IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 23, 2020, accepted September 4, 2020, date of publication September 18, 2020, date of current version October 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3025227

Adaptive and Feature-Preserving Mesh Denoising Schemes Based on Developmental Guidance

NANNAN LI[1], (Member, IEEE), SHAOYANG YUE[1], ZHIYANG LI[1], SHENGFA WANG[2,3], AND HUI WANG[4]

[1]School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China
[2]DUT-RU International School of Information and Software Engineering, Dalian University of Technology, Dalian 116600, China
[3]Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology, Dalian 116600, China
[4]School of Information Science and Technology, Shijiazhuang Tiedao University, Shijiazhuang 500000, China

Corresponding author: Hui Wang (wangh@stdu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61802045 and Grant 61972267, in part by the China Postdoctoral Science Foundation under Grant 2020M670727, in part by the Open Project Program of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, under Grant VRLAB2020A04, in part by the Fundamental Research Funds for the Central Universities under Grant 3132020213, in part by the Natural Science Foundation of Hebei Province under Grant F2018210100, and in part by the Youth Talent Support Program of Universities of Hebei Province under Grant BJ2018003.

ABSTRACT Distinguishing among different kinds of features as well as noises on 3D mesh models is crucial for feature-preserving mesh denoising. This paper proposes to address this issue via in-depth analysis of the intermediate products of the denoising processes, and one framework is presented for raising adaptive and feature-preserving mesh denoising schemes. Firstly, by analyzing the changes of the facet normals during the denoising process, we propose the definition of developmental guidance, which helps to assess the current filtering status and predict the positions of feature and smooth regions. Then, by incorporating the guidance, we put forward one interpolation-based denoising scheme, which affords an efficient way to interpolate and recover different levels of features and is robust to severe noises. Besides, we also introduce the guidance to the optimization-based model, and the achieved global scheme is tested to be stable and robust to irregular samplings. Both the theoretical analysis and extensive experimental results on synthetic and real-world noises have demonstrated the attractive advantages of our whole framework, such as being adaptive, efficient, robust, feature-preserving, etc.

INDEX TERMS Bilateral filtering, feature-preserving, guided filter, linear interpolation, mesh denoising.

I. INTRODUCTION

Feature preserving 3D model denoising plays an essential role in preprocessing of point clouds [1], [2] or triangular meshes [3]–[10], which are obtained by scanning devices or via various kinds of digitalization processes. Previous methods can be roughly divided into four categories, namely, local methods, global methods, multi-step and deep learning based methods.

The local average schemes [11]–[15] are well-known to be fast. However, the operation of averaging can not avoid losing the original information, which usually obtains over-smoothed results or lost of weak features, especially when the number of iterations is large. Global optimization based methods, both isotropic [16], [17] and anisotropic [18]–[20], treat all the facet normals on feature and smooth regions equally. This may lead to the lost of details or weak features. Multi-step methods [21]–[23] usually add feature detection process prior to the filtering or the vertex updating process, which usually need heavy parameters tuning work and are time-consuming. Deep learning based methods [9], [24] have achieved success for certain dataset. However, the time-consuming training processes and the generalization ability of the networks still need to be researched.

Although lots of previous works have been carried out for mesh denoising, the research in this area is still active due to the existing challenges. From one hand, distinguishing the features from noises is not an easy work, especially when the noises are in high frequency. From the other hand, weak features are hard to be preserved or even recovered in the denoising processes. Faced with these challenges, the above conventional methods treating all the vertices or facets equally seem ineffective to some extent. Therefore,

The associate editor coordinating the review of this manuscript and approving it for publication was Sudipta Roy.

in this work, we propose a novel way to adaptively analyze the denoising process using the facet normals, which helps treat the feature and smooth regions differently and gives rise to one framework for putting forward adaptive denoising schemes.

In this paper, we propose a general adaptive and feature-preserving denoising framework based on a developmental guidance as shown in Fig. 1. Firstly, the given noisy model is over-smoothed to reflect the general structure information. Then the developmental guidance is introduced based on analyzing the changes of the facet normals of the original noisy model, over-smoothed one and the current denoised model in the denoising iteration. By introducing the developmental guidance into the linear interpolation of facet normals, we propose the adaptive and feature-preserving interpolation-based denoising scheme, which is fast and efficient. Furthermore, we incorporate the developmental guidance into a global optimization to adaptively penalize the noises. Our primary contributions can be summarized as:

- A framework is proposed for putting forward adaptive and feature-preserving denoising schemes for mesh models based on the newly defined developmental guidance.
- A linear interpolation based denoising scheme is proposed, which affords an efficient way to recover different levels of features and is robust to extremely noisy models.
- A global optimization based scheme is proposed, which is non-iterative and stable, and robust to irregular mesh sampling.

II. RELATED WORKS

The mesh filtering methods mainly include local average based methods, global optimization based methods, multi-step methods, and deep learning based methods.

A. LOCAL METHODS

Local methods include the classical ones for mesh filtering. Given that the facet normals can better represent the local surface geometric information than the vertex positions, researchers usually prefer to filter the facet normals and then use the filtered normals to update the positions of the vertices.

The local methods usually use kernel functions to weight the geometric differences between the concerned facet and its neighborhood to achieve the purpose of average-based denoising. Researchers usually consider using different geometric information of the facets or modifying the convolution kernel functions [15], [25] to improve the denoising performance. Fleishman *et al.* [26] and Zheng *et al.* [27] proposed the extensions of bilateral filtering on images [28]. Fleishman *et al.* [26] considered local differences in vertex positions and vertex normal differences to perform Gaussian convolutions. Zheng *et al.* [27] considered Gaussian convolutions based on centroid distances and facet normal differences, which provided more detailed processing for features compared to [26]. Unlike previous filtering methods, which consider facet normal information and vertex information, Centin *et al.* [14] updated normals based on model curvature information through local curvature changes and overall neighborhood curvature changes to maintain the features. Liu *et al.* [15] and Yadav *et al.* [25] changed the normal weights by changing the convolution kernel functions.

Since preserving the feature is always the core of denoising, a series of techniques have been proposed by researchers to guide the denoising process to protect or even recover the features. Zhang *et al.* [29] proposed to use the guidance signal to improve the bilateral filtering. Wang *et al.* [30] proposed the rolling guidance filtering on the basis of Zheng *et al.* [27]. Each step in the iteration process uses the original normal direction as a guide to better maintain the characteristics of the original model. Liu *et al.* [11] enhanced features by identifying feature and non-feature surfaces and proposing a measurement of neighborhood selection on the basis of guided filtering [29] and achieved better results than guided filtering. In the filtering process, the proper selection of the neighborhood patches can also afford ideal results. Li *et al.* [12] utilized the edge-based neighborhood to preserve sharp features. Hurtado *et al.* [13] selected the neighborhood by considering the gradient information in addition to the normal information. The local iterative method can gradually recover the features well, and the calculation efficiency is high. However, the number of iterations for these methods should be adjusted manually, and improper settings may lead to over-smoothing.

B. GLOBAL METHODS

Global algorithms for mesh filtering based on optimization models have also received much attention during these years. Zheng *et al.* [27] proposed an anisotropic denoising method based on bilateral weighted Laplace operator, however, the Laplace operator cannot well distinguish weak features from noises, and it also cannot handle models with large noises very well. He *et al.* [31] proposed a minimal mesh denoising algorithm based on edge operators and Zhang *et al.* [32] raised the TV normal filtering, which can, to some extent, solve the problems existed in [27], but they suffered undesired staircase effects in curved regions, especially for the method of [31]. Later on, Zhao *et al.* [18] optimized the non-convex problem and proposed an upgraded alternating strategy to solve the minimum problem. And Wang *et al.* [33] used the theory of compressed perception to recover features from the residuals of the Laplacian factorization. Recently, the second-order normalization has been well exploited in optimization-based algorithms. Liu *et al.* [7] combined the second-order normalization with the normal fidelity term. On the basis of [7], Zhong *et al.* [8] considered the overall variational differences and the fidelity of the normal, and better maintained the details of the model. The global method can better retain the overall structure of the concerned model, but it is not that good at maintaining local geometric details.

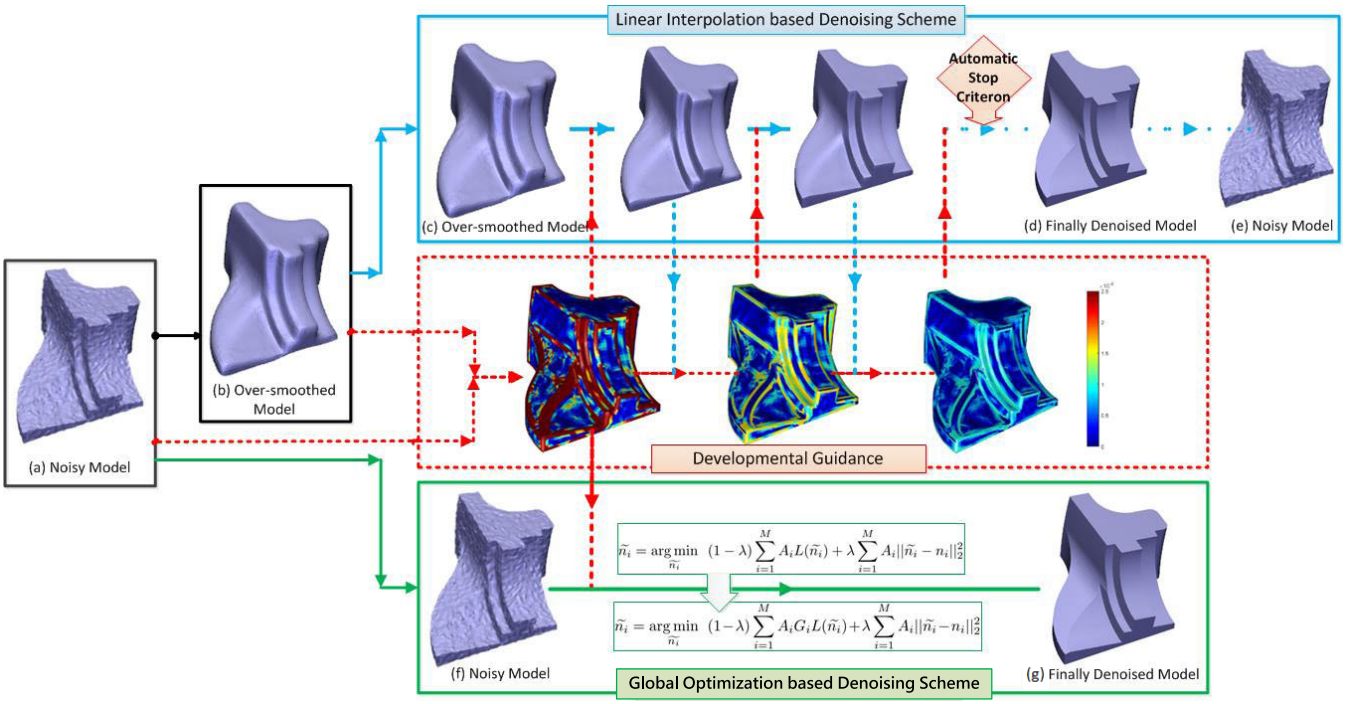


FIGURE 1. The illustration of our framework. The blue and green rectangles show the two proposed denoising schemes, and the red dashed rectangle shows the construction of the proposed developmental guidance.

