Title:
Surface Smoothing Based on Discrete Orthogonal Polynomial

Authors:
Lei Lu, lulei@haut.edu.cn, Henan University of Technology
Ning Li, ln_1013@163.com, Henan University of Technology
Wei Pan, vpan@foxmail.com, OPT Machine Vision Tech Co.,Ltd
Wenming Tang, tangwenming@sziit.edu.cn, Shenzhen Institute of Information Technology

Keywords:
Mesh Processing, Surface Denoising, Dimensional Accuracy, Process Optimization

DOI: 10.14733/cadconfP.2024.301-306

Introduction:

The triangular mesh surface model serves as a versatile representation method for three-dimensional objects, playing a significant role in various aspects of computer-aided engineering, including reverse design, rapid prototyping, 3D printing, and virtual simulation. Consequently, the reconstruction and processing of triangular mesh surface models constitute a significant research focus within the field. A large amount of 3D grid data in industry is obtained through 3D measurement, where the resulting grid model often incorporates noise introduced during scanning and digitization. This noise can significantly impede the usability of the grid model, necessitating its removal during the pre-processing stage.

Taubin [1] proposed a non-shrinkage two-step smoothing method implemented through signal processing. Vollmer et al. [2]introduced a simple, fast Laplacian smoothing algorithm. However, it results in surface contraction and cannot preserve sharp features. Researchers later developed various isotropic denoising methods based on volume preservation, frequency or differential properties [3]. Due to the difficulty in preserving geometric features with isotropic methods, anisotropic denoising methods gained widespread attention. Various methods based on diffusion or differential information have been proposed, such as [4] To address this issue, anisotropic methods have emerged. An early work by Hildebrandt and Polthier [4] uses mean curvature flow to preserve features while denoising mesh shapes. Subsequently, two-step methods, such as bilateral filtering techniques [5] and others [9, 10, 11], have been proposed to better preserve features. These methods typically involve normal smoothing and vertex updating, showing promising results for robust, feature-preserving mesh denoising.In recent years, researchers have explored classification techniques for distinguishing features during mesh denoising [12, 13, 14]. However, these strategies often focus on local neighborhoods and are susceptible to noise. To mitigate this, Lu et al. [10] introduced a pre-filtering technique before denoising to reduce the impact of excessive noise

In this paper, we adopt a method that first estimates the normal direction using non-local similar structures [15], and then filters the normal direction using orthogonal polynomial fitting line matrices to achieve filtering and denoising effects on the triangular mesh [6]. The matrix constructed with non-local homologous similar structures has been proven to be more representative and robust. Additionally, orthogonal polynomials of different orders do not interfere with each other, reducing both time costs and computational complexity in the denoising process. Experiments have shown that our method can

produce results comparable to or of higher quality than state-of-the-art methods. The specific algorithm flow chart is shown as the following Fig. 1.

Principle:
The algorithm is divided into three stages, namely the construction of non-local similar structure, orthogonal polynomial fitting vector matrix, according to the position of the new normal update point, the specific algorithm flow chart is shown as follows

