axes points into the normal direction and the

other two align with the principal curvature directions. There also is the bandwidth selection problem for the anisotropic weight function.

In order to preserve small features and be less influenced by outliers, robust implicit PSSs are defined by combining implicit PSSs with robust local kernel regression [11]. Instead of the ordinary least squares criterion, ψ-type M-Estimators are used to assign outliers less weight, i.e. additional weight functions are used in the objective function. Besides the bandwidth used in the spatial weight function, two additional bandwidths σ_r and σ_n are introduced in the other two new weight functions. The σ_n is used in the weight function of differences between predicted gradients and sample normals. The σ_r is used in the weight function of residuals of values of implicit surface function. Smaller values of the σ_n lead to sharper results.

Based on the fact that sharp features are formed by multiple surfaces, a forward search method is introduced in [12] to find points of a smooth region by a maximum residual criterion. Sharp features are identified by intersections of surfaces. Final results are robust to outliers since the forward search method gets rid of outliers from the fitting procedure.

Moving least squares interpolation scheme is analyzed in [21], where a surface is reconstructed from point cloud data with normal vectors. The surface is defined by implicit function

$$F(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} \theta_i(\boldsymbol{x}) \left((\boldsymbol{x} - \boldsymbol{p}_i)^T \boldsymbol{n}_i \right)}{\sum_{i=1}^{n} \theta_i(\boldsymbol{x})}$$

where $\theta_i(\boldsymbol{x})$ is a weight function and $\boldsymbol{n}_i$ is the normal vector at point $\boldsymbol{p}_i$. Under some assumptions about sample densities, local feature size, and the bandwidth of the weight function, the surface $F(\boldsymbol{x}) = 0$ will lie in neighborhoods of underlying surfaces. The size of the neighborhoods are bounded by a value comparable with the point space of input point cloud. It can be proved that the projection procedure converges and the resultant surface is isotopic to the underlying surfaces [22].

In this paper, we follow the original definition of point-set surface [1],

but with the explicit goal of ensuring the residual errors between the input points and the resulting surface to be spatially independent and free from systematic bias. Although other filters such as bilateral filters on meshes [23] and feature sensitive filtering [24, 25] exist, our approach differs in several aspects, 1) input model, our approach works directly on discrete unorganized points versus polygonal mesh; 2) output model, our approach generates a continuous, implicitly defined MLS surface, rather than another polygonal mesh; 3) noise characteristic, our approach explicitly quantifies the spatial correation of the error distribution through Moran's I and statistics Z.

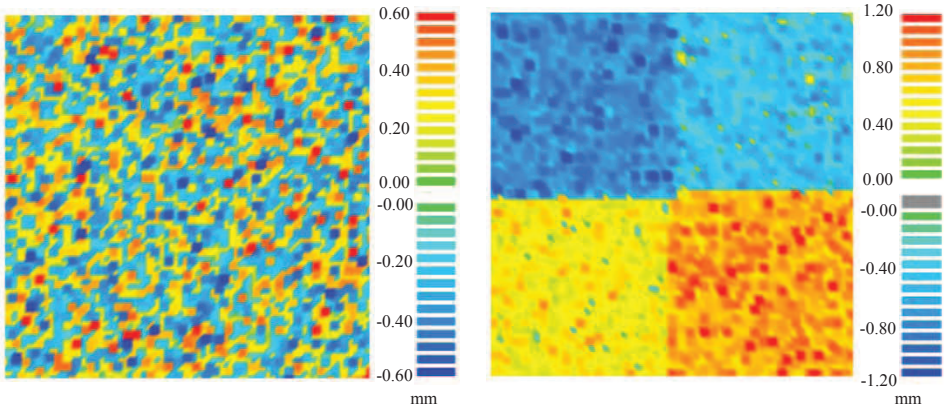
3 INTRODUCTION TO SPATIAL AUTO-CORRELATION MEASURE

Randomness is used in this paper to characterize spatial autocorrelation of some distributions. Fig. 1 shows a spatially independent pattern and a spatially dependent pattern of height distributions, where different colors denote different values of height. Randomness can be measured by Moran's I, a classic spatial autocorrelation statistic developed by Moran [26]. Spatial autocorrelation means that adjacent observations of the same phenomenon are correlated. Lower spatial autocorrelation corresponds to higher randomness. Moran's I is defined as

$$I = \frac{n}{\sum_i \sum_j \omega_{ij}} \frac{\sum_i \sum_j \omega_{ij} \left(X_i - \overline{X}\right) \left(X_j - \overline{X}\right)}{\sum_i \left(X_i - \overline{X}\right)^2}, \tag{2}$$

where n is the number of spatial units indexed by i and j; X is the variable of interest; $\overline{X}$ is the mean of X; and ω_{ij} is a matrix of spatial weights. The value of I ranges from -1 to 1. A value of I close to -1 indicates that the variables X are negative relevant. A value of I close to 1 indicates that the variables X are positive relevant. A value of I close to zero indicates that the variables X are independent.

The statistics Z transformed from the Moran's I is calculated by



(a) $I = 0.002, Z = 0.436$ (b) $I = 0.767, Z = 155$

Figure 1: (a) Spatially independent pattern; (b) Spatially dependent pattern.

$$Z = \frac{I - E(I)}{\sqrt{Var(I)}}, \tag{3}$$

where $E(I)$ and $Var(I)$ are the expected value and variance of Moran's I respectively under hypothesis of no spatial autocorrelation. The $E(I)$ is calculated by

$$E(I) = \frac{-1}{n-1}$$

The $Var(I)$ is calculated by

$$Var(I) = \frac{nS_4 - S_3 S_5}{(n-1)(n-2)(n-3) S_0^2},$$

where

$$S_0 = \sum_i \sum_j \omega_{ij},\ S_1 = \frac{1}{2} \sum_i \sum_j (\omega_{ij} + \omega_{ji})^2,\ S_2 = \sum_i \left(\sum_i \omega_{ij} + \sum_j \omega_{ji} \right)^2,$$

$$S_3 = \frac{n \sum_i \left(X_i - \overline{X}\right)^4}{\left(\sum_i \left(X_i - \overline{X}\right)^2\right)^2},\ S_4 = \left(n^2 - 3n + 3\right) S_1 - n S_2 + 3 S_0^2,\ S_5 = 6 S_0^2.$$

The following hypothesis is tested by the statistics Z:

- H_0: Deviations are of spatially statistical independence.

- H_a: Deviations are of spatially statistical dependence.

Following convention [13], a significance level of 0.02 is adopted which means the probability of rejecting the hypothesis H_0 is 0.02. The corresponding critical value for the test statistics Z is $Z_{0.02} = 2.33$. Values of the statistics Z greater than 2.33 or smaller than -2.33 indicate spatial autocorrelation that is significant at the 2% level. if $|Z| < Z_{0.02}$, the hypothesis H_0 is accepted. Otherwise, the hypothesis H_a is accepted. A higher significant level corresponds to the smaller critical value of the statistics Z. For example, values of the statistics Z greater than 1.96 or smaller than -1.96 indicate spatial autocorrelation that is significant at the 5% level.

4 SPATIAL VARIATION OF RESIDUALS IN RECONSTRUCTED PSSs

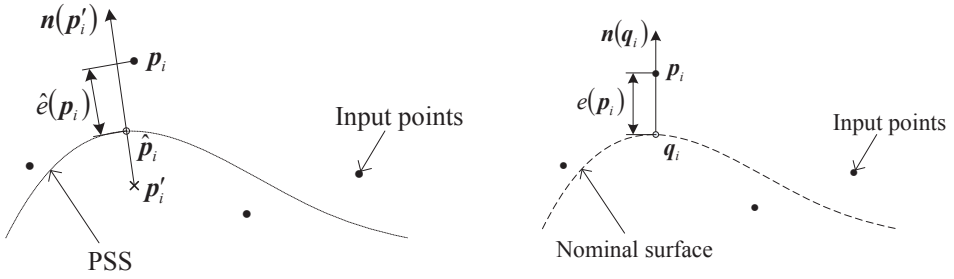
In this section, we briefly describe several concepts including residual errors in surface reconstructions, the randomness measure for characterizing the spatial autocorrelation of such residuals, and potential systematic bias in such an error measure. These concepts form the basis for subsequent presentation of our proposed method in reconstructing the PSS.

4.1 Residual errors in surface reconstruction

We first describe the PSS computing process and then the residual errors of the input points with respect to both the resulting PSS surface and the nominal surface.

Assume $P = \{\boldsymbol{p}_i, i = 1, \ldots, n\}$ is an input point set. We adopt the following two steps to define a PSS [1]:

1. MLS projection procedure: In the jth iteration, fit a plane to the neighborhood of point $\boldsymbol{p}_i^j$. Let $\boldsymbol{p}_i^{j+1}$ be the projection point of $\boldsymbol{p}_i^j$ on the plane. The iteration is ended when the Euclidean norm $\left\|\boldsymbol{p}_i^j - \boldsymbol{p}_i^{j+1}\right\|$



(a) (b)

Figure 2: a) error between input points and the PSS; b) error between input (sampled) points and the corresponding nominal surface.

is smaller than a given threshold. The termination produces $\boldsymbol{p}_i'$ and $\boldsymbol{n}(\boldsymbol{p}_i')$, $i = 1, \ldots, n$, where $\boldsymbol{p}_i'$ is the projection point of $\boldsymbol{p}_i$, and $\boldsymbol{n}(\boldsymbol{p}_i')$ is the computed normal vector at point $\boldsymbol{p}_i'$.

2. Polynomial fitting procedure: Fit a polynomial surface to the neighborhood of each point $\boldsymbol{p}_i'$ and find the intersection point between the surface and the line passing $\boldsymbol{p}_i'$ and parallel with $\boldsymbol{n}(\boldsymbol{p}_i')$. Then the intersection point $\widehat{\boldsymbol{p}}_i$ is the point on the PSS.

Denote the point set on the PSS obtained by the above procedure as $\widehat{P} = \{\widehat{\boldsymbol{p}}_i, i = 1, \ldots, n\}$, where point $\widehat{\boldsymbol{p}}_i$ corresponds to point $\boldsymbol{p}_i$. Note, for notational convenience the PSS defined by the point set $\widehat{P}$ is sometimes simply noted as $\widehat{P}$. The signed error at point $\boldsymbol{p}_i$ is given by

$$\widehat{e}(\boldsymbol{p}_i) = \langle \boldsymbol{p}_i - \widehat{\boldsymbol{p}}_i, \boldsymbol{n}(\boldsymbol{p}_i') \rangle \tag{4}$$

where $\langle\rangle$ denotes the inner product between two vectors. It is also shown in Fig. 2.a. A consistent orientation of the normal vectors is necessary to define errors over the input points, which can be obtained by a minimum spanning tree [27] or some prior knowledge.