C. MULTI-STEP APPROACH

To better preserve the features during the denoising process, many methods propose to detect the features firstly, thus giving rise to the multi-step approaches. Feature detection methods based on tensor voting are widely used to identify noises and features of the mesh. Wei *et al.* [34] processed the noises and features differently by voting on the types of vertices based on multi-scale normal tensor. Yadav *et al.* [35] performed a binary optimization of the proposed tensor equation by introducing the concept of a local binary neighborhood, which can better maintain the features. In order to improve the performance of the existed methods through multiple steps, a series of global filtering [23], [36], local filtering [37], median filtering [38], feature detection [21]–[23], and neighborhood searching [21] methods are introduced. The methods in [21]–[23] are typical multi-step methods, which add feature detection processes to the filtering process or the vertex updating process. Wei *et al.* [37] utilized the advantages of the local consistency of the vertex normal field and the surface normal field to make the two fields complement each other, thereby improving the denoising results. The method from Lu *et al.* [36] combines the global method with l_1-median filtering to remove noises. Since the multi-step methods are meticulous, they usually can achieve good results. However, the calculation complexity is usually large and there are a number of parameters need to be adjusted.

D. DEEP LEARNING METHODS

Deep learning has seen a booming in various research areas in recent years, and some successes have been achieved in the area of mesh filtering. Wang *et al.* [24] proposed a mesh filter by learning cascaded non-linear regression functions from a set of noisy meshes and their ground-truth counterparts. Compared with [24] with one-step learning technique, Wang *et al.* [9] learned one more step, that was from filtered models to their ground-truths for recovering the geometry lost during the first-step denoising procedure. The first step in [9] is to learn the mapping function from the noisy model set to its ground-truth counterpart set using neural networks for removing noises. And then the second step is to learn the reverse procedure of mesh filtering using the learned regression function sequences for the recovery of geometry. Deep learning methods have performed well, but the training processes are time-consuming. And the achieved well trained models are heavily affected by the data set, namely, features that are not included in the training data set cannot be recovered.

III. PREPARATION AND OVERVIEW

A. NOTATIONS AND RELATED CLASSICAL MESH DENOISING ALGORITHMS

Mesh denoising methods usually operate on the triangular meshes, which i-th facet is denoted as f_i, and its corresponding facet normal and centroid are represented as n_i and c_i respectively.

Zheng *et al.* [27] proposed one classical filtering algorithm based on facet normals, which extends the classical bilateral filtering from 2D images to 3D triangular meshes in a natural way as follows:

$$n_i^{t+1} = \frac{\sum_{j\in N(i)} \omega_c(||c_i - c_j||)\omega_s(||n_i^t - n_j^t||)\xi_{ij} n_j^t}{\sum_{j\in N(i)} \omega_c(||c_i - c_j||)\omega_s(||n_i^t - n_j^t||)\xi_{ij}}, \quad (1)$$

where, n_i^t denotes the i-th facet normal achieved by the t-th iteration, $N(i)$ represents the neighborhood of f_i, ξ_{ij} is a parameter related to the sampling rate, and ω_c and ω_s represent the Gaussian kernel functions for smoothing differences in centrods and normals. Neighbor facets which are closer in the centroids or are similar in facet normals contribute larger weights in the above filtering. And ω_c and ω_s are defined as:

$$\begin{cases} \omega_c(||c_i - c_j||) = exp(-\dfrac{||c_i - c_j||^2}{2\sigma_c^2}) \\ \omega_s(||n_i^t - n_j^t||) = exp(-\dfrac{||n_i^t - n_j^t||^2}{2\sigma_s^2}). \end{cases} \tag{2}$$

Here, σ_c and σ_s are the variance parameters of the corresponding Gaussian functions. σ_s is sensitive to the geometry of the mesh model, and only a proper settings of it can afford desirable feature-preserving denoising results. Therefore, various algorithms have been raised to pre-process the noisy models for the prediction of the feature locations. And the other way to preserve the feature information during denoising process is to add the information from the original noisy model back in an iterative way as shown in [30]. The core formula in [30] is:

$$n_i^{t+1} = \frac{\sum_{j\in N(i)} \omega_c(||c_i - c_j||)\omega_s(||n_i^t - n_j^t||)\xi_{ij} n_j}{\sum_{j\in N(i)} \omega_c(||c_i - c_j||)\omega_s(||n_i^t - n_j^t||)\xi_{ij}}. \tag{3}$$

It's clear that, the only difference between Eq. (1) and Eq. (3) is the filtering objects, which changes from n_j^t to n_j. This avoids the lost of original information from the noisy model, thus helping preserve the features.

B. OVERVIEW OF OUR FRAMEWORK

The main purpose of this paper is to provide one simple but efficient way for putting forward adaptive and feature-preserving mesh denoising schemes. Since there exists heavy computational cost in the preprocessing of other methods for feature detection, we propose to analyze and detect the potential features and smooth regions during the denoising process. And this idea can be introduced to various kinds of denoising methods, either new methods or conventional methods. We, in this paper, come up with two schemes, one scheme based on linear interpolation is a brand new way for denoising and other one is the modification of one existed optimization-based denoising method.

As stated in the related works, denoising methods usually rely on multiple steps or iterations, thus the facet normals are adjusted gradually. Take the bilateral filtering introduced above as an example, the facet normals in smooth regions are adjusted in the first several iterations and then tend to be unchanged. That is to say, if we measure the differences between the filtered facet normals and the original noisy facet normals, the differences in smooth regions are tend to be small, and those in the feature regions are relatively large. Therefore, we propose to utilize the observation for raising the developmental guidance for further adaptive denoising.

Then, we introduce the proposed developmental guidance to one brand new denoising scheme based on linear interpolation. It has been verified that the over-smoothed models contain the structure information of the original models. And the noisy models, though contaminated with noises, contain the complete feature and detail information (as stated in [30]). Therefore, we propose to exploit the information contained in the over-smoothed models and noisy models simultaneously through the simple but effective way of linear interpolation. And by setting the interpolation parameter adaptively based on the developmental guidance, the proposed scheme become adaptive to local geometry and feature-preserving. This also provides an efficient way to recover different levels of features for the concerned model. Then by proposing a stop criterion, the iterative scheme can obtain ideal denoising results.

In order to demonstrate the generalization ability of our framework, we also propose one global scheme via modifying one well-known optimization based method. The global filter is known to be non-interative and robust to sampling, however, since the facet normals are treated equally in the optimization process, the weak features are not preserved well. So we introduce the developmental guidance (constructed via one-step pre-filtering) to reweight facet-wise items in the global optimization model. The facet-wise reweighting operation takes much consideration of the local geometric shape, thus helping preserve the features adaptively.

IV. ADAPTIVE LOCAL SCHEME FOR DENOISING

A. LINEAR INTERPOLATION BASED SCHEME

Inspired by the guided normal filter, we propose to preserve the original feature information by incorporating the noisy model into the iterative filtering process. Since over-smoothed models are well-known to contain the low-frequency and structural information. Here, we set up our first scheme based on the linear interpolation between the noisy model and the over-smoothed model to preserve both the low-frequency (structure information) and high-frequency (detailed feature information) features.

For a given noisy model, the above idea can be expressed as:

$$n_i^{t+1} = (1 - \alpha)n_i^t + \alpha n_i, \tag{4}$$

where, n_i^t denotes the updated normal of the i-th facet in the t-th iteration ($t = 0, 1, 2, \ldots$), the α is a balance parameter and n_i^0 is the over-smoothed model. It can be seen that, we directly add the original facet normals back in a linear interpolation way iteratively. And the Eq. (4) can be rewrote as:

$$n_i^{t+1} = n_i^t + \alpha(n_i - n_i^t). \tag{5}$$

The second item in the above equation can be seen as the increment in the t-th iteration, and it is determined by the difference between n_i and n_i^t. We define this difference as the **developmental guidance**, which is denoted as $D_i^t = ||n_i - n_i^t||_2$ (as illustrated in the red dashed rectangle in Fig. 1). Fig. 2 shows the results (on the cube model) from the above

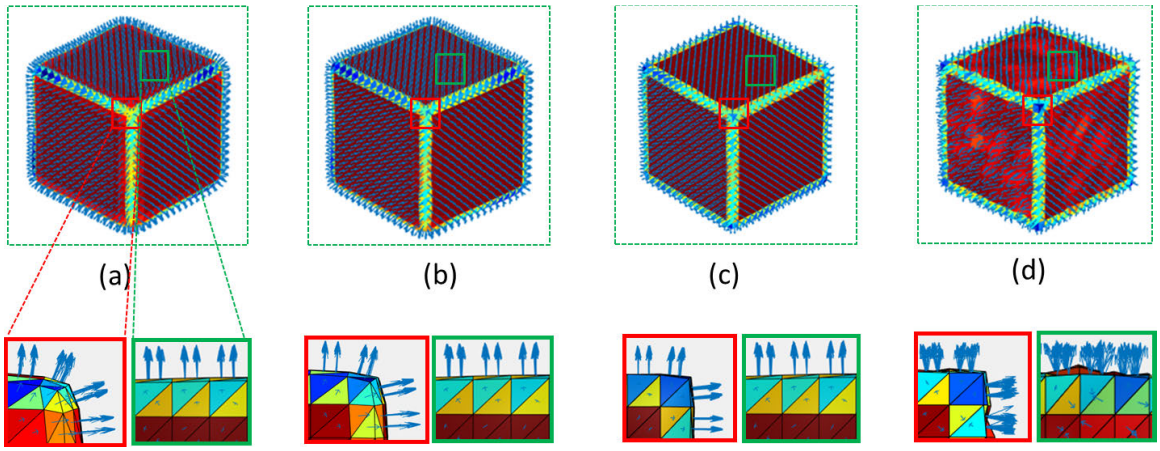


FIGURE 2. The variation of facet normals on the cube model in the process of linear interpolation (from (a) to (d)), with red and green rectangles zooming in to view the feature and smooth regions respectively.

linear interpolation process. It's clear that the features are recovered gradually (from Fig. 2(a) to Fig. 2(d)). During this process, the facet normals on the smooth regions (in green rectangles) change less compared with those on the feature regions (in red rectangles). That is to say, the developmental guidance D_i^t is relatively small in the smooth regions. Thus little information is added to the smooth region during the interpolation, namely, this required that $n_i^{t+1} \approx n_i^t$. To this end, we change the balance parameter to be a function $\phi(x)$ related to D_i^t, and then Eq. (5) is turned into:

$$n_i^{t+1} = n_i^t + \phi(D_i^t)(n_i - n_i^t). \tag{6}$$

Then, from the above analysis, the nonlinear function $\phi(x)$ should satisfy: $\lim_{|x|\to 0}\phi(x) = 0$. On the other hand, D_i^t should be relatively large on the feature regions due to the smoothing effect. That is to say, the features should be added back to the over-smoothed model. Although the noisy model contains noises, it also holds the complete feature information. Therefore, the feature regions should be similar to those on the noisy model, namely, for the facet on the feature regions, it requires: $n_i^t \approx n_i$ and $\lim_{|x|\to+\infty}\phi(x) = 1$. With all of the above considerations, we finally define $\phi(x)$ as:

$$\phi(x) = 1 - exp(-\frac{x^2}{2\sigma^2}). \tag{7}$$

σ is a scale parameter to control the speed for ϕ to increase to 1 and decrease to 0, and it can be empirically set in the interval of (0.1, 3). It can be seen that, if $|D_i^t| \geq 3\sigma$, $\phi(D_i^t) \geq 0.99$, which guarantees the restoration of feature information from the noisy model. From Eq. (6) and Eq. (7), it can be seen that our local scheme is actually an adaptive interpolation process, which judges the i-th facet belongs to feature regions or smooth regions according to the developmental guidance D_i^t and adaptively interpolates from the over-smoothed model and noisy model based on $\phi(D_i^t)$.