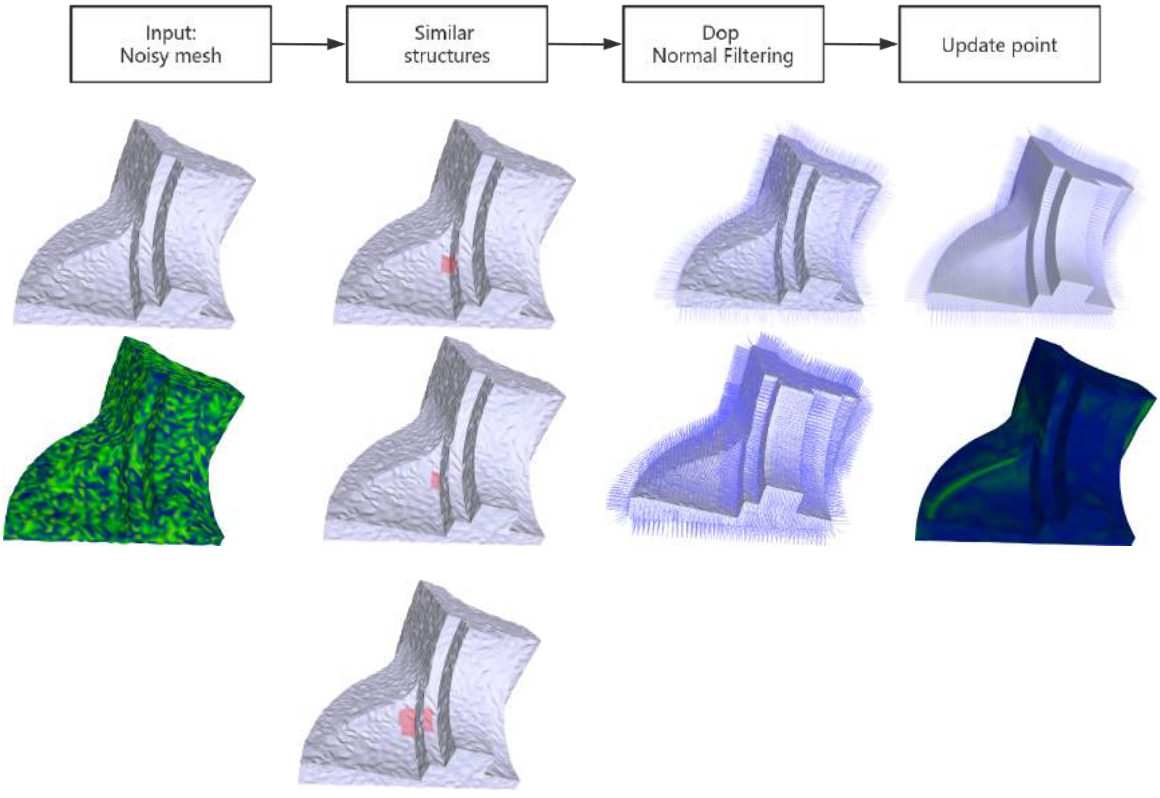

Fig. 1: A general overview of the algorithm.

1. The construction of non-local similar structures:

Given a surface mesh $M = (V, E, F)$ with N vertices, we have the set of vertices V, the set of edges E and the set of faces F. The i-th vertex $v_i \in V$ is represented by the coordinates $v_i = (x_i, y_i, z_i)$ Each face f_i. has a local structure S_i, which consists of the r -ring structure of f_i. Due to the surface irregularities in the noisy triangular mesh model, estimating the normal direction using a single face may not be accurate enough. Therefore, we use the local structure to calculate a normal tensor, which replaces the initial normal, The specific definitions are as follows:

$$T_{ij} = \eta(\|c_i - c_j\|)\phi(\theta_{ij})\mathbf{n}_j^T\mathbf{n}_j \tag{2.1}$$

Where c_i refers to the centroid of the current face f_i, and n_i refers to the normal of f_i. η and ϕ are the weights induced by spatial distances and intersection angles θ_{ij} of two neighboring normals, The specific

definitions are as follows:

$$\Phi(\theta) = e^{-\left(\frac{1-\cos\theta}{1-\cos\delta\theta}\right)^2} \tag{2.2}$$

$$\eta(x) = e^{-\left(\frac{x}{\delta p}\right)^2} \tag{2.3}$$

σ_p and σ_θ are the parameters, Based on experiments, σ_p is set to twice the maximum distance between two points in the set of faces of the r-ring neighborhood of f_i. σ_θ is set to be 30°.

For each local structure S_i, we can derive the accumulated tensor by aggregating all the induced tensor votes (2.4), $i, j \in S_i$. This final tensor encodes the local structure, which provides a reliable, representative orientation.

$$T_i = \sum_{j \in S_i} T_{ij} \tag{2.4}$$

When obtaining the local principal components, we use matrix decomposition technique to decompose the T_i(3*3) matrix. Consequently, three eigenvalues and their corresponding eigenvectors are obtained. We select the eigenvector λ_{iso} associated with the maximum eigenvalue V_{iso} as the principal direction of this local structure. This is also referred to as the local tensor.

