

A novel anisotropic second order regularization for mesh denoising



Zheng Liu^{a,b}, Saishang Zhong^{a,b,*}, Zhong Xie^{a,b}, Weina Wang^c

^a Faculty of Information Engineering, China University of Geosciences, Wuhan 430074, China

^b National Engineering Research Center of Geographic Information System, China University of Geosciences, Wuhan 430074, China

^c Department of Mathematics, Hangzhou Dianzi University, Hangzhou 310018, China

ARTICLE INFO

Article history:

Available online 5 April 2019

Keywords:

Mesh denoising
Feature preserving
Anisotropic second order regularization
Augmented Lagrangian method

ABSTRACT

Mesh denoising is a fundamental, yet not well-solved problem in geometry processing. The main challenge is to remove noise while preserving geometric features and preventing unnatural effects, such as blurring of sharp features and staircase effects in smoothly curved regions. State-of-the-art mesh denoising methods still struggle with this issue. In the paper, we first propose a novel anisotropic second order regularization method to restore face normals of the noisy mesh, and then reconstruct vertex positions of the mesh according to the filtered face normals. The proposed anisotropic second order regularization can maximally recover the underlying surface from the noisy mesh without introducing the unnatural effects. Numerically, an efficient algorithm based on variable splitting and augmented Lagrangian method is proposed to solve the problem. Extensive experiments on a variety of meshes demonstrate that our mesh denoising method outperforms all the compared state-of-the-art methods visually and quantitatively.

© 2019 Published by Elsevier B.V.

1. Introduction

Mesh denoising, which aims to restore high quality meshes from noisy inputs, is an important task in digital geometry processing. The denoised meshes can be used in downstream geometry applications, such as mesh reconstruction, segmentation, animation and visualization. Recently, due to the rapid development of consumer-grade scanner devices, the design of effective mesh denoising methods has become increasingly attractive. The main challenge of mesh denoising is to remove noise while preserving geometric features as well as not introducing unnatural effects. This problem becomes more difficult when noisy meshes consisting of sharp features (including sharp edges and corners) and smooth regions (including smoothly curved regions and fine details).

Over the last decade, great efforts have been made on the mesh denoising problem. Many noticeable methods have been proposed, for examples, bilateral weighting normal filtering (Zheng et al., 2011), ℓ_0 minimization (He and Schaefer, 2013), TV normal filtering (Zhang et al., 2015a) and robust and high fidelity mesh denoising (Yadav et al., 2018b). Although these state-of-the-art methods have achieved great successes, each of them still has limitations in some respects. Specifically, bilateral weighting normal filtering (Zheng et al., 2011) usually blurs sharp features when the noise level is high, and thus cannot produce satisfactory results for the meshes containing sharp features. The recent ℓ_0 minimization (He and Schaefer, 2013) and TV normal filtering (Zhang et al., 2015a) are able to preserve sharp features well. However,

* Corresponding author at: Faculty of Information Engineering, China University of Geosciences, Wuhan 430074, China.

E-mail addresses: liu.zheng.jojo@gmail.com (Z. Liu), saishang@cug.edu.cn (S. Zhong), xiezong@cug.edu.cn (Z. Xie), wnwang2014@163.com (W. Wang).

these two methods suffer from undesired staircase effects in smoothly curved regions, especially for ℓ_0 minimization (He and Schaefer, 2013). Moreover, when the noise level increases, ℓ_0 minimization (He and Schaefer, 2013) will sharpen some fine details and produce false edges in smooth regions. The method proposed by Yadav et al. (2018b) can robustly generate high fidelity results from noisy inputs, but it sometimes oversmoothes fine details. These unnatural effects produced by the aforementioned state-of-the-art methods will degrade the quality of denoising results and cause errors in downstream geometry applications.

To overcome the above mentioned limitations, we first propose a second order regularization model by introducing a novel anisotropic second order operator. The proposed model is applied to the face normal field of the mesh. Then, vertex positions of the mesh can be reconstructed according to the filtered face normals. Taking a noisy mesh as input, our mesh denoising method preserves geometric features well and does not introduce unnatural effects. Specifically, the contributions of the paper are listed as follows:

- We design an anisotropic second order difference operator over triangulated surfaces and propose a normal filtering model using this operator. The proposed second order regularization model can preserve geometric features well and prevent unnatural effects.
- An efficient algorithm based on variable splitting and augmented Lagrangian method is presented to solve the problem.
- We discuss differences between regularization models based on our anisotropic second order, gradient and Laplace operators.