Similarly we can define the residuals of input (sampled) points with respect to a nominal surface. Assume

$$\boldsymbol{p}_i = \boldsymbol{q}_i + \varepsilon \boldsymbol{n}(\boldsymbol{q}_i) \tag{5}$$

where $\boldsymbol{q}_i$ is a point on the nominal surface corresponding to the input point $\boldsymbol{p}_i$, $\boldsymbol{n}(\boldsymbol{q}_i)$ is the unit normal vector of the nominal surface at $\boldsymbol{q}_i$, and ε is a random noise with zero mean. Similar to Eq. (4), the signed error at $\boldsymbol{p}_i$ with respect to the nominal surface is given by

$$e(\boldsymbol{p}_i) = \langle \boldsymbol{p}_i - \boldsymbol{q}_i, \boldsymbol{n}(\boldsymbol{q}_i) \rangle \tag{6}$$

The error is also shown in Fig. 2.b. Under the assumption about the noise ε, the mean of $e(\boldsymbol{p}_i)$ should be zero and the statistics Z of $e(\boldsymbol{p}_i)$ should be $|Z| < 2.33$ for the significance level of 0.02.

One ideal way to measure the quality of a PSS is to compare it with nominal surfaces. However, nominal surfaces are usually not known for an input point set. An alternative to bypass the problem is to measure differences between input points and the constructed PSS. If the differences satisfy some priori knowledge of noises of the input points, the PSS can be considered as a proper approximate of the nominal surface. That is, when the PSS is properly reconstructed, the $e(\boldsymbol{p}_i)$ can be estimated by the $\widehat{e}(\boldsymbol{p}_i)$ given by Eq. (4).

4.2 Computing Moran's I and statistics Z of the residuals

Based on the above definition of the residual errors, the Moran's I of the residual errors between input points and the resulting PSS can be described as

$$I(\widehat{\boldsymbol{e}}) = n\widehat{\boldsymbol{e}}^T W \widehat{\boldsymbol{e}}, \tag{7}$$

where $\widehat{\boldsymbol{e}} = [(\widehat{e}(\boldsymbol{p}_1) - \widehat{\mu}), \ldots, (\widehat{e}(\boldsymbol{p}_n) - \widehat{\mu})]^T / \left\| [(\widehat{e}(\boldsymbol{p}_1) - \widehat{\mu}), \ldots, (\widehat{e}(\boldsymbol{p}_n) - \widehat{\mu})]^T \right\|$, $\widehat{\mu}$ is the mean of the error $\widehat{e}(\boldsymbol{p}_i), i = 1, \ldots, n$, W is the $n \times n$ weight matrix, the ith row and jth column of W is $\omega_{ij} / \sum_{k,l} \omega_{kl}$, and ω_{ij} is spatial weights. The spatial weights ω_{ij} same as those in [13] are used in this paper, i.e.

$$\omega_{ij} = \begin{cases} d_{ij}^{-4}/L_i, & i \neq j \\ 0, & i = j \end{cases} \tag{8}$$

where $L_i = \sum_{j=1, j\neq i}^{n} d_{ij}^{-4}$ and d_{ij} is the distance between point $\widehat{\boldsymbol{p}}_i$ and point $\widehat{\boldsymbol{p}}_j$.

The error $\widehat{e}(\boldsymbol{p}_i)$ is defined by Eq. (4) and is calculated in each iteration that adjusts the bandwidth in our method. Assume $\widehat{\boldsymbol{p}}_i^{(k)}$ is the corresponding point on PSS of point $\boldsymbol{p}_i$ at the kth iteration. Then the error $\widehat{e}(\boldsymbol{p}_i)$ at the kth iteration is calculated by

$$\widehat{e}^{(k)}(\boldsymbol{p}_i) = \left\langle \boldsymbol{p}_i - \widehat{\boldsymbol{p}}_i^{(k)}, \boldsymbol{n}(\boldsymbol{p}_i') \right\rangle \tag{9}$$

Note, the I in Eq. (7) has been transformed into a matrix form. The advantage of the matrix form (7) over the original form (2) is that it is efficient to calculate the Moran's I by the matrix form Eq. ((7)) through vectorized computation. Further, it is clear from the matrix form that the relationship between the Moran's I and the error $\widehat{\boldsymbol{e}}$ is determined by the weight matrix W.

4.3 Systematic bias effect in randomness measure

The randomness measure presented above does not characterize any potential systematic bias in the residuals. Assume $\widehat{e}(\boldsymbol{p}_i) = \widehat{e}_{s_i} + \widehat{e}_{r_i}$, where $\widehat{e}_{s_i}$ and $\widehat{e}_{r_i}$ denote systematic and random errors respectively. If all the systematic deviations are equal to the same constant c, i.e. $\widehat{e}_{s_i} = c$, then the $\widehat{\mu}$ in Eq. (7) is given by

$$\widehat{\mu} = \frac{1}{n}\sum_{i=1}^{n} \widehat{e}_{r_i} + c = \widehat{\mu}_r + c.$$

where $\widehat{\mu}_r$ is the mean of the error $\widehat{e}_{r_i}$. From above equations, we have

$$\widehat{e}(\boldsymbol{p}_i) - \widehat{\mu} = \widehat{e}_{r_i} - \widehat{\mu}_r. \tag{10}$$

Substituting Eq. (10) into Eq. (7), we have

$$I(\widehat{\boldsymbol{e}}) = I(\widehat{\boldsymbol{e}}_r) \tag{11}$$

where $\widehat{\boldsymbol{e}}_r = [(\widehat{e}_{r_1} - \widehat{\mu}_r), \ldots, (\widehat{e}_{r_n} - \widehat{\mu}_r)]^T / \left\| [(\widehat{e}_{r_1} - \widehat{\mu}_r), \ldots, (\widehat{e}_{r_n} - \widehat{\mu}_r)]^T \right\|$. Similarly, we have

$$Z(\widehat{\boldsymbol{e}}) = Z(\widehat{\boldsymbol{e}}_r). \tag{12}$$

It is clear from Eqs. (11) and (12) the Moran's I and statistics Z of residual errors with respect to a surface are the same as those with respect to its offset surfaces. This observation forms the basis for our subsequent bias offsetting operation in the proposed PSS reconstruction method in order to eliminate the systematic error.

5 Computing PSS with controlled spatial variation of residuals

Based on the above residual analysis of a PSS, a method is presented below to compute a PSS from input points with controlled spatial variation of residual errors and reduced systematic bias.

Assume $P = \{\boldsymbol{p}_i, i = 1, \ldots, n\}$ is input point set and let $\widetilde{P} = \{\widetilde{\boldsymbol{p}}_i, i = 1, \ldots, n\}$ as the final output point set with bias corrected, where point $\boldsymbol{p}_i$ corresponds to point $\widetilde{\boldsymbol{p}}_i$. The error at point $\boldsymbol{p}_i$ with respect to the resulting surface can be measured by

$$\widetilde{e}\left(\boldsymbol{p}_i\right) = \langle \boldsymbol{p}_i - \widetilde{\boldsymbol{p}}_i, \boldsymbol{n}\left(\boldsymbol{p}_i'\right)\rangle \tag{13}$$

where $\boldsymbol{p}_i'$ is the MLS projection point of $\boldsymbol{p}_i$, $\boldsymbol{n}\left(\boldsymbol{p}_i'\right)$ is the unit normal vector computed during the MLS projection procedure.

The resulting point set $\widetilde{P}$ should be on a smooth surface and satisfy the following conditions:

1. the mean of $\widetilde{\boldsymbol{e}}$ is equal to zero, and
2. $|Z\left(\widetilde{\boldsymbol{e}}\right)| < 2.33$ for the significance level 0.02.

where $\widetilde{\boldsymbol{e}} = \left[\left(\widetilde{e}\left(\boldsymbol{p}_1\right) - \widetilde{\mu}\right), \ldots, \left(\widetilde{e}\left(\boldsymbol{p}_n\right) - \widetilde{\mu}\right)\right]^T / \left\|\left[\left(\widetilde{e}\left(\boldsymbol{p}_1\right) - \widetilde{\mu}\right), \ldots, \left(\widetilde{e}\left(\boldsymbol{p}_n\right) - \widetilde{\mu}\right)\right]^T\right\|$ and $\widetilde{\mu}$ is the mean of the error $\widetilde{e}\left(\boldsymbol{p}_i\right)$.

The point set $\widetilde{P}$ is obtained in a two-phase process: 1) by defining a PSS from $\widehat{P}$ satisfying $|Z\left(\widehat{\boldsymbol{e}}\right)| < 2.33$, and 2) eliminating the systematic error $\widehat{\mu} = \sum_{i=1}^{n} \widehat{e}\left(\boldsymbol{p}_i\right)/n$ from $\widehat{P}$ to obtain $\widetilde{P}$, where $\widehat{\boldsymbol{e}}$ is the error vector in Eq. (7).

For the first condition, the random spatial variation of residuals can be achieved by making $I = 0$. The vector $\widehat{\boldsymbol{e}}$ in Eq. (7) making $I = 0$ should be a vector in the null space of W. However the null space of W may be empty or consist of infinite number of points including ones that may not represent errors of smooth surfaces. Moreover, it is difficult to compute the null space due to a big dimension n of the matrix W and the $O(n^2)$ complexity of the computation. We bypass the problem by using the corresponding statistics Z which has an explicit threshold for a given significance level. The significance level of 0.02 is adopted and it results in $|Z| < 2.33$.

Since different bandwidths produce different PSSs and thus result in different errors $\widehat{\boldsymbol{e}}$ in Eq. (7) and statistics Z of the error. We can view statistics Z as a function of the bandwidth used in the polynomial fitting procedure. (Note, the initial bandwidth used in the MLS projection has little influence on the final result as later shown in Section 6.1.) If $|Z(\widehat{\boldsymbol{e}})| \geq 2.33$ for current bandwidth, the bandwidth is adjusted by minimizing Z^2 until $|Z| < 2.33$. The increment of the bandwidth is obtained according to the Gauss-Newton algorithm and given by

$$\Delta h = -\frac{Z}{dZ/dh}, \tag{14}$$

where dZ/dh is estimated by $(Z(h + \delta h) - Z(h - \delta h)) / (2\delta h)$ and δh is a small increment of the bandwidth.