B. THEORETICAL ANALYSIS ON THE INTERPOLATION BASED SCHEME

We, in this subsection, will conduct further theoretical analysis on the interpolation based scheme. From Eq. (6) and the fact that n_i^0 is the facet normal from the over-smoothed model, we can use mathematical induction to prove that:

$$n_i^{t+1} = \Phi^t n_i^0 + (1 - \Phi^t)n_i, \tag{8}$$

where

$$\Phi^t = \prod_{k=1}^{t}(1 - \phi(D_i^k)) = exp(-\frac{1}{2\sigma^2}\sum_{k=1}^{t}(D_i^k)^2). \tag{9}$$

And this equation can better reflect our interpolation-based denoising idea. That is, the denoised model can be seen as the interpolated results from the over-smoothed model and the noisy model.

Then, we can easily prove that our iterative local scheme converges to the original noisy model. Since $\sum_{k=1}^{t}(D_i^k)^2 \geq \sum_{k=1}^{t-1}(D_i^k)^2$, we have

$$\begin{cases} \Phi^t < \Phi^{t-1} & if\ n_i^t \neq n_i \\ \Phi^t = \Phi^{t-1} & if\ n_i^t = n_i. \end{cases} \tag{10}$$

And if we define $\epsilon^t = |n_i^{t+1} - n_i| = \Phi^t|n_i^0 - n_i|$, we have $0 \leq \epsilon^t < \epsilon^{t-1}$. Therefore, ϵ^t is a decreasing function of t, ensuring that the output of Eq. (8) is getting closer to the noisy model as the iterations proceed. From Fig. 3, it can be seen that, the interpolation outputs are added with more details iteratively (from Fig. 3(a) to Fig. 3(d)), namely different levels of features are recovered. And this also illustrates the convergence of the interpolation algorithm. Fig. 4(a) demonstrates the convergence measured by the mean squared difference (on the fandisk model with 0.2 Gaussian noises) defined as:

$$\eta^t = \frac{1}{M}\sum_{i=1}^{M}|n_i^t - n_i|^2, \tag{11}$$

where M is the number of the facets on the mesh model.

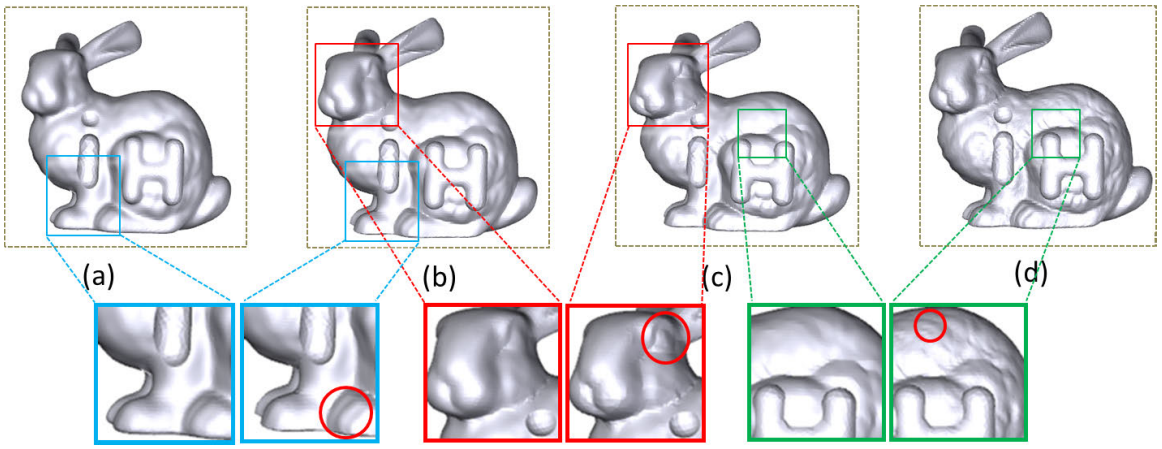


FIGURE 3. The intermediate outputs of our iterative interpolation based denoising scheme, with (a)-(d) denoting the bunny models with more and more details (namely, different levels of features are recovered).

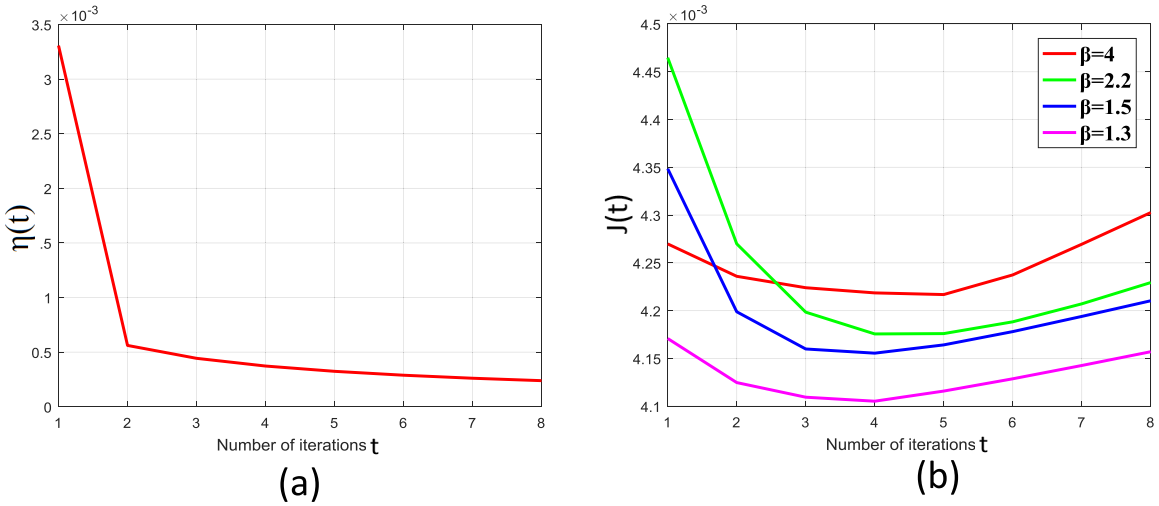


FIGURE 4. (a) shows the convergence of our iterative scheme measured by the mean squared difference, and (b) shows the variation of the proposed objective function for selecting the number of iterations.

TABLE 1. Run time for different methods.

Model	(vertices,faces)	Timing (s)					
		Zheng [27] (Local)	Zhang [29]	Ours (LI)	Zheng [27] (Global)	He [31]	Ours (GO)
Fandisk	(6475,12946)	0.578	0.836	0.315	1.235	3.330	1.319
Max	(30942,61880)	1.599	3.491	1.061	10.321	19.846	10.852
Bunny	(34834,69451)	2.904	4.194	1.469	11.452	22.653	11.995

C. STOPPING CRITERION

To avoid the manual adjustment of the number of iterations, we propose to determine the iteration number based on one stopping criterion defined by the minimization of the following objective function:

$$J(t) = \sum_{i=1}^{M} D_i^t + \beta \sum_{i=1}^{M} L(n_i^t), \tag{12}$$

where, $L(n_i^t) = ||n_i^t - K_i \sum_{j\in N(i)} \omega_{ij} n_j^t||^2$ and $K_i = 1/\sum_{j\in N(i)} \omega_{ij}$, ω_{ij} can be directly defined as the same with the bilateral weights proposed in [27] (the settings of which will be detailed in the section of experiments and discussions), and β is the balance parameter between the first item (sum of differences) and the second item (smooth measurement). This objective function is tested to be effective in measuring the denoising results. As can be seen from Fig. 4(b), with different settings of β, 4 or 5 iterations can afford the minimal values of $J(t)$, and the denoising results

Algorithm 1 Linear Interpolation Based Denoising Algorithm

Require: Noisy model.
Ensure: Denoised model.
1: Compute the facet normals (denoted as $\{n_i\}$) of the noisy model.
2: Achieve the over-smoothed model (denoted as n_i^0) with any kind of smoothing/denoising algorithm.
3: **for** $t = 0 \rightarrow t^S$ **do**
(t^S is the iteration number set according to the stopping criterion)
4: Compute developmental guidance $D_i^t = ||n_i - n_i^t||^2$;
5: Compute the interpolation parameter ϕ according to Eq. (7);
6: Update the facet normals based on adaptive linear interpolation according to Eq. (6);
7: **end for**
8: Update the positions based on the final n_i^t.

are also visually ideal. The overall algorithm is detailed in Algorithm 1.

V. ADAPTIVE GLOBAL SCHEME FOR MESH DENOISING

A. GLOBAL SCHEME BASED ON ADAPTIVE OPTIMIZATION

Global algorithms are well-known to be more stable compared with local schemes due to the non-iterative nature. Zheng *et al.* [27] proposed the global method based on the following global optimization model:

$$\widetilde{n_i} = \arg\min_{\widetilde{n_i}} \ (1-\lambda)\sum_{i=1}^{M} A_i L(\widetilde{n_i}) + \lambda \sum_{i=1}^{M} A_i ||\widetilde{n_i} - n_i||_2^2, \tag{13}$$

where A_i denotes the area of the i-th facet. The minimization of the above objective function is well-known to be effective in smoothing the geometric details, namely, the high-frequency information including the noises together with the weak features. That is to say, it smooths the model surfaces without specific consideration of the features and noises, thus leading to the loss of the features to some level. Although Zheng *et al.* [27] provided the anisotropic setting of L, the strength for smoothing all the facets on the whole model is the same. However, the facet normals on the smooth regions are expected to be smoothed with more strength, while those on the feature regions should be smoothed with less strength. Then, under the action of global optimization, the noises can be removed while the features can be preserved.

To this end, we propose our global scheme as follows:

$$\widetilde{n_i} = \arg\min_{\widetilde{n_i}} \ (1-\lambda)\sum_{i=1}^{M} A_i G_i L(\widetilde{n_i}) + \lambda \sum_{i=1}^{M} A_i ||\widetilde{n_i} - n_i||_2^2, \tag{14}$$

where λ can be empirically set in the interval of $(0.05, 0.1)$, and G_i is defined based on the developmental guidance (as stated in the above section) as:

$$G_i = exp(\frac{D_i}{2\sigma_g^2}), \tag{15}$$

where, σ_g can be empirically set according to the scale of D_i (e.g. $0.5 * mean(D_i)$). The developmental guidance here is not computed in any iterative process, it is inferred

TABLE 2. Comparisons with other methods by setting proper parameters for all the methods.