The rest of the paper is organized as follows. Section 2 briefly reviews prior state-of-the-art mesh denoising methods. In section 3, we introduce a novel anisotropic second order operator over triangulated surfaces. In section 4, we first propose a second order normal filtering model, which can be solved by an iterative algorithm. Then, we discuss differences between the second order, gradient and Laplace operators. The comparison of regularization models based on these operators is given at the end of section 4. Our mesh denoising method is compared to the state-of-the-art methods in section 5. Section 6 concludes the paper.

2. Related work

Mesh denoising is a long standing problem, which has been studied for years. Although there are lots of mesh denoising works in literature, reviewing all of the existing methods is beyond the scope of this paper. Interested readers can refer to the works (Chen and Cheng, 2005; Hildebrandt and Polthier, 2010) for a comprehensive review. Here, we only review noticeable methods which are closely related to this paper.

Filtering methods are widely applied in mesh denoising for their simplicity and efficiency, which can be classified into isotropic schemes and anisotropic schemes. Although the isotropic schemes (Taubin, 1995; Desbrun et al., 1999) can suppress noise, they smooth geometric features. Later on, in order to preserve geometric features, many anisotropic schemes were proposed (Desbrun et al., 2000; Taubin, 2001; Yagou et al., 2002; Tasdizen et al., 2002; Bajaj and Xu, 2003; Wang, 2006). When the noise level is low, the anisotropic schemes can preserve geometric features while removing noise. However, when the noise level increases, the anisotropic schemes will blur geometric features, especially for sharp features. Recently, bilateral filtering methods, adapted from image processing, have achieved great successes in mesh denoising (Jones et al., 2003; Fleishman et al., 2003). Since the bilateral filtering methods also belong to anisotropic schemes, they suffer from the problem of blurring sharp features. More recently, some methods (Zheng et al., 2011; Zhang et al., 2015b) applied the bilateral filtering to face normals, which can preserve sharp features well. However, when the noise level is high, the bilateral normal filtering proposed by Zheng et al. (2011) still blurs sharp features. The guided normal filtering (Zhang et al., 2015b) can preserve sharp features well, but it sometimes lacks the robustness to mesh topology. By using the robust error norm, the robust normal filtering (Yadav et al., 2018b) can also preserve sharp features. Yet, this method sometimes oversmoothes fine details.

In recent years, optimization-based methods have gained popularity in mesh denoising. Zheng et al. (2011) proposed a global normal filtering model based on bilateral weighting Laplacian, which can effectively remove noise while recovering fine details. Yet, it is difficult to generate satisfactory results when the noise level is high, especially for the noisy input with sharp features. By using both static and dynamic guidances, Zhang et al. (2018) introduced a filtering method, which can remove noise while preserving geometric features. Yadav et al. (2018a) proposed a framework optimizing the element-based normal voting tensor for feature-preserving mesh denoising. Moreover, efforts of nonlocal similarity have been made for mesh denoising by Li et al. (2018). Besides, the concept of sparsity has been proved successful in image processing for its excellent edge-preserving property (Wang et al., 2008; Yang et al., 2009; Wu and Tai, 2010; Xu et al., 2011; Hu and Jacob, 2012; Wu et al., 2012). Many researchers extended sparsity-based methods from image processing to geometry processing for preserving sharp features (He and Schaefer, 2013; Zhang et al., 2015a; Wu et al., 2015; Lu et al., 2017). He and Schaefer (2013) proposed using ℓ_0 minimization to remove noise by inducing the sparsity for an edge-based operator. Total variation (TV) regularization was applied on face normals for its well edge-preserving property (Zhang et al., 2015a). Wu et al. (2015) proposed a variational method combining a TV regularization and an ℓ_1 fidelity term. Lu et al. (2017) introduced an ℓ_1 -median normal filter to handle meshes with different kinds of noise and irregular surface sampling. Although these sparsity-based methods can preserve sharp features well while removing noise, they suffer undesired staircase effects in