The overall PSS reconstruction process is as follows:

1. Input point set $P = \{\boldsymbol{p}_i, i = 1, \ldots, n\}$ and set initial bandwidth $h^{(0)}$.

2. Obtain projection point set $P' = \{\boldsymbol{p}'_i, i = 1, \ldots, n\}$ and estimated normal vector $\boldsymbol{n}(\boldsymbol{p}'_i)$ at point $\boldsymbol{p}'_i$ by the MLS projection procedure.

3. Obtain point set $\widehat{P}^{(k)} = \left\{\widehat{\boldsymbol{p}}_i^{(k)}, i = 1, \ldots, n\right\}$ on PSS by the polynomial fitting procedure, where k counts the iteration of the following process that adjusts the bandwidth $h^{(k)}$.

4. Compute signed error $\widehat{e}^{(k)}(\boldsymbol{p}_i)$ by Eq. (9). Calculate Moran's I and statistics Z according to $\widehat{\boldsymbol{p}}_i^{(k)}$ and $\widehat{e}^{(k)}(\boldsymbol{p}_i)$.

5. If $|Z| < 2.33$ or $k = N_{\max}$, go to step 6, where $N_{\max}$ is the allowed maximum number of iterations. Otherwise $h^{(k+1)} = h^{(k)} + \Delta h$ and then $k = k + 1$, go to step 3, where Δh is calculated by Eq. (14).

6. Assume $\widehat{\mu} = \sum_{i=1}^{n} \widehat{e}^{(k)}(\boldsymbol{p}_i)/n$. Define the point set $\widetilde{P} = \{\widetilde{\boldsymbol{p}}_i, i = 1, \ldots, n\}$ by $\widetilde{\boldsymbol{p}}_i = \widehat{\boldsymbol{p}}_i^{(k)} - \widehat{\mu}\boldsymbol{n}(\boldsymbol{p}_i')$. Then $\widetilde{P}$ is the resulting point set with residuals spatially random and bias corrected.

7. (Optional) Define a PSS from the point set $\widetilde{P}$ and normal vectors $\boldsymbol{n}(\boldsymbol{p}_i')$.

In the Step 2 of the method, the bandwidth used in the projection is $h^{(0)}$. The point set P' remains unchanged in the rest of the process. The $h^{(0)}$ will be used as the initial value of the bandwidth of the polynomial fitting procedure.

In the Step 3 of the method, the polynomial surface is fitted in the local coordinate system whose z axis is parallel with $\boldsymbol{n}(\boldsymbol{p}_i')$. The polynomial used is quadratic. There are two reasons for using quadratic polynomials. First, any free form surfaces can be approximated locally by quadratic polynomials; second, higher degree polynomials are more sensitive to noises. In our experiments, quadratic polynomials show better results than cubic or quartic polynomials. The bandwidth used in the fitting procedure is $h^{(k)}$ and the neighborhood consists of points in point set P whose distances to $\boldsymbol{p}_i^{'}$ is no greater than $3h^{(k)}$.

The formula of the fitted polynomial is given by $z = f(x, y)$ in local coordinate systems. Then there should be at most one intersection point between the underlying surface and each line parallel to the z axis. Note that the normal vector of the underlying surface is estimated by $\boldsymbol{n}(\boldsymbol{p}_i')$. If the included angle between $\boldsymbol{n}(\boldsymbol{p}_i')$ and the normal vector at a point near $\boldsymbol{p}_i'$ is greater than 90 degrees, the $z = f(x, y)$ is most likely not monotone in the neighborhood of $\boldsymbol{p}_i'$. Then the range of the neighborhood of $\boldsymbol{p}_i'$ is restricted in the fitting process by not using the point at which the normal vector has an included angle greater than 90 degrees from $\boldsymbol{n}(\boldsymbol{p}_i')$.

In the Step 4 of the method, signed distances are defined according to the computed normal vector $\boldsymbol{n}(\boldsymbol{p}_i')$. Consistent directions of these normal vectors are obtained by a minimum spanning tree [27] or some prior knowledge.

In the optional Step 7 of the method, a PSS can be defined from the noise-filtered point set $\widetilde{P}$. An interpolatory PSS can be defined for example by [20]:

$$f(\boldsymbol{x}) = \boldsymbol{n}(\boldsymbol{x})^T (\boldsymbol{x} - c(\boldsymbol{x})) = 0, \tag{15}$$

where

$$\boldsymbol{n}(\boldsymbol{x}) = \frac{\sum_i \theta(\|\boldsymbol{x} - \widetilde{\boldsymbol{p}}_i\|)\, \boldsymbol{n}_i}{\sum_i \theta(\|\boldsymbol{x} - \widetilde{\boldsymbol{p}}_i\|)},\ c(\boldsymbol{x}) = \frac{\sum_i \theta(\|\boldsymbol{x} - \widetilde{\boldsymbol{p}}_i\|)\, \widetilde{\boldsymbol{q}}_i}{\sum_i \theta(\|\boldsymbol{x} - \widetilde{\boldsymbol{p}}_i\|)},$$

$$\widetilde{\boldsymbol{q}}_i = \boldsymbol{x} - \left(\boldsymbol{n}_i^T (\boldsymbol{x} - \widetilde{\boldsymbol{p}}_i)\right) \boldsymbol{n}_i,\ \boldsymbol{n}_i = \boldsymbol{n}(\boldsymbol{p}_i'),\ \theta(\|\boldsymbol{x} - \widetilde{\boldsymbol{p}}_i\|) = \|\boldsymbol{x} - \widetilde{\boldsymbol{p}}_i\|^{-2}.$$

6 EXPERIMENTS

In this section, we present results from our approach and compare them with those from another adaptive MLS projection by Pauly [15]. We first describe how we choose the initial bandwidth. We then present the results on both synthetic data where the nominal surface is available and real scanned data. We discuss how this method works with different noises and sampling densities. The effect of sharp features is also briefly discussed with an example. Our method is implemented on Matlab 7.6. All the experiments are carried out on a computer with Intel Pentium 3GHz CPU and 2 GB of RAM. The computational cost of our method increases with the number of input data, the number of neighboring points used in MLS projection and polynomial fitting, and the degree of the polynomial during the fitting. In our experiments, around 100 MLS projections per second or 50 quadric polynomial fit per second can be achieved when the bandwidth h is specified to be at 2 to 3 times of average point separation in the data cloud and the radius of the neighborhood equals to $3h$.

Pauly's adaptive MLS projection [15] is compared in our experiments. According to the suggestion in [28], the number of neighbors used to estimate local sample density is specified at $k = 15$, which is then used to obtain the adaptive Gaussian kernel width h in Pauly's MLS projection. The Pauly's MLS projection is done through Pointshop3D which is an interactive system for point-based surface editing [28, 29]. The Pauly's MLS projection is followed by a quadric polynomial fitting in Pointshop3D.

6.1 Initial bandwidth

The bandwidth in the MLS surface depends on both point density and the feature size. More specifically, it should be smaller than the local feature size and be larger than the average point separation s_d. The LFS (Local Feature Size) [21] at a point $\boldsymbol{p} \in S$ is defined as the distance from $\boldsymbol{p}$ to the nearest point of the medial axis of S. The medial axis of S consists of points from which there are two or more closest points in S. Assume the LFS at at a point $\boldsymbol{p}$ is denoted by $l_{fs}(\boldsymbol{p})$. Then the bandwidth at $\boldsymbol{p}$ should not be greater than $\varepsilon_1 l_{fs}(\boldsymbol{p})$, where ε_1 is a given constant usually smaller than 1. Note, in practice, we do not compute the local feature size. However, the concept of local feature size provides users a sense how large the initial h can be. The initial bandwidth is also related to the sample density of the input points. Assume the distance from $\boldsymbol{p} \in P$ to its nearest neighbor in P is $s_d(\boldsymbol{p})$. The bandwidth at $\boldsymbol{p}$ should be greater than $\varepsilon_2 s_d(\boldsymbol{p})$, where ε_2 is a given constant usually greater than 0.5. That is, the initial bandwidth should be chosen within the ranges $[\varepsilon_2 s_d(\boldsymbol{p}), \varepsilon_1 l_{fs}(\boldsymbol{p})]$.

An example on torus is given below. For a point $\boldsymbol{p}$ on the torus shown in Fig. 3, $l_{fs}(\boldsymbol{p}) = r_2$ if $r_1 \geq 2r_2$. A point set consisting of 6227 points is sampled from a torus with $r_1 = 15$ and $r_2 = 5$. The average point space of the point set is about 0.60. Then the point set is contaminated by noise with zero mean ($\mu_0 = 0$) and 0.06 standard deviation ($\sigma_0 = 0.06$). Table 1 gives results of our method applied on the torus using different initial bandwidths, where Z, μ, σ, and μ_{mse} denote statistics Z, mean, standard deviation, and

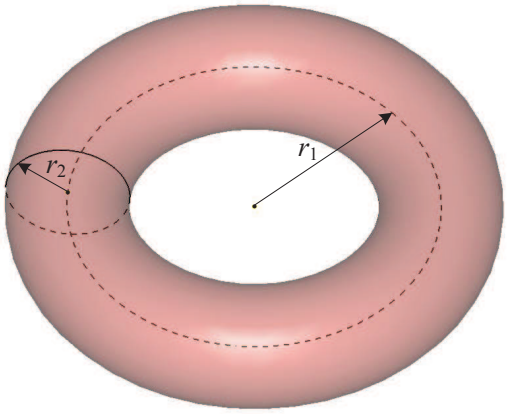


Figure 3: The torus used in experiments.

mean squared error of differences between the input data P and resultant data $\widetilde{P}$ respectively. These initial bandwidths range from one to six times of the average point space. As the table shows, under these different initial bandwidths,

- the condition $|Z| \leq 2.33$ is satisfied for all initial bandwidths.
- μ and σ are identical to μ_0 and σ_0 to the fourth digit in all cases.

This example demonstrates that, due to the built-in iterative search for the optimized bandwidth, there is a wide range of initial widths that all lead to accurate PSS surfaces with random deviations.