Models	Methods and Parameters	MSAE
Fandisk	Fleishman [26] (10)	0.121111
	Jones [39] (1.3,1.4)	0.090312
	Sun [20] (0.55,20,50)	0.038846
	Zheng [27] local (0.3,5,10)	0.024786
	Zheng [27] global (0.3,0.01,10)	0.025894
	He [31] (0.01,0.00346,0.001,0.5,1.414,1000)	0.094055
	Zhang [29] (2,0.25,25,20)	0.028018
	Yadav [25] (0.5,0.2,100)	0.051549
	Our LI (0.5,10,10,0.2)	0.017893
	Our GO (0.01,0.35,1.2,10,0.2)	0.021406
Octaflower	Fleishman [26] (6)	0.134672
	Jones [39] (1.8,1.5)	0.117833
	Sun [20] (0.5,6,20)	0.067908
	Zheng [27] local (0.25,6,20)	0.048115
	Zheng [27] global (0.1,0.33,20)	0.052757
	He [31] (0.01,0.00149,0.001,0.5,1.414,1000)	0.311125
	Zhang [29] (2,0.35,6,20)	0.073122
	Yadav [25] (0.55,0.2,100)	0.231323
	Our LI (3,5,20,0.2)	0.063418
	Our GO (0.01,0.34,1.2,10,0.2)	0.044083
Skull	Fleishman [26] (12)	0.120945
	Jones [39] (1.2,1.6)	0.101446
	Sun [20] (0.5,6,20)	0.102791
	Zheng [27] Zheng local (0.5,6,10)	0.140329
	Zheng [27] Zheng global (0.5,0.01,10)	0.144918
	He [31] (0.000004,1,0.001,0.9,1.414,1000)	0.125912
	Zhang [29] (2,0.5,6,15)	0.093849
	Yadav [25] (0.35,0.2,100)	0.124678
	Our LI (1,4,10,0.2)	0.103291
	Our GO (0.01,0.5,1,20,0.2)	0.102310
iH-bunny	Fleishman [26] (5)	0.170718
	Jones [39] (1.2,1.5)	0.182662
	Sun [20] (0.5,5,20)	0.178188
	Zheng [27] Zheng local (0.5,4,20)	0.092096
	Zheng [27] Zheng global (0.5,0.01,20)	0.097233
	He [31] (0.000004,1,0.001,0.5,1.414,1000)	0.180329
	Zhang [29] (2.7,0.55,4,15)	0.197061
	Yadav [25] (0.55,0.2,100)	0.203320
	Our LI (1.5,6,6,0.2)	0.085990
	Our GO (0.01,0.5,1.5,20,0.2)	0.094225
Max	Fleishman [26] (20)	0.110322
	Jones [39] (1.3,1.4)	0.110420
	Sun [20] (0.35,5,20)	0.091067
	Zheng [27] Zheng local (0.5,4,20)	0.083739
	Zheng [27] Zheng global (0.01,0.5,20)	0.088749
	He [31] (0.002,0.00217874,0.001,0.5,1.414,1000)	0.107232
	Zhang [29] (2,0.4,8,20)	0.093173
	Yadav [25] (0.35,0.2,100)	0.108761
	Our LI (2,4,8,0.2)	0.067870
	Our GO (0.01,0.5,0.8,20,0.2)	0.083742

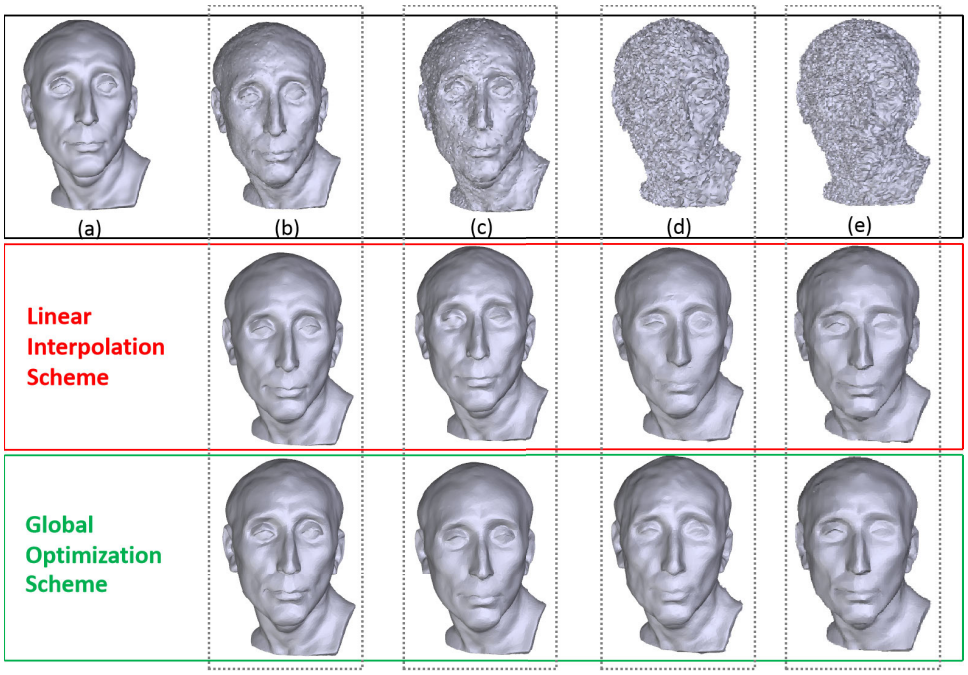


FIGURE 5. Demonstration of the robustness of our schemes to different categories and levels of noises on the Nicolo model. (a) shows the original Nicolo model without noise, (b) and (c) show the models added with 0.3 and 0.5 Gaussian noises, (d) shows the model added with 50% impulse noises with scale of 0.6 mean edge length, and (e) shows the model added with mixed Gaussian-impulse noises (0.3 Gaussian and 50% 0.3 impulse noises).

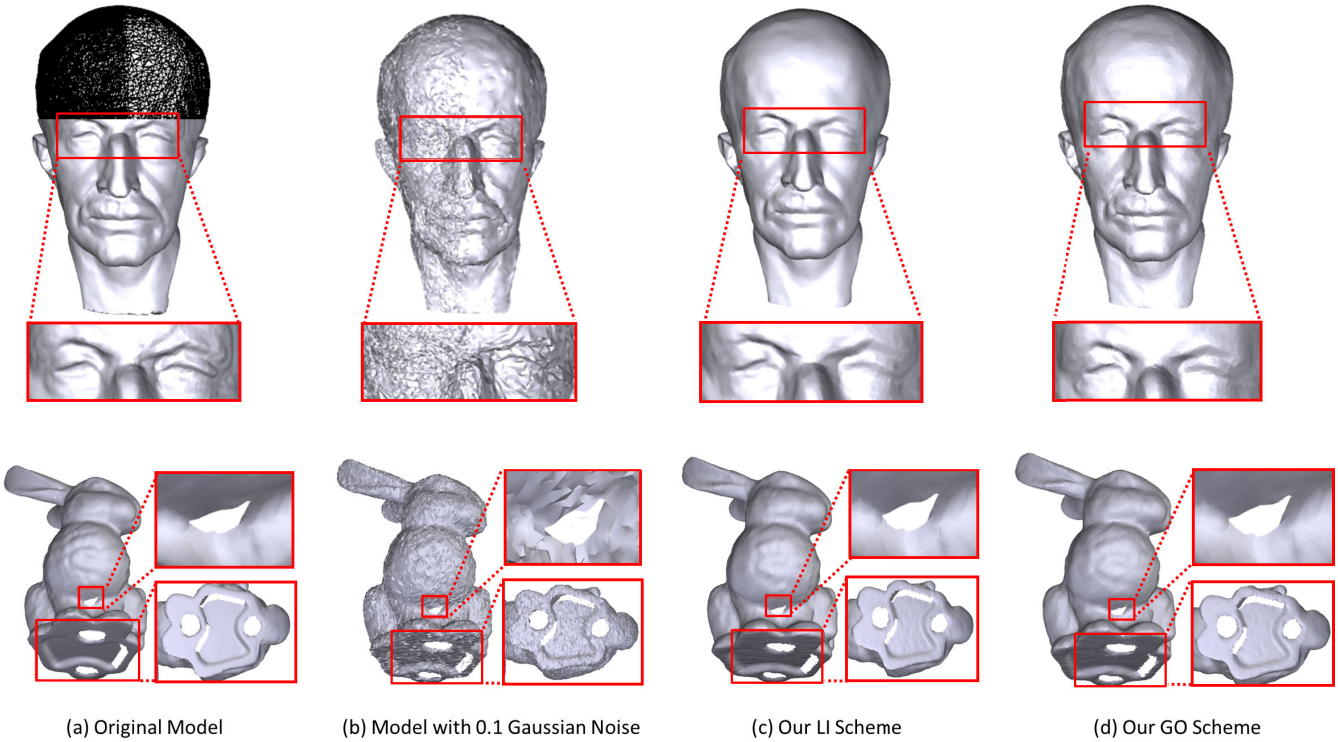


FIGURE 6. Demonstration of the robustness to irregular sampling and defects, with the upper and lower rows showing the models with irregular samplings and defects respectively.

from the one-step over-smoothing as illustrated in the red dashed rectangle in Fig. 1. Given the facet normals of the over-smoothed model n_i^0, then $D_i = n_i - n_i^0$. For original model with large noises, the resulted D_i may be contaminated with noises to some level, we find that a simple step of Gaussian filter on D_i can afford ideal results. It is easy to

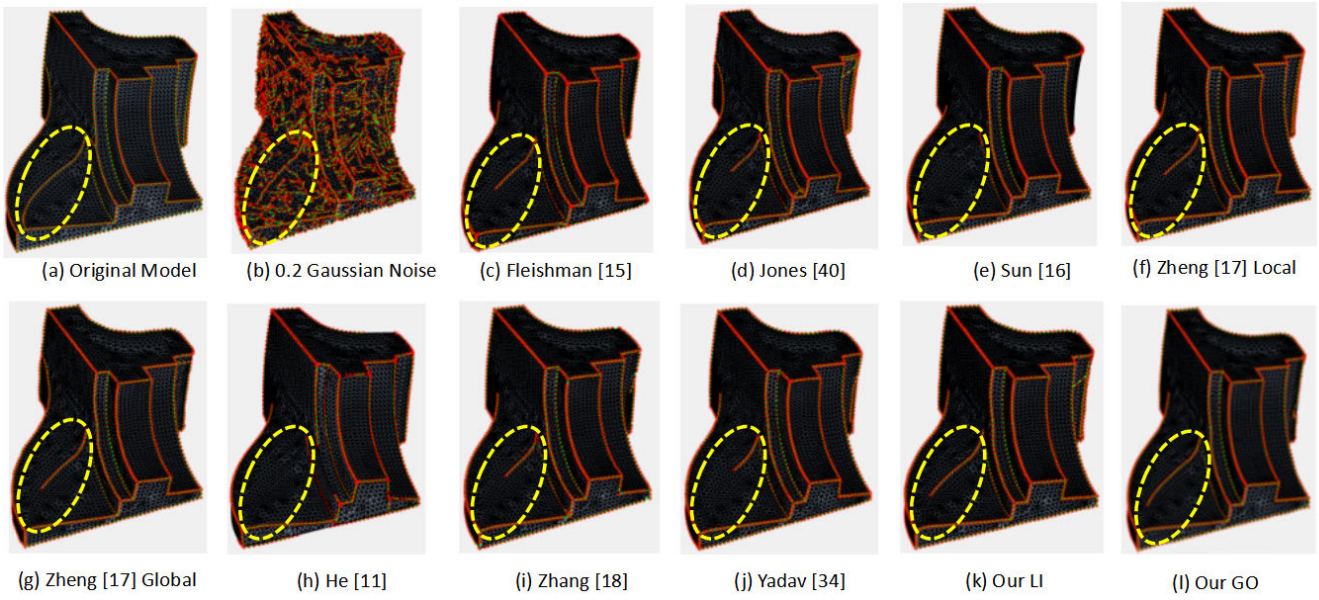


FIGURE 7. Comparisons on feature detection based on the denoised models (achieved by different methods), with yellow dashed ellipses indicating the positions of weak features.

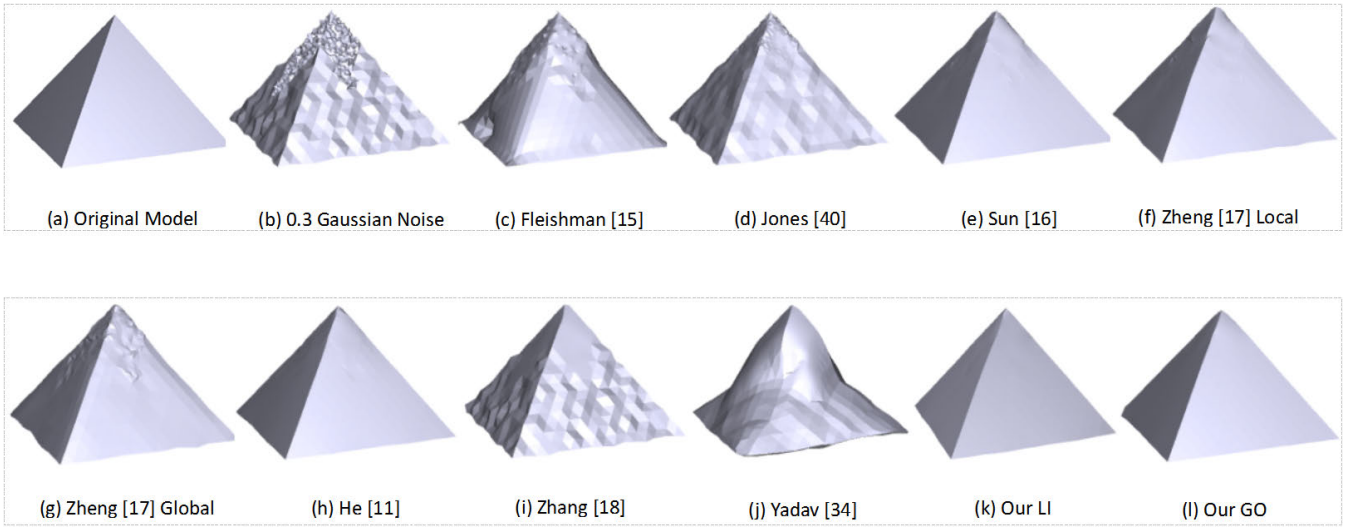


FIGURE 8. Visual comparisons on the synthetic pyramid model.

see that, our global scheme actually reweights the global optimization model based on the developmental guidance. This reweighting operation helps penalize the smooth and feature regions adaptively, enabling the feature-preserving denoising.