After obtaining the representative direction V_{iso} for each local structure, we use V_{iso} as a reference to filter out faces in the face set that are approximately aligned with this local direction. This approach eliminates faces with large errors in the local structure, preparing for the next step of filtering out non-locally aligned similar structures. The filtering criterion involves comparing the angle θ_{th} between the normal of each face f_i in set S_i and the direction V_{iso}. If the noise level in the model is high, the threshold angle θ_{th} is set larger; otherwise, it is set smaller. Expanding the neighborhood range, searching for the R-ring neighborhood of face f_i (with $R>r$), for each face f_i within the neighborhood, identifying its local co-aligned structure S_j^{iso}, comparing S_j^{iso} with S_i^{iso}, and determining the angle θ_{th} between them. For cases with high noise, the angle threshold θ_{th} is set to 40°. while for cases with low noise, it is set to20°. Identifying all local co-aligned structures that meet the criteria and adding them to S_i^{nls} , ultimately obtaining the non-locally co-aligned similar structures for each face in the model.

2. Orthogonal polynomial fitting estimation for normal

Legendre polynomials are polynomials defined on the interval $(-1, 1)$ that are orthogonal with respect to the weight function $w(x) = 1$. In fact, Legendre polynomials are a special case of Jacobi polynomials when $\alpha = \beta = 0$. The expression of Legendre polynomials is

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} \left[(x^2 - 1)^n\right] \tag{2.5}$$

The recurrence formula for Legendre polynomials is:

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x) \tag{2.6}$$

The specific expansion calculation of Legendre polynomials is as follows.

$$\begin{cases} \varphi_0(x) = 1 \\ \varphi_1(x) = x - \frac{(x,\varphi_0(x))}{(\varphi_0(x),\varphi_0(x))}\varphi_0(x) = x - \frac{\int_1^1 x dx}{\int_{-1}^1 dx} \cdot 1 = x \\ \varphi_2(x) = x^2 - \frac{\left(x^2,\varphi_0(x)\right)}{(\varphi_0(x),\varphi_0(x))}\varphi_0(x) - \frac{\left(x^2,\varphi_1(x)\right)}{(\varphi_1(x),\varphi_1(x))}\varphi_1(x) = x^2 - \frac{1}{3} \\ \varphi_3(x) = x^3 - \frac{\left(x^3,\varphi_0(x)\right)}{(\varphi_0(x),\varphi_0(x))}\varphi_0(x) - \frac{\left(x^3,\varphi_1(x)\right)}{(\varphi_1(x),\varphi_1(x))}\varphi_1(x) - \frac{\left(x^3,\varphi_2(x)\right)}{(\varphi_2(x),\varphi_2(x))}\varphi_2(x) = x^3 - \frac{3}{5}x \end{cases} \tag{2.7}$$

For each non-local similar structure S_i^{nls} for the isotropic structure S_i^{iso} associated with the face f_i, we append the face normals of S_i^{nls} as rows to a matrix M. Note that the dimensions of this matrix are $r \times 3$. This matrix already has a maximal rank of 3 and is a low rank matrix. To make the low rank matrix approximation meaningful, we reshape the matrix M to be close to a square matrix R. irst, we need to construct the orthogonal bases for the X and Y directions of matrix R. Assuming the fitting order in the X-direction is a and in the Y-direction is b, the expressions for these two orthogonal bases can be represented as:$B_x = \{\varphi_0(x), \varphi_1(x), \varphi_2(x).....\varphi_a(x)\}$,$B_y = \{\varphi_0(y), \varphi_1(y), \varphi_2(y).....\varphi_a(y)\}$.

The expression of the fitting matrix is obtained by discretizing B_x at uniform intervals within the range [-1, 1], resulting in a total of n nodes. From this discretization, the basis matrix $B_x(n \times a)$ and its transpose matrix $B_x^T(a \times n)$ are obtained. Similarly, the basis matrix in the Y direction $B_y(m \times b)$ and its transpose matrix $B_y^T(b \times m)$ are also obtained through m uniform intervals. Here, Z represents the matrix after orthogonal polynomial filtering.

$$Z_{m\times n} = B_{y(m\times b)} * B^T_{y(b\times m)} * R_{m\times n} * B_{x(n\times a)} * B^T_{x(a\times n)} \tag{2.8}$$

When constructing the similar structures for each face, adjacent faces form a set of similar structures $S = \{f_1, f_2, \ldots, f_n\}$, where overlapping regions may occur. In such cases, a weight is assigned to each face f_i to track its frequency of use. The final matrix is then obtained by accumulating contributions from each face, which are divided by the corresponding weight coefficient. This process results in the updated normal direction for each face.