Table 1: Results of the torus with different initial bandwidths

Initial bandwidth	Z	μ	σ	μ_{mse}
0.6	0.59	0	0.0601	0.0036
1.2	1.98	0	0.0604	0.0037
1.8	0.73	0	0.0602	0.0036
2.4	1.19	0	0.0603	0.0036
3	-1.14	0	0.0600	0.0036
3.6	0.75	0	0.0602	0.0036

6.2 Experimental results on synthetic data

Experimental results for a set of synthetic shapes including a sphere, a cylinder, a hyperboloid, a torus and a freeform wavy surface are shown in Fig. 4 - Fig. 8. In the remainder of this paper, we adopt the following notation: the input data P, the resultant point set P' from Pauly's method, the resultant point set of our method $\widetilde{P}$, and the nominal surfaces S respectively. Differences between P and S, P' and S, $\widetilde{P}$ and S are also shown in Fig. 4 - Fig. 8. Table 2 shows characteristics of the input data, initial and resultant bandwidths, where n, s_d, μ_0, and σ_0 denote number of input points, average point space, mean and standard deviation of added noises with respect to the respective nominal surfaces.

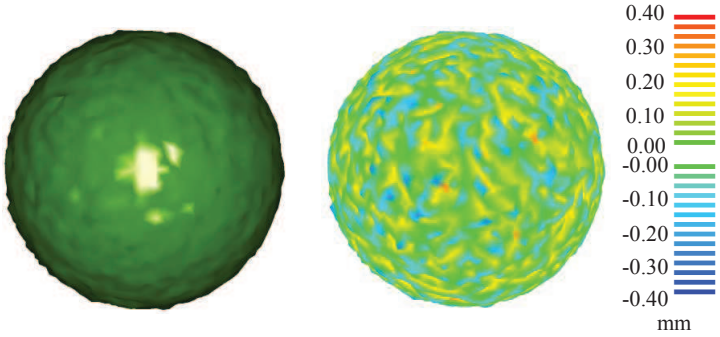
Table 2: Input data, initial bandwidths and resultant bandwidths

Examples	n	s_d	μ_0	σ_0	Initial h	Resultant h
Sphere	2,606	0.59	0	0.1	1.5	7.34
Cylinder	5,166	0.50	0	0.1	1.5	3.04
Hyperboloid	10,032	0.44	0	0.1	1.5	3.85
Torus	6,227	0.60	0	0.06	1.0	3.17
Wavy surface	40,000	0.08	0	0.02	0.5	0.5

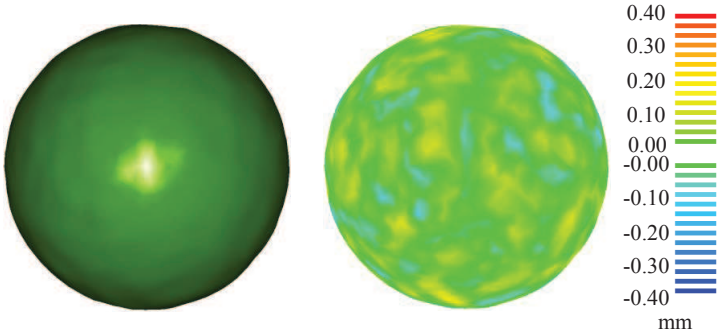
The statistics of the results, including statistics Z, mean error μ, standard deviation σ, and mean squared error (denoted by μ_{mse}) of the differences between P and S, P and $\widetilde{P}$, are given in Table 3. As the results show,

- The differences between the input data P and the noise-filtered resulting data $\widetilde{P}$ have become random at the significance level 0.02 (that is, all Zs are smaller than 2.33). This means that random noises have been removed in $\widetilde{P}$.

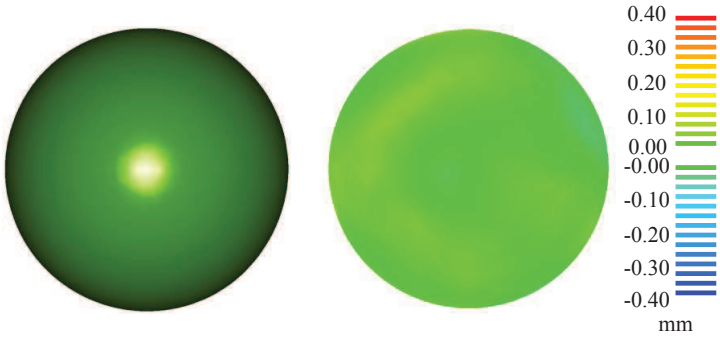
- The mean and standard deviation of the input data with respect to the resultant data $\widetilde{P}$ are similar to those with respect to the nominal surfaces S. The differences of the mean and standard deviation of the



(a) Input data P and differences between P and S

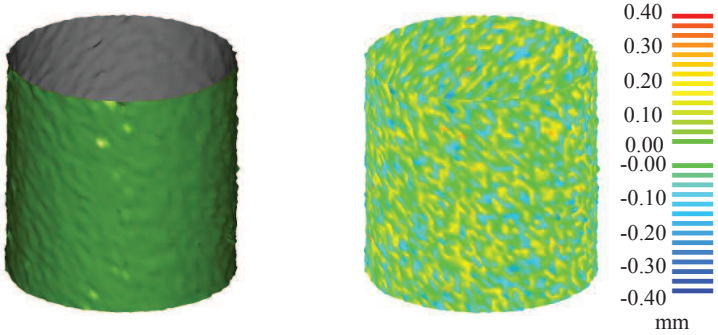


(b) Output data P' of Pauly's method and differences between P' and S

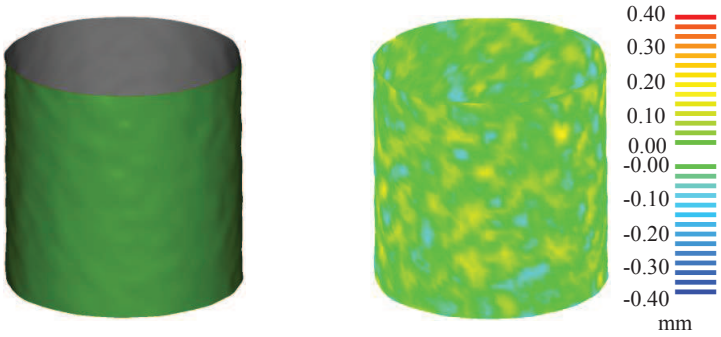


(c) Output data $\widetilde{P}$ of our method and differences between $\widetilde{P}$ and S

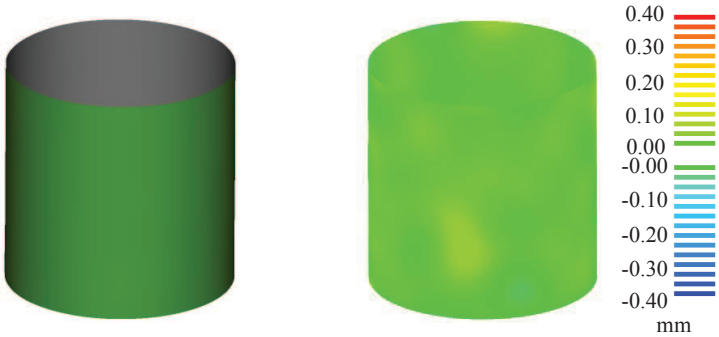
Figure 4: Results obtained from a sphere.



(a) Input data P and differences between P and S

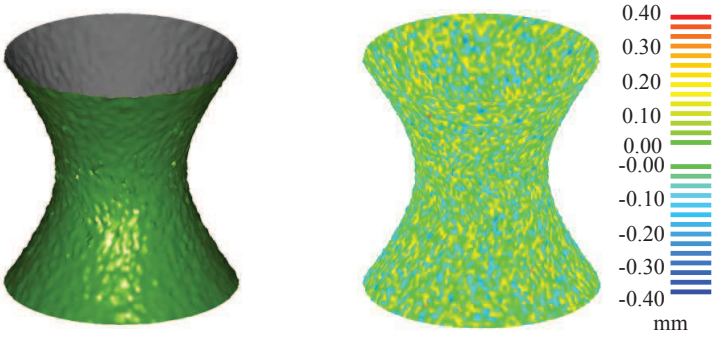


(b) Output data P' of Pauly's method and differences between P' and S

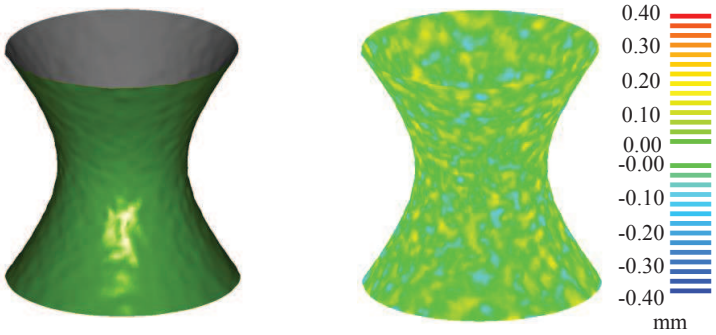


(c) Output data $\widetilde{P}$ of our method and differences between $\widetilde{P}$ and S

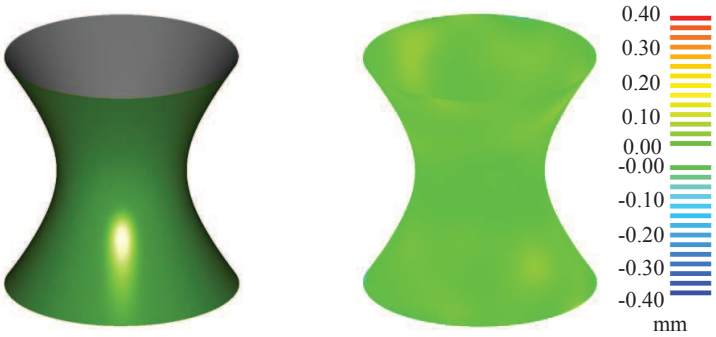
Figure 5: Results obtained from a cylinder.