B. UPDATES OF VERTEX POSITIONS

With the facet normals filtered, we adopt the method introduced in Yadav [25] to update the vertex positions. There are two main parameters in this method, namely, the number of vertex updating and the isotropic factor, which will be discussed in the experiment and discussion section.

VI. EXPERIMENTS AND DISCUSSIONS

In this section, we demonstrate the performance of our approach via experiments in various aspects. All the experiments were conducted on a 3.5GHz Intel(R) Core(TM) i7 computer with 16GB memory. Since the iterative scheme of our work is based on the simple linear interpolation, and the global scheme is not iterative, both of them are with low time cost. For all the experiments in our work, we utilize the method in [26] to generate the over-smoothed model. More timing details concerning versatile models for our linear interpolation based scheme (denoted as LI scheme) and global optimization based scheme (denoted as GO scheme) are shown in Table 1. For the methods concerned in this

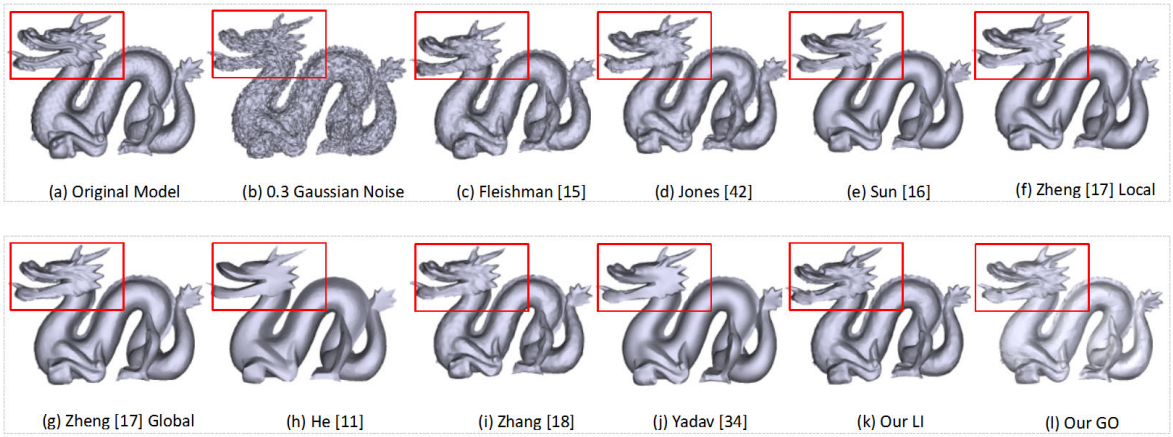


FIGURE 9. Visual comparisons on the synthetic dragon model.

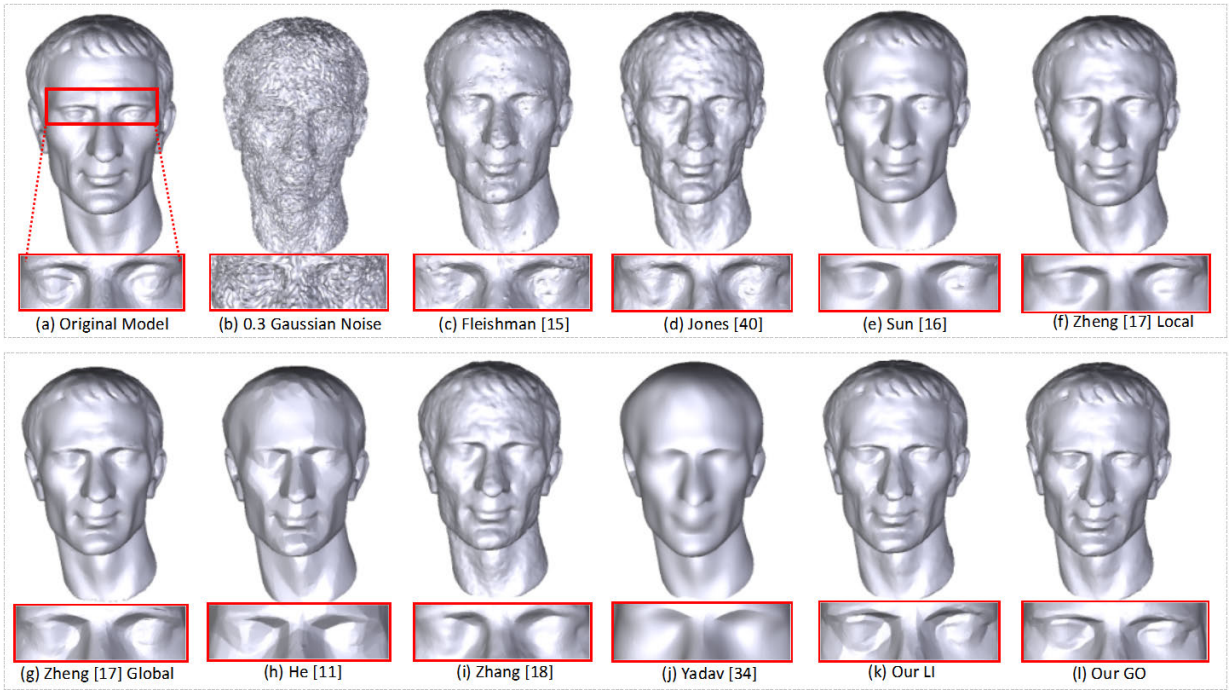


FIGURE 10. Visual comparisons on the synthetic Julius model.

table, we implemented Zheng [27] (Local and global) in Matlab as our methods and implemented Zhang *et al.* [29] and He *et al.* [31] in C++. We adjusted the parameters as shown in Table 2 and achieved the ideal results with the same running time as reported in the corresponding papers.

A. PROPERTIES OF OUR SCHEMES

To show the performances and properties of our schemes, we conduct experiments on a number of models in various complex cases.

1) ROBUSTNESS TO DIFFERENT CATEGORIES AND LEVELS OF NOISES

Here, we add three kinds of synthetic noises to the 3D models, which mainly include the zero-mean Gaussian noises, impluse and mixed Gaussian-impulse noises. We add 0.3 and 0.5 Gaussian noises (with standard deviation proportional to the average edge length of the original model) to the Nicolo model and achieve the noisy model as shown in Fig. 5(b) and Fig. 5(c). Then we add 50% impulse noises with scale of 0.6 mean edge length to the Nicolo model as shown in Fig. 5(d), and add mixed Gaussian-impulse noises (0.3 Gaussian and 50% 0.3 impulse noises) to the Nicolo model as shown in Fig. 5(e). And the denoised results in the second and third rows demonstrate the robustness of our shemes. It can be seen that, both of our schemes can achieve ideal feature-preserving denoising results even when the models are contaminated with severe noises that are visually unclear. And the linear interpolation based scheme performs better in dealing with severe noises due to the iterative and

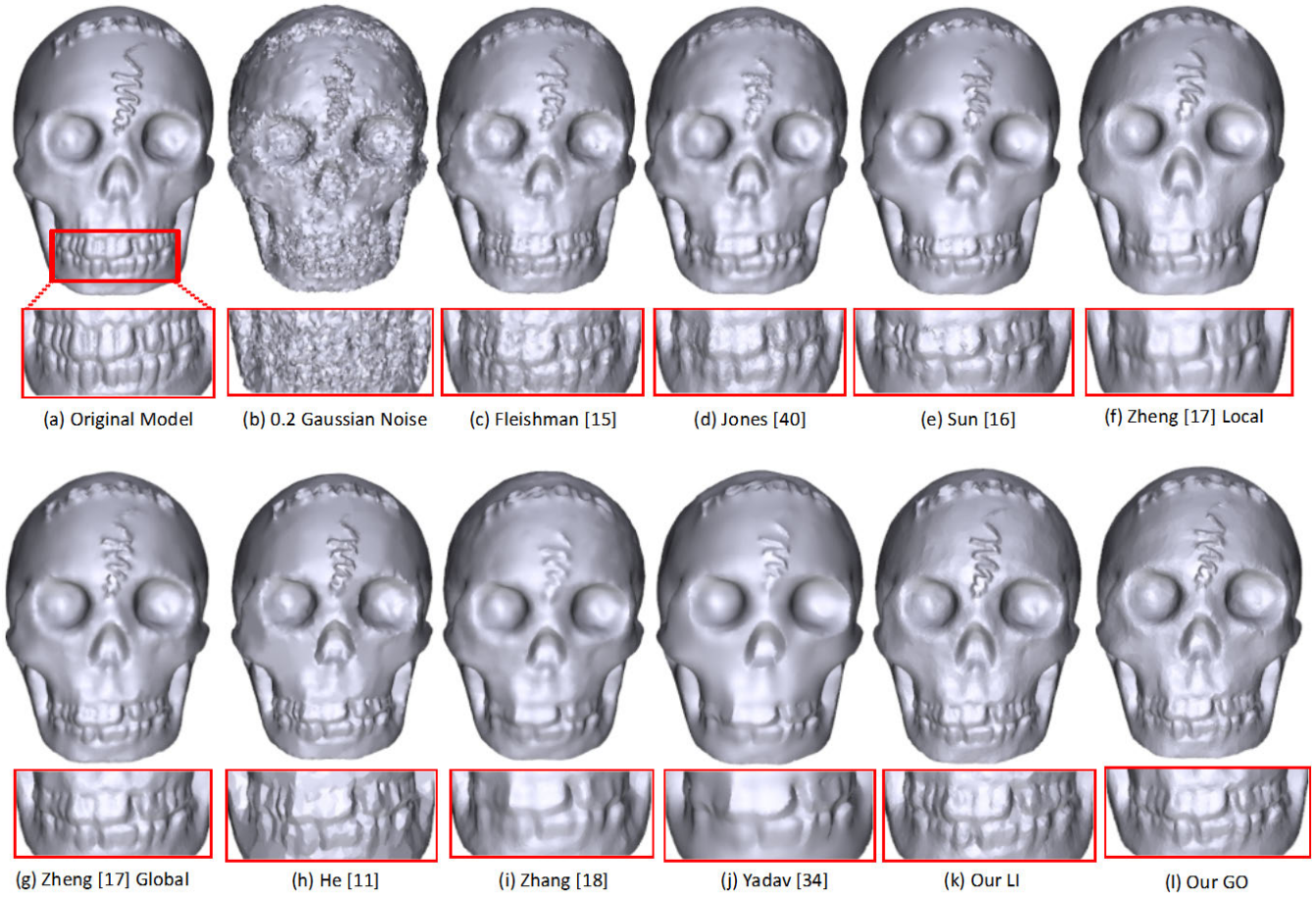


FIGURE 11. Visual comparisons on the synthetic skull model.

adaptive interpolation process for preserving and recovering the geometric features.