3.After obtaining the updated normals corresponding to each triangle, we update the position of each vertex in the triangle according to the new normals. We calculate the new position of the point using the following formula:

$$x_i' = \mathbf{x}_i + \frac{1}{|F_V(i)|} \sum_{k \in F_V(i)} \mathbf{n}_k' \left(\mathbf{n}_k' \cdot (\mathbf{c}_k - \mathbf{x}_i)\right) \tag{2.9}$$

$F_V(i)$ refers to the set of torus adjacent to a triangle , andrefers to the center of the triangle where the current point is located.

Experiments and results:

In our experiments, we tested our method on numerous mesh models corrupted by synthesized or original scan noise. Additionally, we evaluated several state-of-the-art mesh denoising methods on the same test set for comparison. The selected state-of-the-art mesh denoising methods include Bilateral Normal Filter (BNF)[5], the Unilateral Normal Filter (UNF)[11], the Guided Normal Filter (GNF)[7]. the L1-median Filter (L1) [10] and Half-kernel Laplacian Operator (HLO) [8].

We compare the state-of-the-art techniques with our approach from a quantitative perspective. Specifically, we employ E_v and MSAE (mean square angular error) to respectively evaluate the positional error and normal error, as suggested by previous works [11, 5]. These two metrics are calculated between the smoothing results and their corresponding ground truth.

According to [11], E_v is the L^2 vertex-based mesh-to-mesh error metric, and MSAE measures the mean square angular error between the face normals of the denoised mesh and those of the ground truth.

$$E_v = \sqrt{\frac{1}{3\sum_{k\in F} A_k} \sum_{i\in V} \sum_{j\in F_V(i)} A_j dist(x_i', T)^2}, \tag{2.10}$$

Where A_k is the area of face k, and $dist(x_i^{'}, T)$ is the L^2 distance between the updated vertex $x_i^{'}$ and a triangle of the reference mesh T which is closest to $x_i^{'}$.

$$\text{MSAE} = \frac{\sum_{k \in F} \theta_k^2}{N_F} \tag{2.11}$$

Where θ_k is the angle between the k-th face normal of the denoised model and its corresponding normal in the ground-truth model, and N_F is the number of faces in the 3D shape.

Table 1: Quantitative comparisons with representative mesh smoothing methods.

Models	Methods	MSAE($\times 10^{-2}$)	$E_v(\times 10^{-3})$
cube	BNF	0.889	4.758
Figure 2	UNF	0.135	1.229
($\sigma_n = 0.2l_e$)	GNF	0.425	1.267
$\|V\|$:12288	L1	0.153	1.303
$\|F\|$:6146	HLO	0.153	1.303
	OURS	**0.075**	**1.224**
fandisk	BNF	12.182	8.536
Figure 2	UNF	11.324	10.437
($\sigma_n = 0.2l_e$)	GNF	10.739	6.976
$\|V\|$:7229	L1	20.817	8.147
$\|F\|$:14454	HLO	**8.545**	18.208
	OURS	10.349	**5.983**

Noisy BNF [5] UNF [11] GNF [7] L1 [10] HLO [8] Ours

Fig. 2: Coloured Ev for Cube and Fandisk with noise $\sigma_n = 0.2l_e$.

Conclusions:

In this paper, the construction of non-local similar structures based on normal matrices is more representative and robust. Orthogonal polynomials are easier to implement for preserving features in mesh denoising compared to regular polynomials. Similar to other methods, our approach has limited robustness against excessive noise and irregular triangulation. Regarding future work, we can consider fitting orthogonal polynomials separately to the normals after partitioning the mesh. For example, designating the mesh's corner and edge parts as feature segments, and the remaining parts as non-feature segments, and then applying thresholding separately to these segments. We hope that through this method, we can design more robust algorithms

Lei Lu, https://orcid.org/0000-0002-3050-6542
Ning Li, https://orcid.org/0009-0006-9149-8087
Wei Pan, https://orcid.org/0000-0002-0933-2453
Wenming Tang, https://orcid.org/0000-0002-1427-3216