(a) Input data P and differences between P and S

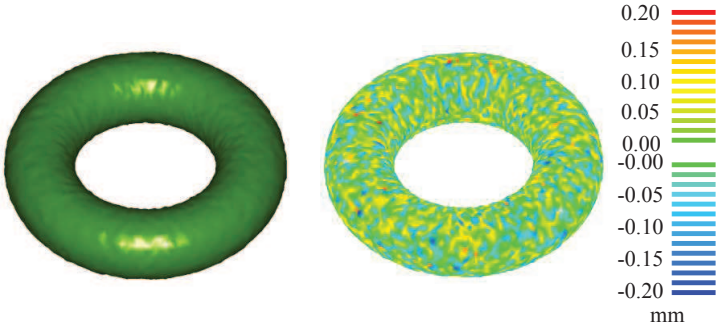


(b) Output data P' of Pauly's method and differences between P' and S

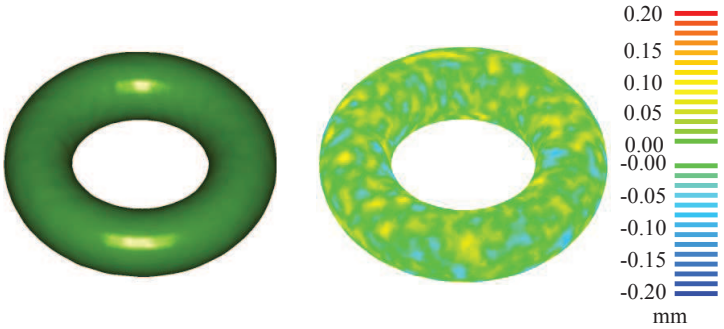


(c) Output data $\widetilde{P}$ of our method and differences between $\widetilde{P}$ and S

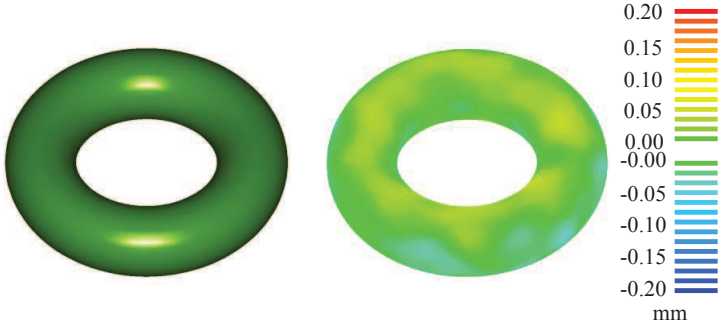
Figure 6: Results obtained from a hyperboloid.



(a) Input data P and differences between P and S

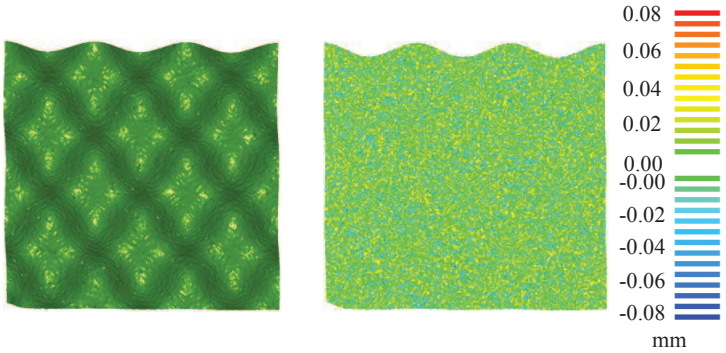


(b) Output data P' of Pauly's method and differences between P' and S

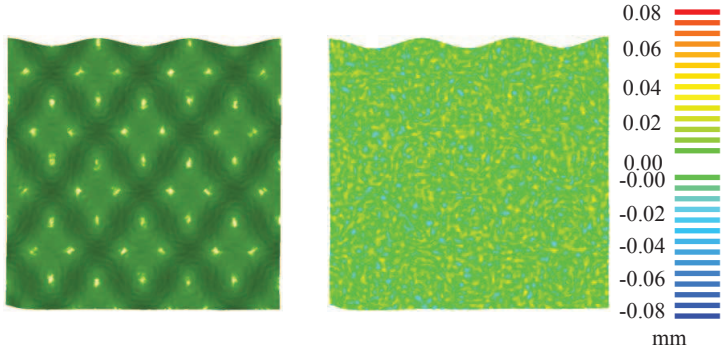


(c) Output data $\widetilde{P}$ of our method and differences between $\widetilde{P}$ and S

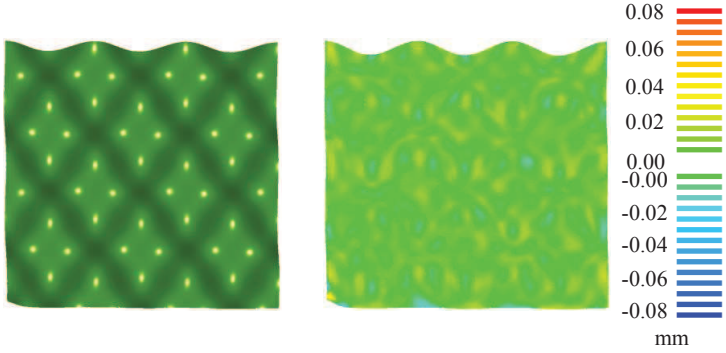
Figure 7: Results obtained from a torus.



(a) Input data P and differences between P and S



(b) Output data P' of Pauly's method and differences between P' and S



(c) Output data $\widetilde{P}$ of our method and differences between $\widetilde{P}$ and S

Figure 8: Results obtained from a wavy surface.

input data with respect to nominal surfaces versus with respect to the reconstructed $\widetilde{P}$ are smaller than 5 percents. This further confirms that the resulting point set surfaces $\widetilde{P}$ approximates true surfaces as the residuals become random.

Table 3: Differences between input point set P and nominal surface S, P and estimated point set $\widetilde{P}$ on S

Examples	P and S				P and $\widetilde{P}$			
	Z	μ_0	σ_0	μ_{mse0}	Z	μ	σ	μ_{mse}
Sphere	-1.15	0	0.1	0.01	-0.15	0	0.1006	0.010
Cylinder	2.48	0	0.1	0.01	-1.06	0	0.0985	0.0097
Hyperboloid	-0.44	0	0.1	0.01	-1.04	0	0.0997	0.0099
Torus	-0.52	0	0.06	0.0036	1.81	0	0.0604	0.0036
Wavy surface	-0.58	0	0.02	0.0004	-2.07	0	0.0200	0.0004

Table 4: Deviations between nominal surface S and point set $\widehat{P}$ (before compensation) and between nominal surface S and point set $\widetilde{P}$ (after compensation)

Examples	S and $\widehat{P}$			S and $\widetilde{P}$		
	μ	σ	μ_{mse}	μ	σ	μ_{mse}
Sphere	-0.477	0.018	0.23	-0.000008	0.018	0.00032
Cylinder	-0.0070	0.012	0.00019	0.000020	0.012	0.00014
Hyperboloid	-0.017	0.012	0.00043	0.000024	0.012	0.00015
Torus	-0.059	0.016	0.0037	0.000001	0.016	0.00026
Wavy surface	0.000043	0.0043	0.000018	-0.000005	0.0043	0.00002

Table 4 further compares the effect of correcting systematic bias (Step 6 of the method described in Section 5). It shows that

- The point set with controlled spatial distribution of residuals without systematic bias correction, $\widehat{P}$, exhibited larger mean errors μ and larger

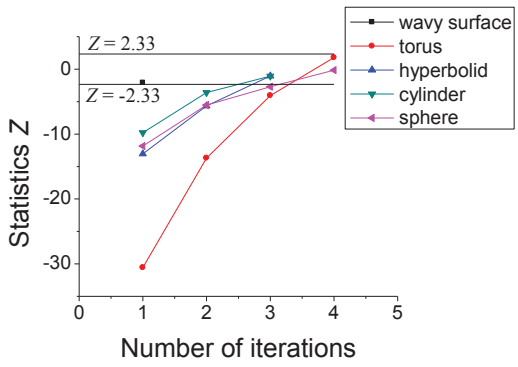


Figure 9: Statistics Z with respect to the number of iterations of our method.

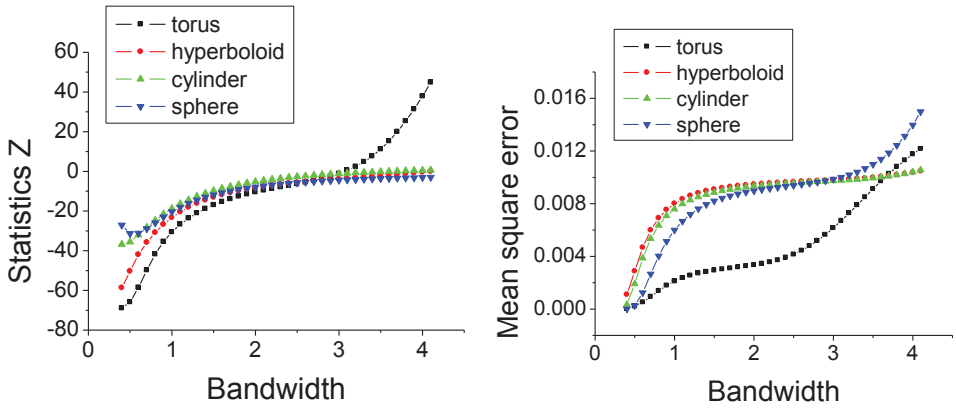


Figure 10: The statistics Z and mean squared error of differences between input point set P and point set $\widehat{P}$ before compensating for systematic errors.

mean squared error μ_{mse} to the nominal surface S than the point set $\widetilde{P}$ with compensation to these nominal surfaces. Therefore, as the tables shows, the compensation removes the systematic bias.

Fig. 9 shows the convergence of our method in various examples. The stop criterion of the iteration is given by $|Z| < 2.33$ for the significance level 0.02 and the two lines $|Z| = 2.33$ are plotted in the figure. As the figure shows, only several number of iterations are needed for all the examples. The initial and resulting h are the same for the wavy surface in Table 2. It suggests that the initial h happens to produce a MLS surface satisfying the randomness requirement. It also indicates that such bandwidth is not

unique.

Table 5 details the parameters used in Pointshop3D for the synthetic data. Compared with surfaces obtained by our method shown in Fig. 4.c - Fig. 8.c, surfaces obtained by Pauly's method shown in Fig. 4.b - Fig. 8.b are less smooth. Color plots of Fig. 4.b - Fig. 8.b and Fig. 4.c - Fig. 8.c visually indicate that results of our method are closer to nominal surfaces than those from PointShop3D with Pauly's method.

Quantitative results are in Table 6. Left column of Table 6 gives the statistics Z, mean, standard deviation, mean square error of the differences between the input data and the data obtained by Pauly's method. Right column of Table 6 gives the mean, standard deviation, mean square error of the differences between nominal surfaces and the data obtained by Pauly's method. Compared with the right column of Table 3, the differences between P and P' in Table 6 are not random, and mean and standard deviation of the differences are rather different to those of added noises. Compared with the right column of Table 4, the difference between S and P' in Table 6 are bigger. This indicates that results of our method are closer to nominal surfaces than those from PointShop3D with Pauly's method.