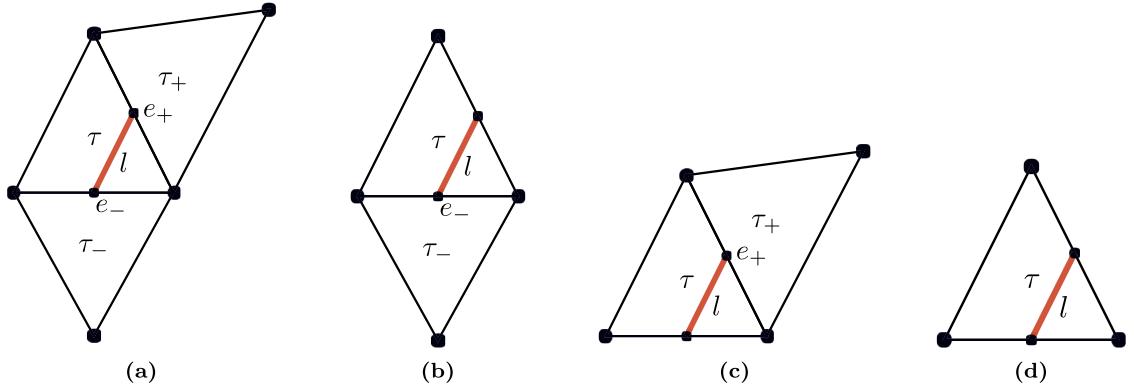


Fig. 1. (a) The illustration of second order variation over the line l colored in red; (b), (c) and (d) are the illustrations of three boundary conditions.

smoothly curved regions for their sparsity requirement. These unnatural effects in smooth regions will degrade the quality of denoising results.

Wang et al. (2014) proposed a method based on ℓ_1 -analysis compressed sensing to decouple features and noise. However, when the mesh is corrupted by high level noise, it is a challenge for their method to distinguish the features from the noise. The reason is that both the features and the noise belong to high frequency signals. More recently, a data-driven method (Wang et al., 2016) has been proposed for mesh denoising. This method first learns non-linear regression functions which map the filtered face normal descriptors to the face normals of the clean mesh, then uses the learned functions to compute filtered face normals. This method can efficiently remove noise without the assumptions about geometric features of the underlying surface and noise patterns. However, the performance of this data-driven method depends on the completeness of the training data set.

3. Anisotropic second order operator on triangulated surfaces

In this section, we introduce notations followed by definitions of a second order operator over triangulated surfaces. A mesh of arbitrary topology with no degenerate triangles in \mathbb{R}^3 is represented as M . The set of vertices, edges, and triangle faces of M are denoted as $\{v_i : i = 1, 2, \dots, V\}$, $\{e_i : i = 1, 2, \dots, E\}$, and $\{\tau_i : i = 1, 2, \dots, T\}$, respectively. Here V , E , and T are the numbers of vertices, edges, and faces of M .

Here, we define a second order difference operator over triangulated surfaces. Let e_+ and e_- be any two edges of the triangle τ , and l be a line connecting the midpoints of e_+ and e_- ; see the illustration of the descriptions in Fig. 1a. Assume $u = (u_1, u_2, \dots, u_T) \in \mathbb{R}^T$ is a scalar field on the triangulated surface M . This means that the value of u restricted on the triangle τ is u_τ . For any $u \in \mathbb{R}^T$, the isotropic second order variation of u over the line l can be defined as

$$\begin{aligned}\mathcal{D}^{iso}(u)|_l &= (u_{\tau_+} - u_\tau) - (u_\tau - u_{\tau_-}) \\ &= u_{\tau_+} + u_{\tau_-} - 2u_\tau.\end{aligned}$$

The above definition is similar to the definition of second order difference in image processing. Without loss of generality, by considering boundary conditions (see Figs. 1b, 1c and 1d), for any $u \in \mathbb{R}^T$, we can define

$$\mathcal{D}^{iso}(u)|_l = \begin{cases} u_{\tau_+} + u_{\tau_-} - 2u_\tau, & e_+ \notin \partial M \text{ and } e_- \notin \partial M, \\ u_{\tau_-} - u_\tau, & e_+ \in \partial M \text{ and } e_- \notin \partial M, \\ u_{\tau_+} - u_\tau, & e_+ \notin \partial M \text{ and } e_- \in \partial M, \\ 0, & e_+ \in \partial M \text{ and } e_- \in \partial M. \end{cases}$$