2) ROBUSTNESS IRREGULAR SAMPLING

Our global scheme is based on the global optimization with sampling rate parameter incorporated in the optimization model, therefore it is robust to the irregular sampling. As shown in the upper row in Fig. 6, the left and right parts of the Max model is with different sampling rate. The Nicolo model shown in the Fig. 5 is also with irregular sampling rate. From the denoising results in these two figures, it can be seen that both of our schemes can well deal with such complex cases and obtain the visually correct results. Our global scheme can better preserve the geometric features in case of irregular sampling and obtain uniform denoising results.

3) ROBUSTNESS TO DEFECTS

The raw models obtained with scanning devices are usually contaminated with different kinds of defects. Holes are among the most common defects. So we specifically dig holes on the bunny model as shown in the lower row in Fig. 6. It's clear that both of our schemes can obtain ideal denoising results with the boundaries preserved smooth and complete.

B. QUALITATIVE AND QUANTITATIVE COMPARISONS

In this section, we will make vast comparisons with the classical methods and the state-of-the-arts. The contrastive methods and their parameters are as follows: the bilateral filtering proposed by Fleishman *et al.* [26] (n_1), the robust statistics proposed by Jones *et al.* [39] (σ_{r1}, σ_g), the unilateral filtering algorithm proposed by sun *et al.* [20] (T, n_1, n_2) the normal filter proposed by Zhang *et al.* [29] ($r, \sigma_r, k_{iter}, V_{iter}$), the L_0 minimization proposed by He *et al.* [31] ($\lambda_1, \alpha_0, \beta_0, u_\alpha, \beta_{max}$), the global ($\lambda_2$, σ_{s1},n_1) and local (σ_{s2},n_1,n_2) methods proposed by Zheng *et al.* [27], and robust feature-preserving denoising proposed by Yadav *et al.* [25] (σ_s, σ_c, V_{iter}). The LI and GO methods proposed in this paper contain four parameters(σ, n_1, n_2, λ_I) and five parameters(λ, σ_s, σ_g, n_2, λ_I) respectively. Among them, n_2, λ_1 are the shared parameters in the methods concerned. n_1 is the number of iterations for normal updating, n_2 is the number of iterations for vertice position updating, λ_I is the isotropic smoothing or the mesh fidelity factor, which can be empirically set to the range of [0.2,0.5]. λ is the balance parameter for our global filter. σ_s

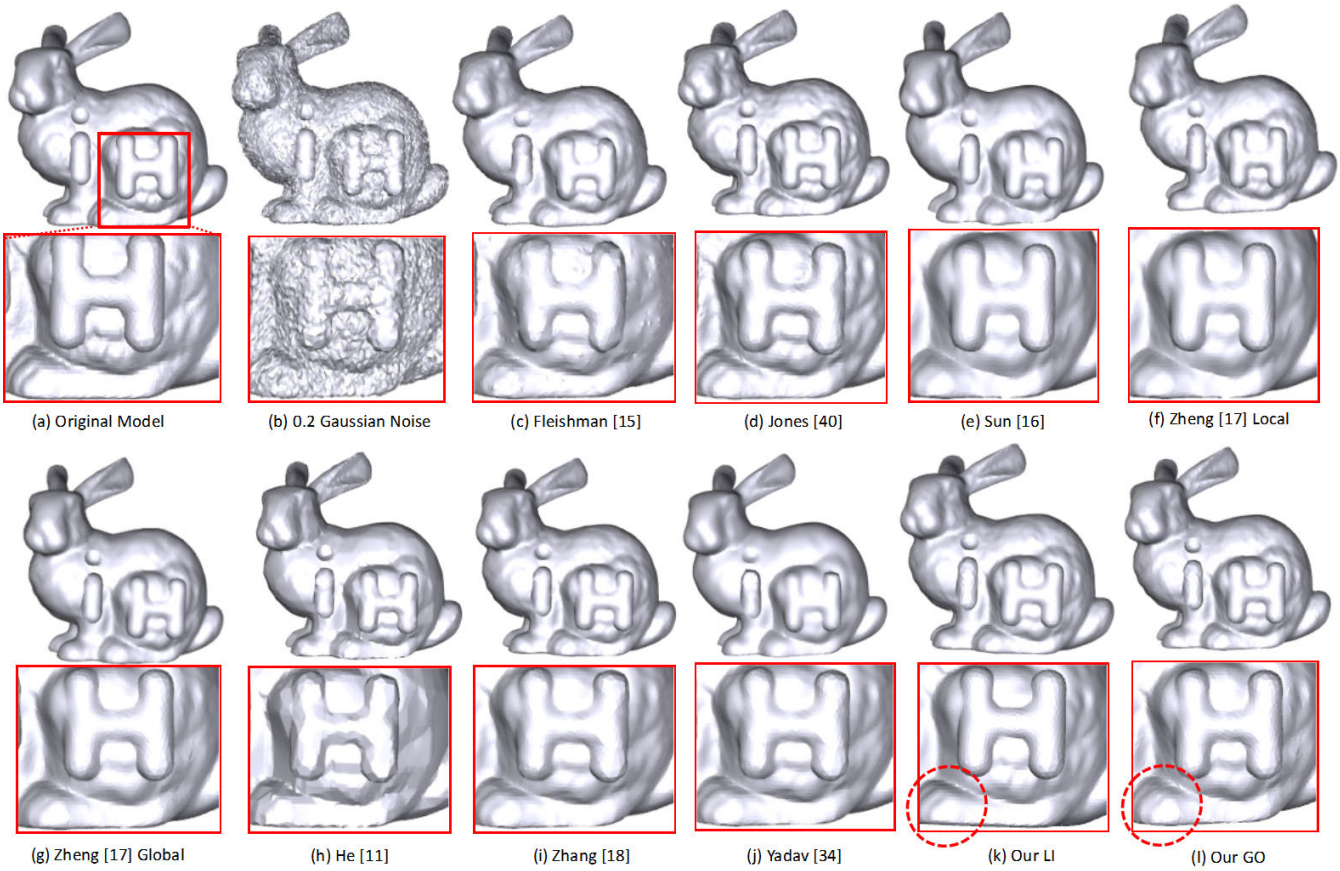


FIGURE 12. Visual comparisons on the synthetic iH-bunny model.

is the Gaussian convolution parameter of the global filter. σ and σ_g are the adaptive coefficients of local and global algorithms, respectively. The adaptive coefficients may be selected according to the noise level of the model concerned.

1) QUANTITATIVE COMPARISONS

To present fair comparisons, we adjust all the parameters in the above methods as shown in Table 2. Here, we introduce the Mean Square Angular Error (MSAE) to measure the filtering results, which is defined as:

$$MSAE = E\left[(n_i^{GT}, \widetilde{n}_i)\right], \tag{16}$$

where, n_i^{GT} and $\widetilde{n}_i$ denote the ground truth facet normal and filtered facet normal respectively, and $E[\cdot]$ represents the expectation. It can be seen from the table that, our linear interpolation scheme can obtain low MSAE values on different models. And the global scheme also obtains ideal results.

2) QUALITATIVE COMPARISONS

For the qualitative comparisons, we conduct experiments on two main categories of models, namely, synthetic models and real-world raw models, and we add different levels of Gaussian noises to the concerned models. We specifically choose the classical synthetic models of fandisk, pyramid, skull, dragon, bunny, Julius and Max. The fandisk model (as shown in Fig. 7) contains both strong features and weak features, which are the focal points of denoising algorithms. We adopt the feature detection method introduced in [40] to detect the features on filtered models achieved with different algorithms. It can be seen that, both of our schemes can preserve the features very well and the weak features detected by our schemes can better approximate those on the original model.

Furthermore, the denoising results in Fig. 8 to Fig. 13 demonstrate that our method can obtain more feature-preserving results compared with other methods, which can be seen more clearly via the zoom-in figures. For the curve on the head of the skull model, the teeth on the skull model, the details on the dragon head, Fleishman *et al.* [26] and Jones *et al.* [39] preserve the details, however, the noises are not removed thoroughly, and more artifacts are brought in. Although Zhang *et al.* [29] and Yadav *et al.* [25] can remove the noises, the geometric features have been removed to different levels. The local and global methods from Zheng *et al.* [27] obtain better results compared with others, while our method takes additional consideration on the discriminations on features and noises, thus can better preserve the geometric details. For the pyramid and Max model with

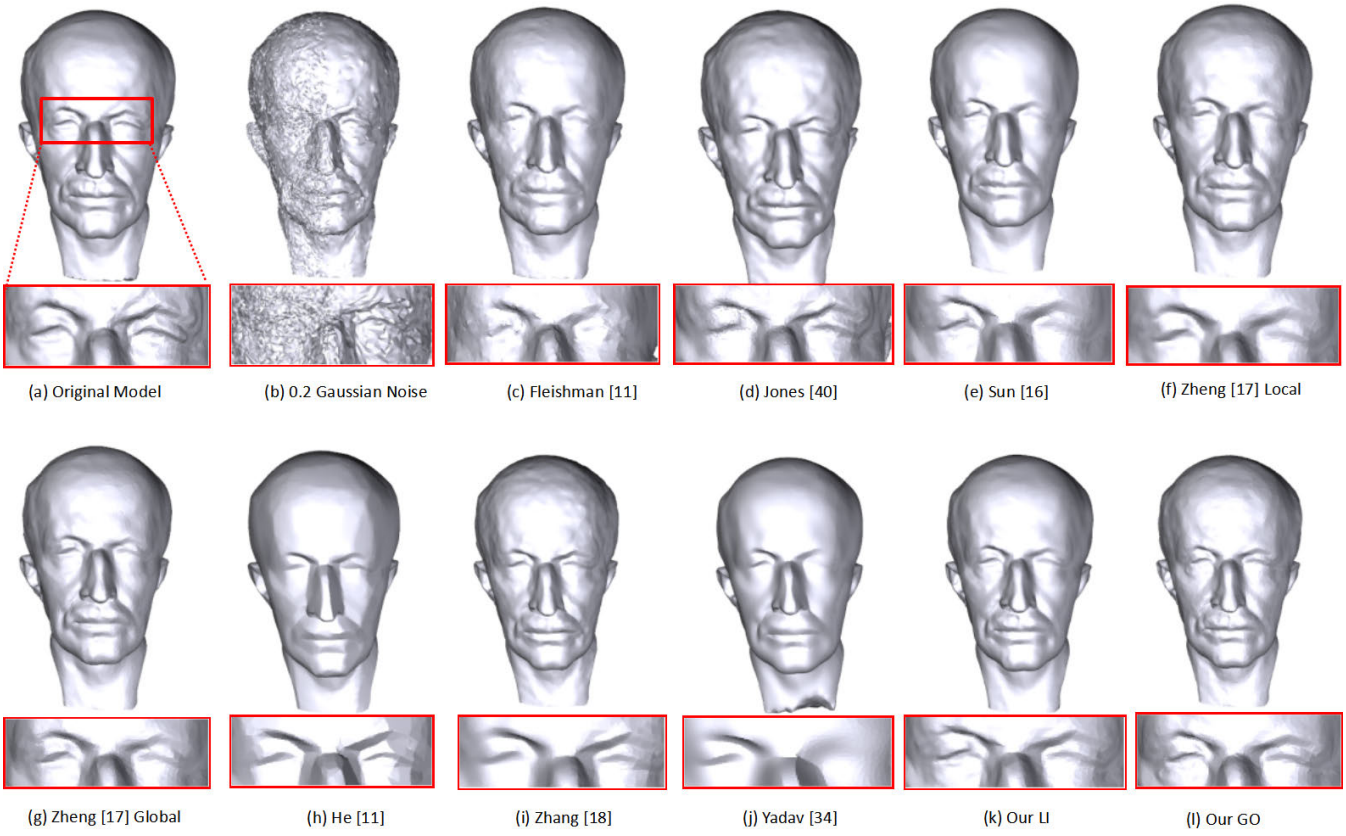


FIGURE 13. Visual comparisons on the synthetic Max model with irregular sampling rate.