Table 5: Parameters of Pointshop3D for the synthetic data

		Brent Minimization Options		
Examples	scale radius	Max Iteration	Tolerance	Search Range
Sphere	20.0	10	1e-4	6
Cylinder	20.0	10	1e-4	6
Hyperboloid	20.0	10	1e-4	3
Torus	20.0	10	1e-4	3
Wavy surface	200.0	10	1e-4	1

The above experiments demonstrate that the proposed method based on controlled spatial variation of residuals leads to accurate reconstruction of point-set surfaces from input data. Fig. 10 illustrates how the statistics

Table 6: Differences between input point set P and resultant point set P' of Pauly's method, nominal surface S and point set P'.

Examples	P and P'				S and P'		
	Z	μ	σ	μ_{mse}	μ_0	σ_0	μ_{mse0}
Sphere	-15.29	0.00176	0.088	0.0077	0.00173	0.039	0.0015
Cylinder	-19.36	-0.00045	0.089	0.0079	-0.00044	0.039	0.0015
Hyperboloid	-28.76	0.00050	0.090	0.0081	0.00050	0.039	0.0015
Torus	-23.43	-0.00051	0.053	0.0028	-0.00052	0.024	0.0006
Wavy surface	-53.02	0.00001	0.018	0.0003	0.00001	0.0077	0.0001

Z and mean squared error vary over different bandwidths. These are the errors between P and $\widehat{P}$. It illustrates that directly minimizing mean squared error cannot produce a proper value of the bandwidth, while minimizing the squared Z can obtain one.

6.3 Experimental result on real scanned data

The application of this method on actual scanned data has also been conducted. Fig. 11.a shows a sculptured face (P) obtained with Minolta Vivid 9i where the actual noise is unknown. Fig. 11.b and Fig. 11.c show results obtained by Pauly's method and our method respectively. Parameters of Pointshop3D for the scanned data are given in Table 8.

Fig. 11.d and Fig. 11.e show differences between the input data and results of the face. Fig. 12.a shows the Stanford bunny [30]. Fig. 12.b and Fig. 12.c show results obtained by Pauly's method and our method respectively. Fig. 12.d and Fig. 12.e show differences between the input data and results of the bunny. As Fig. 11 and Fig. 12 graphically show that noise in the measurement data has been suppressed by both methods.

Quantitative results of the face and the bunny are given in Table 7 and Table 9. As indicated by these tables, differences between input data and results of our method are more random and have lower means and standard

deviations than those of Pauly's method.

Table 7: Results on the sculpture and the bunny of our method

Examples	n	s_{dis}	Z	μ	σ	μ_{mse}	Resultant h
Sculpture	18,498	0.45	0.25	0	0.0304	0.0009	0.72
Stanford bunny	34,984	1.04	-0.58	0	0.0328	0.0011	1.57

Table 8: Parameters of Pointshop3D for the scanned data

		Brent Minimization Options		
Examples	scale radius	Max Iteration	Tolerance	Search Range
Face	10.0	10	1e-4	1
Stanford bunny	5.0	10	1e-4	1

Table 9: Results on the sculpture and the bunny of Pauly's method

Examples	Z	μ	σ	μ_{mse}
Face	25.58	-0.00114	0.0414	0.0017
Stanford bunny	25.74	-0.00069	0.0454	0.0021

6.4 Influence of noise level and sample density

The influence of noise levels and sample densities over the resulting PSSs are examined below. Specifically, we examine how the resulting h and the deviations between the input point set P and the noise-filtered point set $\widetilde{P}$ vary with respect to noise levels and sample densities.

Table 10 shows how sample densities of the torus affect the resulting point sets. First, the noise level is kept the same ($\mu_0 = 0$, $\sigma_0 = 0.6$). As the table shows, denser points lead to smaller bandwidth, but the statistics of the resulting point sets including Z, μ, and σ are still the same and consistent with the added noise μ_0 and σ_0.

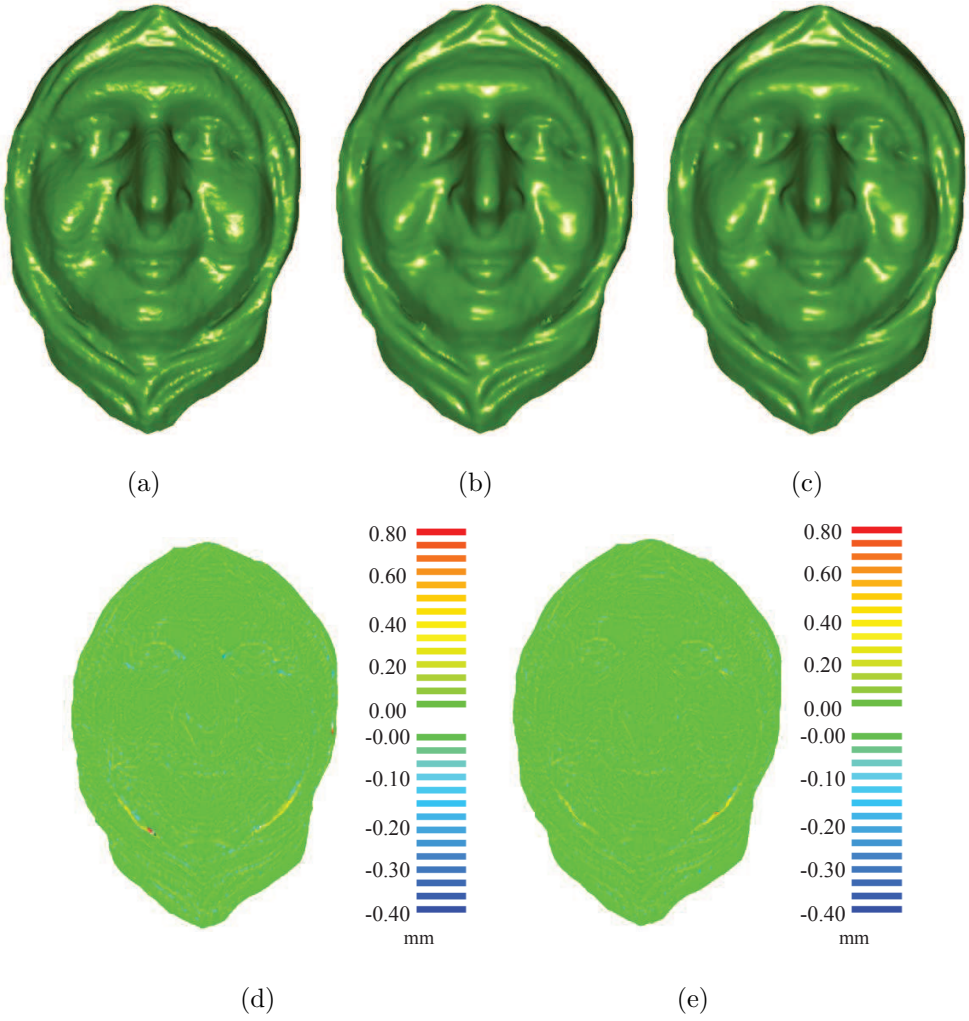


Figure 11: Results on a measured sculpture. (a) input data; (b) resultant PSS obtained by Pauly's method; (c) resultant PSS obtained by our method; (d) differences between (a) and (b); (e) differences between (a) and (c).

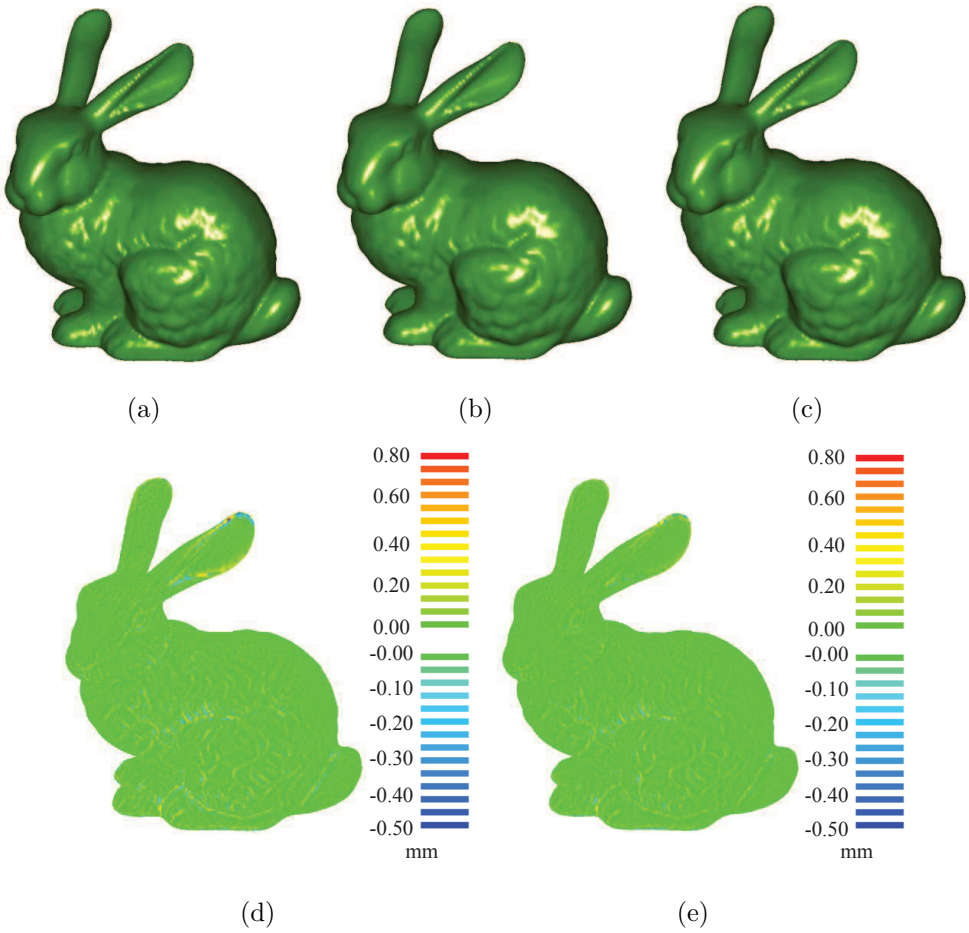
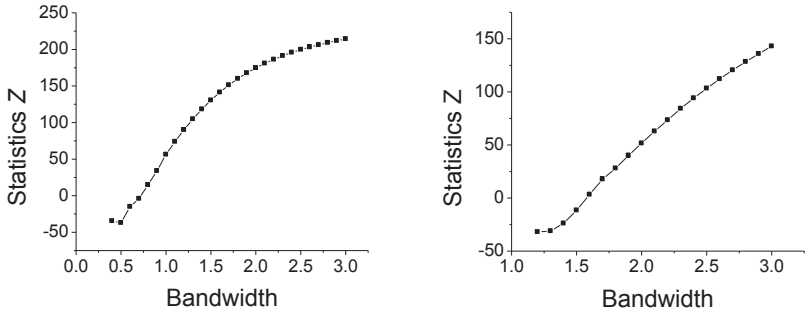


Figure 12: Results on the Stanford bunny. (a) input data; (b) resultant PSS obtained by Pauly's method; (c) resultant PSS obtained by our method; (d) differences between (a) and (b); (e) differences between (a) and (c).