Inspired by Zheng et al. (2011), for preserving geometric features, we describe the second order variation of u in an anisotropic manner

$$\mathcal{D}^{ani}(u)|_l = \begin{cases} w_{e_+}(u_{\tau_+} - u_\tau) - w_{e_-}(u_\tau - u_{\tau_-}), & e_+ \notin \partial M \text{ and } e_- \notin \partial M, \\ w_{e_-}(u_{\tau_-} - u_\tau), & e_+ \in \partial M \text{ and } e_- \notin \partial M, \\ w_{e_+}(u_{\tau_+} - u_\tau), & e_+ \notin \partial M \text{ and } e_- \in \partial M, \\ 0, & e_+ \in \partial M \text{ and } e_- \in \partial M, \end{cases}$$

where the weight w_e is defined as

$$w_e = \exp(-\|u_{e,1} - u_{e,2}\|^2/2\sigma^2). \quad (1)$$

$u_{e,1}$ and $u_{e,2}$ of (1) are two values restricted on triangles sharing the common edge e .

Thus, for any $u \in \mathbb{R}^T$, we can define the anisotropic second order difference operator as

$$\mathbf{D}: u \mapsto \mathbf{D}(u), \mathbf{D}(u)|_l = \mathcal{D}^{ani}(u)|_l, \forall l. \quad (2)$$

Furthermore, we extend the above concepts to handle vectorial data. In particular, for a \mathfrak{N} -channel vectorial field

$$\mathbf{u} = \underbrace{u \times \cdots \times u}_{\mathfrak{N}} \in \mathbb{R}^{\mathfrak{N} \times T},$$

the second order operator $\mathbf{D}(\mathbf{u})$ can be computed channel by channel.

4. Variational mesh denoising

Our method belongs to two-stage mesh denoising methods, i.e., face normal filtering followed by vertex updating (Zheng et al., 2011; Zhang et al., 2015a,b; Lu et al., 2017; Yadav et al., 2018b). First, we apply a second order regularization method to optimize face normals of the noisy mesh. Then, we reconstruct the vertex positions according to the filtered face normals. At last, we discuss the normal filtering models based on different operators including the anisotropic second order, gradient and Laplace operators.

4.1. Anisotropic second order normal filtering

To optimize face normals of the noisy input, we present a variational model with an anisotropic second order regularization. An algorithm based on variable splitting and augmented Lagrangian method is introduced to solve the problem.

4.1.1. Normal filtering model

Given a noisy mesh M^{in} , we denote its face normals as N^{in} . To remove noise of N^{in} , we treat the face normals N as a variable and propose the following normal filtering model:

$$\min_{N \in C_N} \left\{ \frac{\alpha}{2} \sum_{\tau} \text{area}(\tau) \|N_{\tau} - N_{\tau}^{in}\|^2 + \sum_l \text{len}(l) \|\mathbf{D}(N)|_l\| \right\}, \quad (3)$$

where $C_N = \{N \in \mathbb{R}^{3 \times T} : \|N_{\tau}\|_2 = 1, \forall \tau\}$, α is a positive parameter, $\text{area}(\tau)$ is the area of triangle τ and $\text{len}(l)$ is the length of line l . The first term of model (3) is the fidelity term, which helps the solution to harmonize well with the input face normals. The second term is the smooth term based on the anisotropic second order operator (2). This term can make the solution smooth and feature-preserving.

4.1.2. Augmented Lagrangian method for solving the normal filtering model

Due to the nondifferentiability and nonlinear constraints of the normal filtering model (3), it is difficult to directly solve the problem. Fortunately, variable splitting and augmented Lagrangian method (ALM) are well known for their efficiency in ℓ_1 related optimization problems (Wu and Tai, 2010; Wu et al., 2012; Zhang et al., 2015a). Here, we introduce an auxiliary variable and employ ALM to solve the minimization problem of (3). Furthermore, since the weights (1) of the second order operator are estimated from the noisy input, we dynamically update them at each iteration for improving the quality of the denoising result.

We first introduce an auxiliary variable P and reformulate the problem (3) as

$$\min_{N, P} R(P) + \psi(N) + \frac{\alpha}{2} \sum_{\tau} \text{area}(\tau) \|N_{\tau} - N_{\tau}^{in}\|^2 \quad (4)$$

$$\text{s.t., } P = \mathbf{D}(N),$$

where $R(P) = \sum_l \text{len}(l) \|P_l\|$, and

$$\psi(N) = \begin{cases} 0, & N \in C_N \\ +\infty, & N \notin C_N. \end{cases}$$