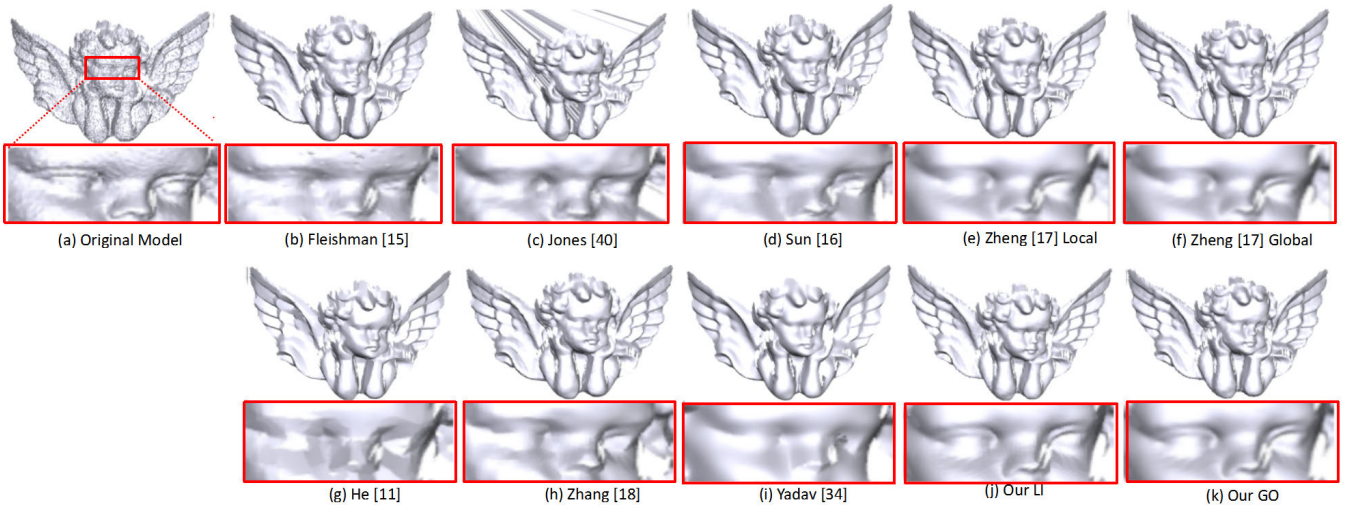


FIGURE 14. Visual comparisons on the raw angel model.

irregular sampling rate, the methods from Sun *et al.* [20], Fleishman *et al.* [26] and Yadav *et al.* [25] obtain the obvious asymmetric denoising results. The global optimization based methods (e.g. Zheng *et al.* [27] and our GO scheme) can better deal with the irregular sampling. Besides, our schemes can preserve the detailed features.

As for the real-world raw data, we conduct experiments on the angel, wilhelm, rabbit and vase model (as shown from Fig. 14 to Fig. 17). Although the raw models contain low level of noises, they are contaminated with different kinds of defects, such as the open boundaries, holes, etc. From the zoom-in subfigures in Fig. 14, it can be seen that

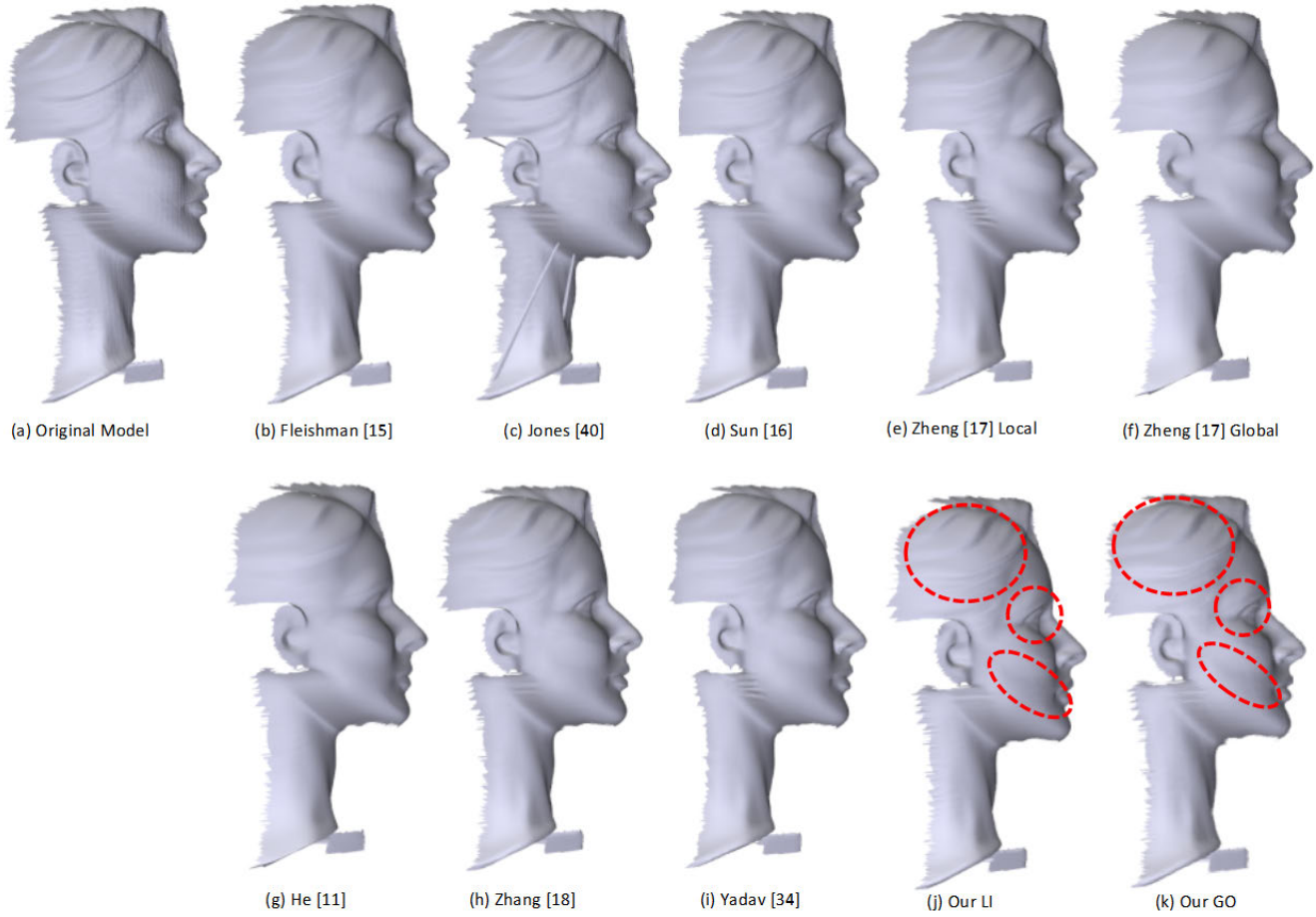


FIGURE 15. Visual comparisons on the raw wilhelm model, with the red dashed circles indicating the recovered feature regions.

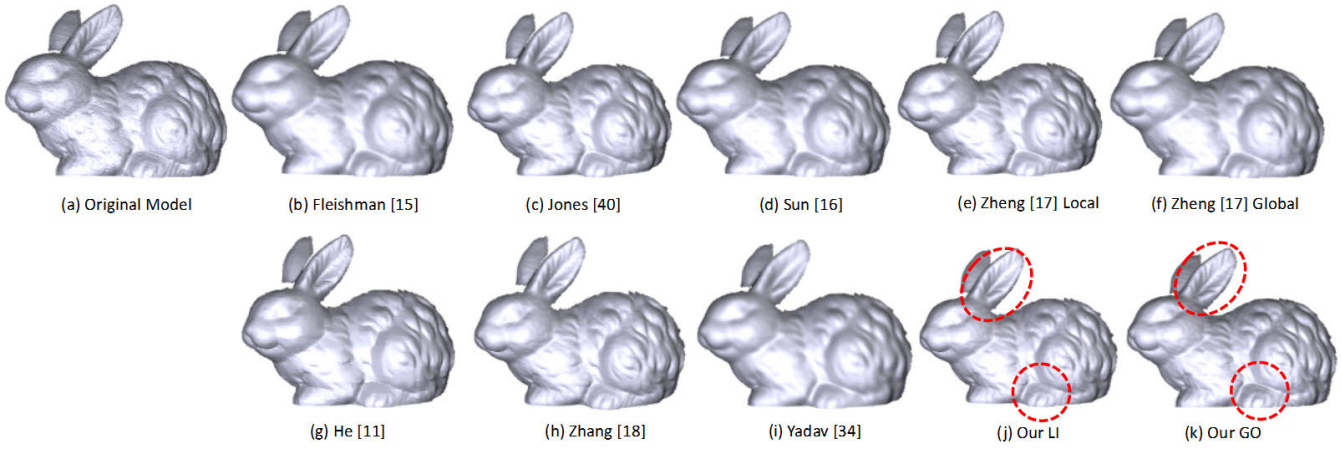


FIGURE 16. Visual comparisons on the raw rabbit model.

Zhang *et al.* [29], Yadav *et al.* [25] and the others remove the features around eyes and nose of the angel model. In comparison, our methods can remove the noises, and at the same time well preserve the geometric structures and details (esp. the eyes of the angels). The wilhelm model and rabbit model (as shown in Fig. 15 and Fig. 16) both contain the hairs and muscles. And the denoising results show that our methods can simultaneously remove the noises and preserve the details and weak features as highlighted with the dashed red circles in the figures. As for the vase model in Fig. 17, since the boundary

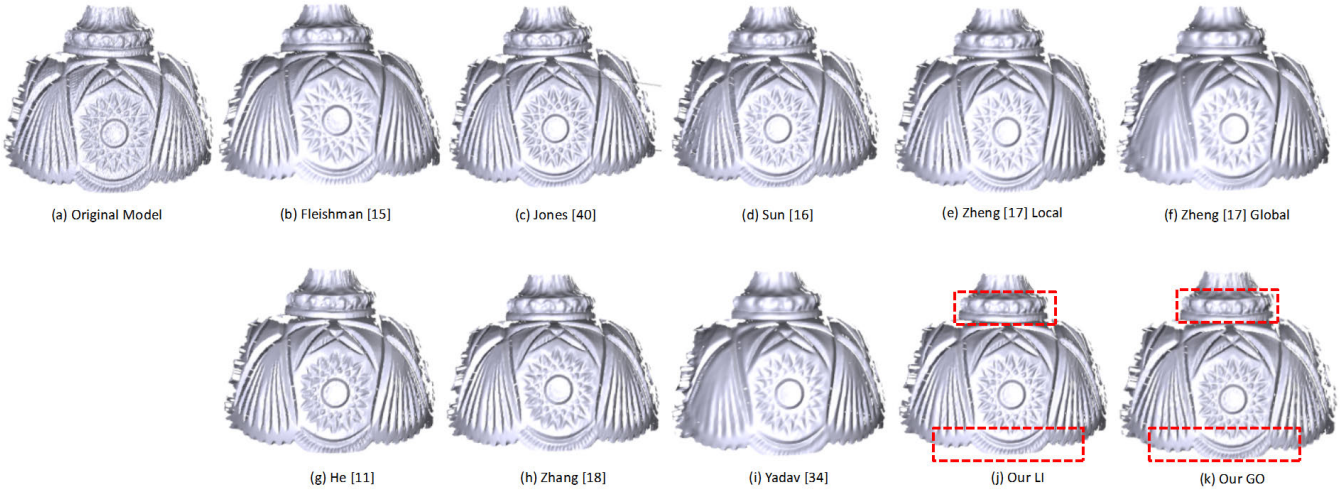


FIGURE 17. Visual comparisons on the raw vase model.

of it is open and irregular, the method in Jones *et al.* [39] may encounter the boundary problem. And the other methods destroy the textures in the vase model to different levels. Thanks to our developmental guidance, our adaptive schemes can better detect and protect the features and main textures.