(a) (b)

Figure 13: The statistics Z with respect to the bandwidth for: (a) the sculpture; (b) the Stanford bunny.

Table 10: Results of the torus with different sample densities

n	s_{dis}	Z	μ	σ	μ_{mse}	Resultant h
6,227	0.60	1.78	0	0.060	0.0036	3.17
10,988	0.45	1.82	0	0.060	0.0036	3.02
24,581	0.31	1.50	0	0.060	0.0036	2.90

Table 11 shows the influence of the various noise levels in the torus sampled with total $6,227$ points. As the table shows, as the noise level decreases, the resulting h decreases. However, the means and standard deviations , μ and σ, between the input and resulting point sets are consistent with the corresponding noise levels μ_0 and σ_0.

Experiments on more complex examples are also conducted. Fig 14 and Table 12 show results obtained from the Venus model when the input data has different sample densities and the same noise level ($\mu_0 = 0$, $\sigma_0 = 0.2$ of the noise). As the results show, higher sample density produces a little bit smaller bandwidth. Noises have been effectively removed in all cases in spite of different sample densities, i.e. $\mu \approx \mu_0$, $\sigma \approx \sigma_0$ and $|Z| \leq 2.33$. As Fig 14 shows, more detailed features can be captured in the resulting set when the input data is denser.

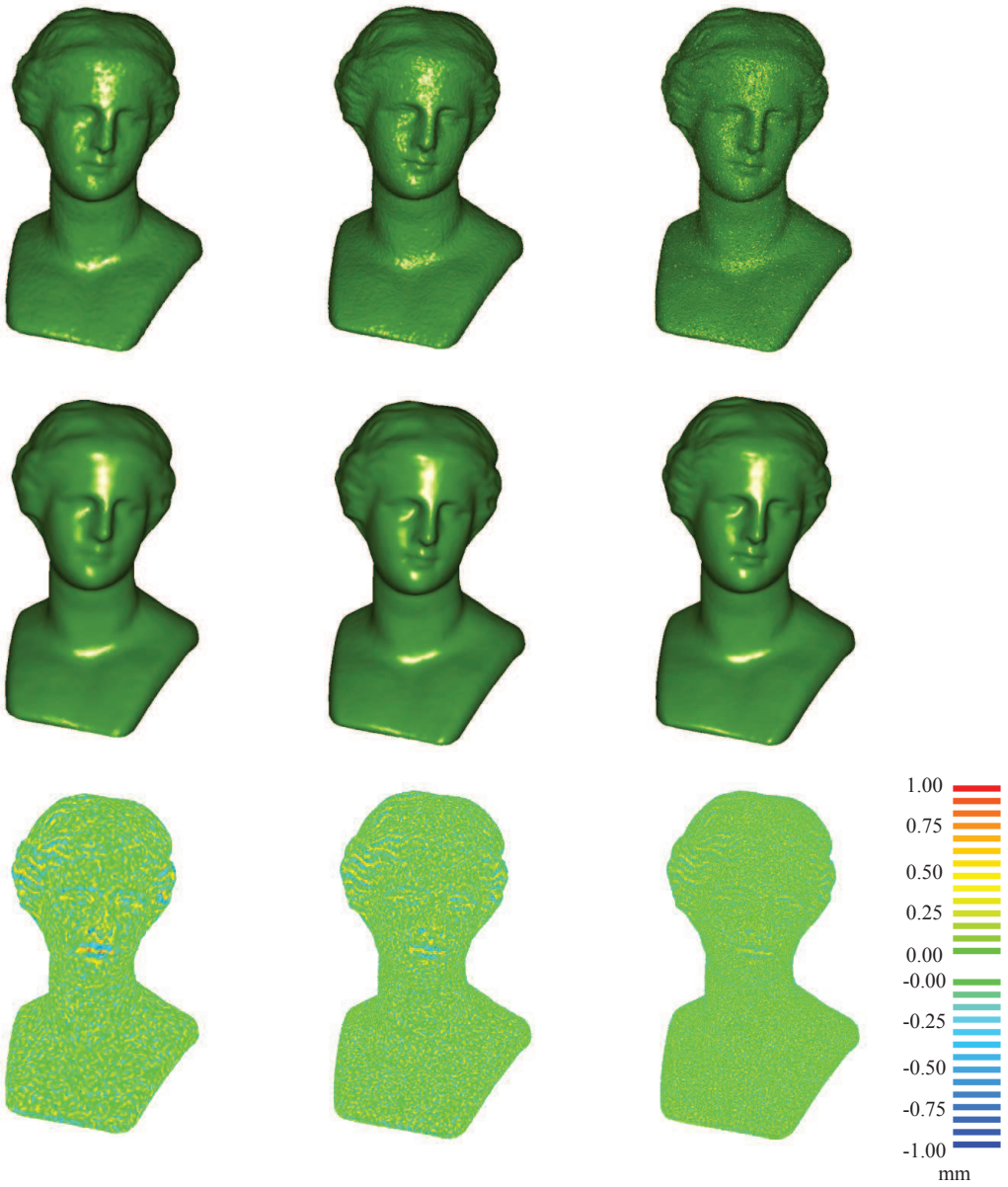


Figure 14: Results of the Venus model with different sample densities. Average point space: left: 1.74; middle: 1.13; right: 0.61. First row: input data; second row: resultant data; third row: differences between input data and resultant data.

Table 11: Results of the torus at different noise levels

μ_0 of noise	σ_0 of noise	Z	μ	σ	μ_{mse}	Resultant h
0	0.06	1.78	0	0.060	0.0036	3.17
0	0.04	0.48	0	0.040	0.0016	2.87
0	0.02	0.93	0	0.020	0.0004	2.59

Table 12: Results of the Venus when the input data has different sample densities

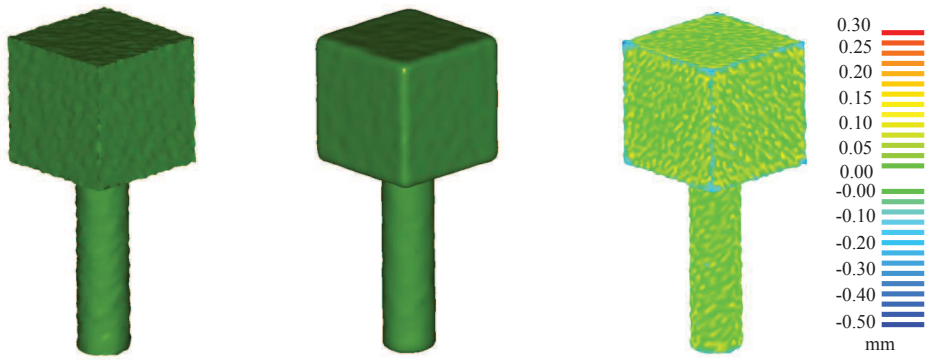
n	s_d	Z	μ	σ	σ_{mse}	Resultant h
23,076	1.74	0.10	0	0.210	0.0440	4.22
63,827	1.13	1.10	0	0.204	0.0415	3.43
255,176	0.61	1.15	0	0.199	0.0398	2.70

6.5 Experimental results on sharp features

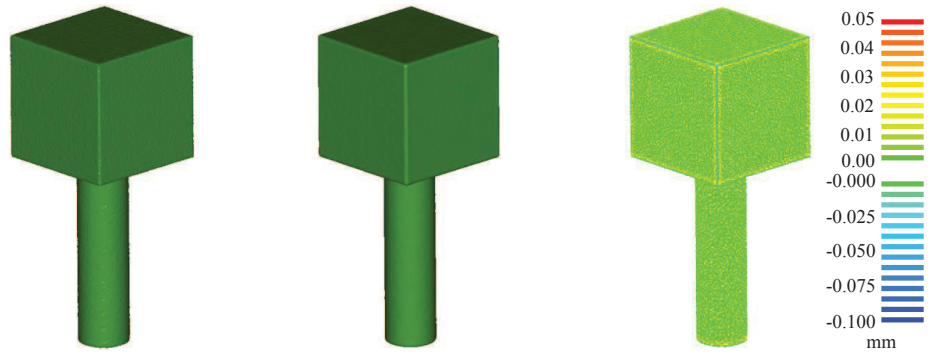
Our proposed method of controlling spatial variation of residuals and offsetting the bias has been found very effective on a variety of freeform shapes. We now extend this method to point cloud with sharp features.

Fig. 15 shows an example with sharp features, where the input data with different sample densities and noise levels are used. As the figure shows, sharp features are smoothed out when the points are not dense enough and the noise is high (Fig. 15.a). On the other hand, sharp features are preserved when the points are dense and the noise is low (Fig. 15.b).

Fig. 16 shows the results obtained from the Fandisk model, where two examples have different average point spaces and the same noise level. The model is obtained from [31]. Noises are added to the Fandisk model after upsampling. As Fig. 16 shows, sharp features are preserved when sample densities are increased. Fig 17 shows how the statistics Z vary with respect to the bandwidth for examples shown in Fig. 15.a and Fig. 16.a.. In both cases, the bandwidths can be automatically found by our method that ensure the resulting PSS surfaces with random error distribution, i.e. statistics $|Z|$

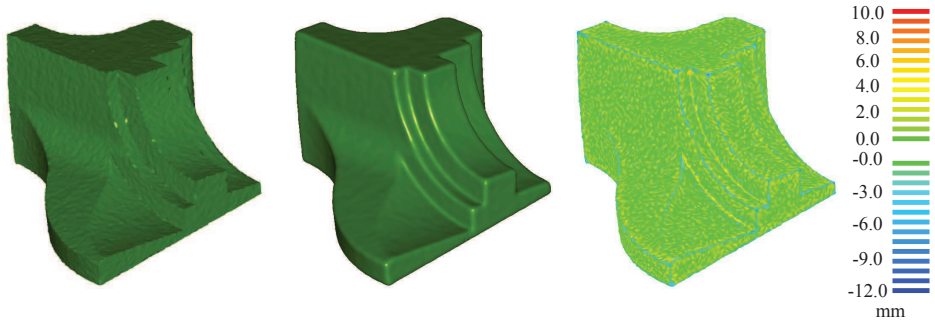


(a)

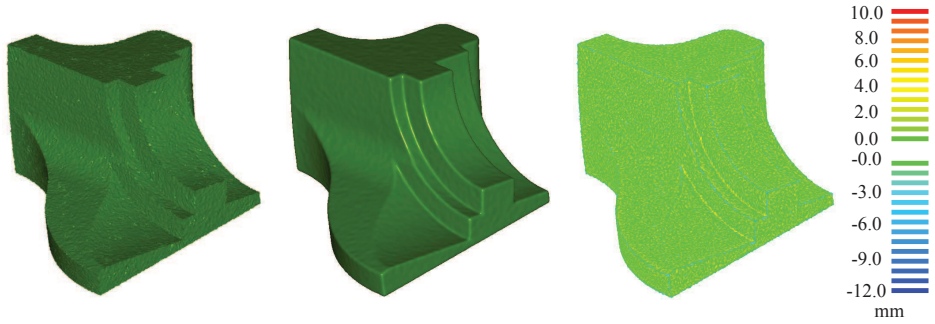


(b)

Figure 15: Results of the object consisting of box and cylinder. Left: input data; middle: approximated nominal surfaces; right: differences between input data and approximated nominal surfaces. Average point space: (a) 0.35, (b) 0.09; standard deviation of noise: (a) 0.05, (b) 0.01.