For the above constrained optimization problem, we define the following augmented Lagrangian function

$$L(N, P; \lambda) = R(P) + \psi(N) + \frac{\alpha}{2} \sum_{\tau} \text{area}(\tau) \|N_{\tau} - N_{\tau}^{in}\|^2 + \sum_l \text{len}(l) (\lambda_l \cdot (P_l - \mathbf{D}(N)|_l)) + \frac{r}{2} \sum_l \text{len}(l) \|P_l - \mathbf{D}(N)|_l\|^2,$$

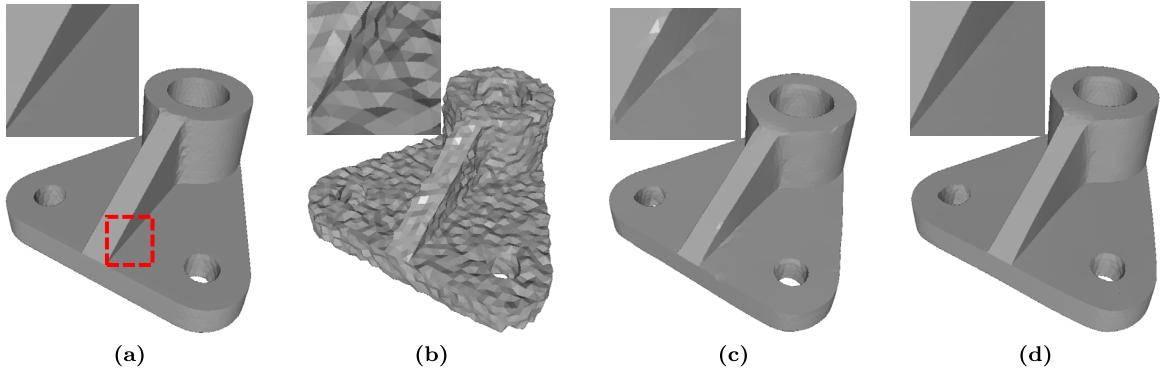


Fig. 2. Denoising results of Part (corrupted by Gaussian noise, standard deviation = 0.2 mean edge length). From left to right: clean mesh, noisy mesh, results of solving model (3) with fixed weights and dynamic weights.

where λ is a Lagrange multiplier and r is a positive penalty coefficient. This primal variables update procedure can be separated into two subproblems: the N-subproblem and P-subproblem.

For solving the N-subproblem, we fix the variable P and solve the following minimization problem

$$\min_{\mathbf{N}} \psi(\mathbf{N}) + \frac{\alpha}{2} \sum_{\tau} \text{area}(\tau) \|\mathbf{N}_{\tau} - \mathbf{N}_{\tau}^{in}\|^2 + \frac{r}{2} \sum_l \text{len}(l) \|\mathbf{D}(\mathbf{N})|_l - (\mathbf{P}_l + \frac{\lambda_l}{r})\|^2.$$

This problem is a quadratic optimization with the unit normal constraints $\psi(\mathbf{N})$. Here, we adopt an approximate strategy to solve the problem. We first ignore $\psi(\mathbf{N})$ and solve the following quadratic programming

$$\min_{\mathbf{N}} \frac{\alpha}{2} \sum_{\tau} \text{area}(\tau) \|\mathbf{N}_{\tau} - \mathbf{N}_{\tau}^{in}\|^2 + \frac{r}{2} \sum_l \text{len}(l) \|\mathbf{D}(\mathbf{N})|_l - (\mathbf{P}_l + \frac{\lambda_l}{r})\|^2, \quad (5)$$

then project the solution to a unit sphere. The solution of the problem (5) is achieved by a sparse linear system, which can be solved using various numerical packages such as Eigen, Taucs, and Math Kernel Library (MKL).

Similarly, we can solve the P-subproblem by fixing the variable N, and solve

$$\min_{\mathbf{P}} R(\mathbf{P}) + \frac{r}{2} \sum_l \text{len}(l) \|\mathbf{P}_l - (\mathbf{D}(\mathbf{N})|_l - \frac{\lambda_l}{r})\|^2. \quad (6)$$

As the problem (6) is decomposable, we can solve each \mathbf{P}_l independently. For each \mathbf{P}_l , we need to solve the following problem

$$\min_{\mathbf{P}_l} \|\mathbf{P}_l\| + \frac{r}{2} \|\mathbf{P}_l - (\mathbf{D}(\mathbf{N})|_l - \frac{\lambda_l}{r})\|^2$$

which has a closed form solution as

$$\mathbf{P}_l = \max(0, 1 - \frac{1}{r\|\Psi\|})\Psi, \quad (7)$$

where $\Psi = \mathbf{D}(\mathbf{N})|_l - \frac{\lambda_l}{r}$.