References:

[1] Taubin, G.: A signal processing approach to fair surface design. In:Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, 1995, pp. 351–358. http://dx.doi.org/10.1145/218380.218386

[2] Vollmer, J.; Mencl, R.; Mueller, H.: Improved Laplacian smoothing of noisy surface meshes. In Computer graphics forum, vol. 18, pp. 131–138. Wiley Online Library, 1999. http://doi.org/10.1111/1467-8659.00334

[3] Kim, B.; Rossignac, J. Geofilter: Geometric selection of mesh filter parameters. In Computer Graphics Forum, vol. 24, pp. 295–302. Citeseer, 2005. http://doi.org/10.1111/j.1467-8659.2005.00854.x

[4] Hildebrandt, K.; Polthier, K.: Anisotropic filtering of non-linear surface features. In Computer Graphics Forum, vol. 23, pp. 391–400. Wiley Online Library, 2004. http://doi.org/10.1111/j.1467-8659.2004.00770.x

[5] Lee, K. W.; Wang, W. P.: Feature-preserving mesh denoising via bilateral normal filtering, in Ninth International Conference on Computer-Aided Design and Computer Graphics (CAD-CG'05), pp. 6, IEEE, 2005. http://doi.org/10.1145/1925059.1925086.

[6] O'Leary, P.; M. Harker, M.: Discrete polynomial moments and Savitzky-Golay smoothing, International Journal of Computer and Information Engineering, 4(12), 2010, 1993-1997. https://publications.waset.org/vol/48.

[7] Zhang, W.; Deng, B.; Zhang, J.; Bouaziz, S.; Liu, L.: Guided mesh normal filtering, *Computer Graphics Forum*, 34, 2015, 23-34, Wiley Online Library. http://doi.org/10.1111/cgf.12742.

[8] Pan, W.; Lu, X.; Gong, Y.; Tang, W.; Liu, J.; He, Y.; Qiu, G.: Hlo: Half-kernel Laplacian operator for surface smoothing, *Computer-Aided Design*, 121, 102807, 2020. http://doi.org/10.1016/j.cad.2019.102807.

[9] Chen, S.; et al.: Towards uniform point distribution in feature-preserving point cloud filtering, Computational Visual Media, 9(2), 2023, 249-263. http://doi.org/10.1007/s41095-022-0278-4.

[10] Lu, X.; Chen, W.; Schaefer, S.: Robust mesh denoising via vertex pre-filtering and l1-median normal filtering, Computer Aided Geometric Design, 54, 2017, 49-60. http://doi.org/10.1016/j.cagd.2017.02.011.

[11] Sun, X.; Rosin, P. L.; Martin, R.; Langbein, F.: Fast and effective feature-preserving mesh denoising, IEEE Transactions on Visualization and Computer Graphics, 13(5), 2007, 925-938. http://doi.org/10.1109/tvcg.2007.1065.

[12] Fan, H.; Yu, Y.; Peng, Q.: Robust feature-preserving mesh denoising based on consistent subneighborhoods, *IEEE Transactions on Visualization and Computer Graphics*, 16(2), 2009, 312-324, IEEE. http://dx.doi.org/10.1109/tvcg.2009.70.

[13] Wei, M.; Liang, L.; Pang, W. M.; Wang, J.; Li, W.; Wu, H.: Tensor voting guided mesh denoising, IEEE Transactions on Automation Science and Engineering, 14(2), 2016, 931-945. http://doi.org/10.1109/tase.2016.2553449.

[14] Wei, M.; JYu, J.; Pang, W. M.; Wang, J.; Qin, J.; Liu, L.; Heng, P. A.: Bi-normal filtering for mesh denoising, IEEE Transactions on Visualization and Computer Graphics, 21(1), 2014, 43-55. http://doi.org/10.1109/tvcg.2014.2326872.

[15] Lu, X.; Schaefer, S.; Luo, J.; Ma, L.; He, Y.: Low-rank matrix approximation for 3D geometry filtering, IEEE Transactions on Visualization and Computer Graphics, 28(4), 2020, 1835-1847. http://doi.org/10.1109/tvcg.2020.3026785.