(a)

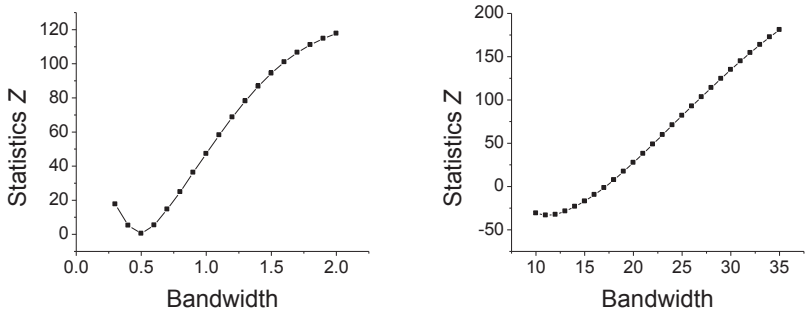


(b)

Figure 16: Results of the Fandisk model. Left: input data; middle: approximated nominal surfaces; right: differences between input data and approximated nominal surfaces. Average point space: (a) 8.56, (b) 4.39; standard deviation of noise is 1.0 for both (a) and (b).

smaller than 2.33 under the significance level of 0.02.

Therefore, to directly use our method in processing sharp features, denser data at the sharp corners is needed. In order to more effectively deal with sharp features, future work would look into how methods such as robust statistics [12], feature aware [32], and sparse reconstruction [33] can be applied in combination with our proposed method.



(a) (b)

Figure 17: The statistics Z with respect to the bandwidth for: (a) the object shown in Fig. 15.a; (b) the Fandisk model shown in Fig. 16.a.

7 CONCLUSION

This paper presents a novel method for defining a continuous point-set surface from input points through controlled spatial variation of residual errors and offset of systematic bias of the mean error. The random variation of residuals between input points and the resulting point sets is obtained through varying the bandwidth so that the statistics Z corresponding to Moran's I reaches the specified random level. Potential bias is corrected by offsetting the resulting points along the computed normal.

A set of computational experiments demonstrate the method is effective in removing random noise in the input data and leads to accurate reconstruction of point-set surfaces for freeform shapes. The resultant PSSs exhibit similar residual characteristics (randomness, mean, standard deviation) with respect to the input sampled points as the nominal surfaces do.

Future work would look into how the method can be extended to handle sharp features more effectively.

ACKNOWLEDGMENTS

We gratefully acknowledge the financial support from NSF grants (#0900597 and #1030347).

References

[1] Levin, D., 2003. "Mesh-independent surface interpolation". *Geometric Modeling for Scientific Visualization,* ***3***, pp. 37–49.

[2] Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., and Silva, C., 2003. "Computing and rendering point set surfaces". *IEEE Transactions on Visualization and Computer Graphics,* ***9***(1), pp. 3–15.

[3] Amenta, N., and Kil, Y., 2004. "Defining point-set surfaces". *ACM Transactions on Graphics,* ***23***(3), pp. 264–270.

[4] Yang, P., Schmidt, T., and Qian, X., 2009. "Direct digital design and manufacturing from massive point-cloud data". *Computer-Aided Design & Applications,* ***6***, pp. 685–699.

[5] Yang, P., and Qian, X., 2008. "Adaptive slicing of moving least squares surfaces: toward direct manufacturing of point set surfaces". *Journal of Computing and Information Science in Engineering,* ***8***, p. 031003.

[6] Zhang, D., Yang, P., and Qian, X., 2009. "Adaptive NC path generation from massive point data with bounded error". *Journal of Manufacturing Science and Engineering,* ***131***, p. 011001.

[7] Yang, P., and Qian, X., 2009. "Direct Boolean intersection between acquired and designed geometry". *Computer-Aided Design,* ***41***(2), pp. 81–94.

[8] Azariadis, P., and Sapidis, N., 2005. "Drawing curves onto a cloud of points for point-based modelling". *Computer-Aided Design,* ***37***(1), pp. 109–122.

[9] Liu, Y., Paul, J., Yong, J., Yu, P., Zhang, H., Sun, J., and Ramani, K., 2006. "Automatic least-squares projection of points onto point clouds with applications in reverse engineering". *Computer-Aided Design,* ***38***(12), pp. 1251–1263.

[10] Du, M., and Liu, Y., 2008. "An extension on robust directed projection of points onto point clouds". *Computer-Aided Design,* ***40***(5), pp. 537–553.

[11] Oztireli, C., Guennebaud, G., and Gross, M., 2009. "Feature preserving point set surfaces based on non-linear kernel regression". In Computer Graphics Forum, Vol. 28, Blackwell Publishing, pp. 493–501.

[12] Fleishman, S., Cohen-Or, D., and Silva, C., 2005. "Robust moving least-squares fitting with sharp features". In ACM SIGGRAPH 2005 Papers, ACM, pp. 544–552.

[13] Yan, Z., Yang, B., and Menq, C., 1999. "Uncertainty analysis and variation reduction of three dimensional coordinate metrology. Part 1: geometric error decomposition". *International Journal of Machine Tools and Manufacture,* ***39***(8), pp. 1199–1217.

[14] Yang, P., and Qian, X., 2007. "Direct computing of surface curvatures for point-set surfaces". In Eurographics symposium on point-based graphics, pp. 29–36.

[15] Pauly, M., 2003. "Point primitives for interactive modeling and processing of 3D geometry". PhD thesis, Diplom Informatiker University of Kaiserslautern.

[16] Dey, T., and Sun, J., 2005. "An adaptive MLS surface for reconstruction with guarantees". In Proceedings of the third Eurographics symposium on Geometry processing, Eurographics Association, pp. 43–52.

[17] Wang, H., Scheidegger, C., and Silva, C., 2008. "Optimal bandwidth selection for MLS surfaces". In Proceedings of Shape Modelling International, pp. 111–120.

[18] Guennebaud, G., and Gross, M., 2007. "Algebraic point set surfaces". *ACM Transactions on Graphics,* ***26***(3), pp. 23.1–23.9.

[19] Adamson, A., and Alexa, M., 2006. "Anisotropic point set surfaces". In Proceedings of the 4th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa, ACM, pp. 7–13.

[20] Alexa, M., and Adamson, A., 2007. "Interpolatory point set surfaces-convexity and hermite data". *ACM Transactions on Graphics,* ***28***(2), pp. 20:1–20:10.

[21] Kolluri, R., 2008. "Provably good moving least squares". *ACM Transactions on Algorithms (TALG),* ***4***(2), pp. 1–25.

[22] Dey, T., Goswami, S., and Sun, J., 2005. "Extremal surface based projections converge and reconstruct with isotopy". *Technical report OSU-CISRC-05-TR25, Ohio State University.*

[23] Fleishman, S., Drori, I., and Cohen-Or, D., 2003. "Bilateral mesh denoising". *ACM Transactions on Graphics (TOG),* ***22***(3), pp. 950–953.

[24] Shen, Y., and Barner, K., 2004. "Fuzzy vector median-based surface smoothing". *Visualization and Computer Graphics, IEEE Transactions on,* ***10***(3), pp. 252–265.

[25] Sun, X., Rosin, P., Martin, R., and Langbein, F., 2007. "Fast and effective feature-preserving mesh denoising". *IEEE Transactions on Visualization and Computer Graphics*, pp. 925–938.

[26] Moran, P., 1950. "Notes on continuous stochastic phenomena". *Biometrika,* ***37***(1-2), p. 17.

[27] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., and Stuetzle, W., 1992. "Surface reconstruction from unorganized points". *Computer Graphics-New York-Association for Computing Machinery,* ***26***, pp. 71–78.

[28] Pauly, M., Keiser, R., Kobbelt, L., and Gross, M., 2003. "Shape modeling with point-sampled geometry". *ACM Transactions on Graphics,* ***22***(3), pp. 641–650.

[29] Zwicker, M., Pauly, M., Knoll, O., and Gross, M., 2002. "Pointshop 3D: an interactive system for point-based surface editing". In Proceedings of the 29th annual conference on Computer graphics and interactive techniques, ACM, pp. 322–329.

[30] *http://www.graphics.stanford.edu/data/3Dscanrep/.*

[31] *http://www.ics.uci.edu/∼pablo/files/data/manifolds/.*

[32] Lipman, Y., Cohen-Or, D., and Levin, D., 2007. "Data-dependent MLS for faithful surface approximation". In Proceedings of the fifth Eurographics Symposium on Geometry Processing, Eurographics Association, pp. 59–67.

[33] Sharf, A., Greif, C., and Cohen-or, D., in press. "L1-sparse reconstruction of sharp point set surfaces". *ACM Transactions on Graphics.*

VITAE

Yu Liu is currently a research associate at the Illinois Institute of Technology (IIT). He is a post-doctoral researcher at the Huazhong University of Science and Technology. He received an undergraduate degree in mechatronic engineering from Tianjin University in 2000 and a doctor's degree in the mechatronic engineering of Huazhong University of Science and Technology in 2008. His research interests include reverse engineering, CAD/CAM, and digital manufacturing.

Xiaoping Qian is an associate professor in the Department of Mechanical, Materials and Aerospace Engineering at IIT. He received his Ph.D. in mechanical engineering from the University of Michigan, Ann Arbor in 2001. His research interests lie in the general area of computational design.