VII. CONCLUSION

This paper has proposed two schemes for mesh denoising, namely, adaptive linear interpolation based scheme and global optimization based scheme for feature-preserving denoising. By introducing developmental guidance into both of the proposed schemes, we make the denoising processes adaptive to the positions of feature and smooth regions. Extensive experiments have shown the feature-preserving results obtained by the two schemes. With the proposed stop criterion, the iterative linear interpolation based scheme is time-saving and robust to severe noises. And the global scheme can better preserve the details compared to the existing global methods, and shows robustness to irregular sampling. The feature-preserving property of the two schemes verify the effectiveness of the proposed developmental guidance. Our future work will concentrate on exploring more kinds of developmental kind of guidance and dealing with more complex models in the real world with different defects.

REFERENCES

[1] Y. Sun, H. Chen, J. Qin, H. Li, M. Wei, and H. Zong, "Reliable rolling-guided point normal filtering for surface texture removal," *Comput. Graph. Forum*, vol. 38, no. 10, pp. 721–732, 2019.

[2] M. Wei, X. Guo, J. Huang, H. Xie, H. Zong, R. Kwan, F. L. Wang, and J. Qin, "Mesh defiltering via cascaded geometry recovery," *Comput. Graph. Forum*, vol. 38, no. 7, pp. 591–605, Oct. 2019.

[3] W. Zhao, X. Liu, S. Wang, X. Fan, and D. Zhao, "Graph-based feature-preserving mesh normal filtering," *IEEE Trans. Vis. Comput. Graphics*, early access, Sep. 27, 2019, doi: 10.1109/TVCG.2019.2944357.

[4] Q. Shen, Y. Sheng, C. Chen, G. Zhang, and H. Ugail, "A PDE patch-based spectral method for progressive mesh compression and mesh denoising," *Vis. Comput.*, vol. 34, no. 11, pp. 1563–1577, Nov. 2018.

[5] W. Zhao, X. Liu, Y. Zhao, X. Fan, and D. Zhao, "NormalNet: Learning-based normal filtering for mesh denoising," 2019, *arXiv:1903.04015*. [Online]. Available: http://arxiv.org/abs/1903.04015

[6] F. Gawrilowicz and J. A. Barentzen, "Optimal, non-rigid alignment for feature-preserving mesh denoising," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2019, pp. 415–423.

[7] Z. Liu, S. Zhong, Z. Xie, and W. Wang, "A novel anisotropic second order regularization for mesh denoising," *Comput. Aided Geometric Design*, vol. 71, pp. 190–201, May 2019.

[8] S. Zhong, Z. Xie, J. Liu, and Z. Liu, "Robust mesh denoising via triple sparsity," *Sensors*, vol. 19, no. 5, p. 1001, Feb. 2019.

[9] J. Wang, J. Huang, F. L. Wang, M. Wei, H. Xie, and J. Qin, "Data-driven geometry-recovering mesh denoising," *Comput.-Aided Des.*, vol. 114, pp. 133–142, Sep. 2019.

[10] H. Chen, J. Huang, O. Remil, H. Xie, J. Qin, and Y. Guo, "Structure-guided shape-preserving mesh texture smoothing via joint low-rank matrix recovery," *Comput.-Aided Des.*, vol. 115, pp. 122–134, Oct. 2019.

[11] S. Liu, S. Rho, R. Wang, and F. Jiang, "Feature-preserving mesh denoising based on guided normal filtering," *Multimedia Tools Appl.*, vol. 77, no. 17, pp. 23009–23021, Sep. 2018.

[12] T. Li, J. Wang, H. Liu, and L.-G. Liu, "Efficient mesh denoising via robust normal filtering and alternate vertex updating," *Frontiers Inf. Technol. Electron. Eng.*, vol. 18, no. 11, pp. 1828–1842, Nov. 2017.

[13] J. Hurtado, M. Gattass, A. Raposo, and J. Coelho, "Adaptive patches for mesh denoising," in *Proc. 31st SIBGRAPI Conf. Graph., Patterns Images (SIBGRAPI)*, Oct. 2018, pp. 1–8.

[14] M. Centin and A. Signoroni, "Mesh denoising with (Geo) metric fidelity," *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 8, pp. 2380–2396, Aug. 2018.

[15] B. Liu, J. Cao, W. Wang, N. Ma, B. Li, L. Liu, and X. Liu, "Propagated mesh normal filtering," *Comput. Graph.*, vol. 74, pp. 119–125, Aug. 2018.

[16] G. Taubin, "A signal processing approach to fair surface design," in *Proc. 22nd Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1995, pp. 351–358.

[17] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr, "Implicit fairing of irregular meshes using diffusion and curvature flow," in *Proc. 26th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1999, pp. 317–324.

[18] Y. Zhao, H. Qin, X. Zeng, J. Xu, and J. Dong, "Robust and effective mesh denoising using l0 sparse regularization," *Comput.-Aided Des.*, vol. 101, pp. 82–97, Aug. 2018.

[19] T. Li, W. Liu, H. Liu, J. Wang, and L. Liu, "Feature-convinced mesh denoising," *Graph. Models*, vol. 101, pp. 17–26, Jan. 2019.

[20] X. Sun, P. L. Rosin, R. Martin, and F. Langbein, "Fast and effective feature-preserving mesh denoising," *IEEE Trans. Vis. Comput. Graphics*, vol. 13, no. 5, pp. 925–938, Sep. 2007.

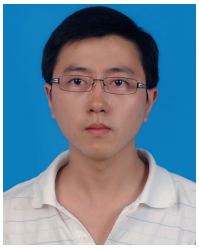
[21] J. Wang, X. Zhang, and Z. Yu, "A cascaded approach for feature-preserving surface mesh denoising," *Comput.-Aided Des.*, vol. 44, no. 7, pp. 597–610, Jul. 2012.

[22] X. Lu, Z. Deng, and W. Chen, "A robust scheme for feature-preserving mesh denoising," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 3, pp. 1181–1194, Mar. 2016.
[23] Y. Xing, L. Bai, J. Tan, and P. Hong, "Robust mesh denoising based on collaborative filters," *J. Adv. Mech. Des., Syst., Manuf.*, vol. 12, no. 4, 2018, Art. no. JAMDSM0084.
[24] P.-S. Wang, Y. Liu, and X. Tong, "Mesh denoising via cascaded normal regression," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–12, Nov. 2016.
[25] S. K. Yadav, U. Reitebuch, and K. Polthier, "Robust and high fidelity mesh denoising," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 6, pp. 2304–2310, Jun. 2019.
[26] S. Fleishman, I. Drori, and D. Cohen-Or, "Bilateral mesh denoising," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 950–953, Jul. 2003.
[27] Y. Zheng, H. Fu, O. K.-C. Au, and C.-L. Tai, "Bilateral normal filtering for mesh denoising," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 10, pp. 1521–1530, Oct. 2011.
[28] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. ICCV*, vol. 98, Jan. 1998, p. 2.
[29] W. Zhang, B. Deng, J. Zhang, S. Bouaziz, and L. Liu, "Guided mesh normal filtering," *Comput. Graph. Forum*, vol. 34, no. 7, pp. 23–34, 2015.
[30] P.-S. Wang, X.-M. Fu, Y. Liu, X. Tong, S.-L. Liu, and B. Guo, "Rolling guidance normal filter for geometric processing," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–9, Nov. 2015.
[31] L. He and S. Schaefer, "Mesh denoising via L_0 minimization," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 64–75, 2013.
[32] H. Zhang, C. Wu, J. Zhang, and J. Deng, "Variational mesh denoising using total variation and piecewise constant function space," *IEEE Trans. Vis. Comput. Graphics*, vol. 21, no. 7, pp. 873–886, Jul. 2015.
[33] R. Wang, Z. Yang, L. Liu, J. Deng, and F. Chen, "Decoupling noise and features via weighted ℓ_1-analysis compressed sensing," *ACM Trans. Graph.*, vol. 33, no. 2, pp. 18–30, 2014.
[34] M. Wei, L. Liang, W.-M. Pang, J. Wang, W. Li, and H. Wu, "Tensor voting guided mesh denoising," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 931–945, Apr. 2017.
[35] S. K. Yadav, U. Reitebuch, and K. Polthier, "Mesh denoising based on normal voting tensor and binary optimization," *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 8, pp. 2366–2379, Aug. 2018.
[36] X. Lu, W. Chen, and S. Schaefer, "Robust mesh denoising via vertex pre-filtering and L1-median normal filtering," *Comput. Aided Geometric Des.*, vol. 54, pp. 49–60, May 2017.
[37] M. Wei, J. Yu, W.-M. Pang, J. Wang, J. Qin, L. Liu, and P.-A. Heng, "Bi-normal filtering for mesh denoising," *IEEE Trans. Vis. Comput. Graphics*, vol. 21, no. 1, pp. 43–55, Jan. 2015.
[38] M. Wei, J. Huang, X. Xie, L. Liu, J. Wang, and J. Qin, "Mesh denoising guided by patch normal co-filtering via kernel low-rank recovery," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 10, pp. 2910–2926, Oct. 2019.
[39] T. R. Jones, F. Durand, and M. Desbrun, "Non-iterative, feature-preserving mesh smoothing," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 943–949, Jul. 2003.
[40] X. Wang, J. Cao, X. Liu, B.-J. Li, X.-Q. Shi, and Y.-Z. Sun, "Feature detection of triangular meshes via neighbor supporting," *J. Zhejiang Univ. Sci. C*, vol. 13, no. 6, pp. 440–451, 2012.

NANNAN LI (Member, IEEE) was born in Dalian, Liaoning, China, in 1987. She received the B.S. and Ph.D. degrees in computational mathematics from the Dalian University of Technology, Dalian, in 2010 and 2017, respectively.

She is currently an Associate Professor in information science and technology with Dalian Maritime University. Her research interests include computer graphics, differential geometry analysis, and machine learning.

SHAOYANG YUE was born in Nanyang, Henan, China, in 1993. He received the B.S. degree in network engineering from the School of Information Engineering, Huanghuai University, Zhumadian, Henan. He is currently pursuing the M.S. degree in computer science and technology with Dalian Maritime University. His research interests include computer graphics and deep learning.



ZHIYANG LI was born in Dandong, Liaoning, China, in 1982. He received the Ph.D. degree in computation mathematics from the Dalian University of Technology, China, in 2011.

He is currently an Associate Professor with the Information Science and Technology College, Dalian Maritime University, China. He has published more than 50 papers in international journals and conferences. His research interests include computer vision, cloud computing, and computer graphics.

SHENGFA WANG was born in Dalian, Liaoning, China, in 1984. He received the B.S. and Ph.D. degrees in computational mathematics from the Dalian University of Technology in 2007 and 2013, respectively. His research interests include computer graphics, diffusion geometry, and differential geometry processing and analysis.

He is currently an Associate Professor with the DUT-RU International School of Information and Software Engineering, and the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology.

HUI WANG was born in Shijiazhuang, Hebei, China, in 1984. He received the Ph.D. degree in computational mathematics from the Dalian University of Technology, in 2013.

He is currently an Associate Professor with the School of Information Science and Technology, Shijiazhuang Tiedao University, China. His research interests include computer graphics, digital geometry processing, and image processing.