In summary, the overall algorithm is outlined in Algorithm 1. In each iteration, Algorithm 1 first solves the above two subproblems, and then updates the Lagrange multiplier and weights (1). We dynamically update the weights (1) in each iteration according to the updated N in the previous step. This dynamic weighting strategy makes our algorithm be more robust to the input noisy mesh. The reason is explained as follows. Since noise and sharp features belong to high frequency signals, the weights (1) estimated from the noisy mesh may not be accurate. If we fix these inaccurate weights during the iterations, some of the noise will be left in the denoising result, and some sharp features will be blurred. This phenomenon will become more serious, when the noise level increases. In contrast, Algorithm 1 with the dynamic weighting scheme can greatly improve the quality of the denoised result. An example can be seen in Fig. 2.

4.2. Vertex updating

After obtaining filtered face normals by minimizing the normal filtering model (3), we reconstruct the denoised mesh using the vertex updating scheme presented by Zhang et al. (2018). This vertex updating scheme can overcome the trianglewise orientation ambiguity problem produced by the traditional vertex updating scheme proposed by Sun et al. (2007). Since the scheme proposed by Zhang et al. (2018) is iterative, the number of iterations is empirically fixed as 30 in our experiments for producing well results. Due to this vertex updating scheme is not our contribution, we refer readers to the work (Zhang et al., 2018) for further information.

Algorithm 1: ALM for Solving Anisotropic Second Order Normal Filtering Model (3).

Initialization: $N^{-1} = 0, P^{-1} = 0, \lambda^0 = 0, k = -1, K = 70, \epsilon = 1e-5;$

repeat

- Solve N-subproblem**
For fixed (λ^k, P^{k-1}) , compute N^k from (5);
Normalize N^k ;
- Solve P-subproblem**
For fixed (λ^k, N^k) , compute P^k from (7);
- Update** Lagrange multiplier
 $\lambda^{k+1} = \lambda^k + r(P^k - D(N^k));$
- Update** weights (1) according to N^k ;

until $\sum_{\tau} \text{area}(\tau) \|N_{\tau}^k - N_{\tau}^{k-1}\|^2 < \epsilon$ or $k \geq K$;

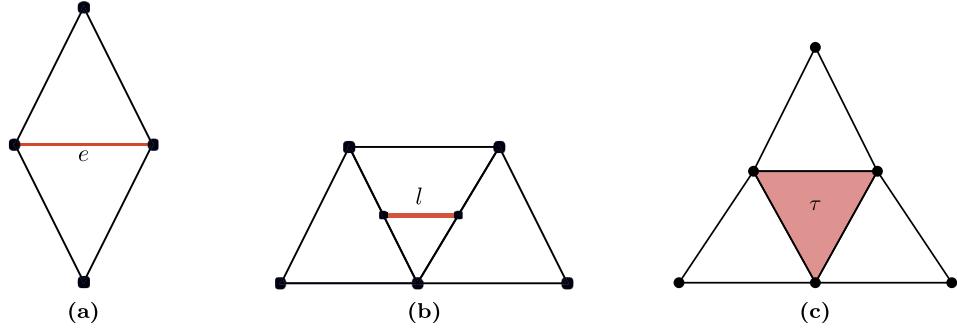


Fig. 3. The illustration of (a) first order difference over edge e , (b) second order difference over line l and (c) Laplacian coordinate over triangle τ .

4.3. Discussion on regularization models using the anisotropic second order, gradient, and Laplace operators

In this subsection, we compare four regularization models applied in the face normal field. These models are based on the second order, gradient and Laplace operators respectively. Basically, discretizations of the three operators are totally different. Our anisotropic second order operator (2) can be seen as a set of second order differences defined over lines of the mesh; see Fig. 3b. The discretization of gradient operator is the set of first order differences over edges (Zhang et al., 2015a); see Fig. 3a. The Laplace operator can be seen as a set of Laplacian coordinates over triangle faces (Zheng et al., 2011); see Fig. 3c. Without considering boundary conditions, the numbers of face elements related to the first order difference, second order difference and Laplacian coordinate are 2, 3, and 4, respectively.

The regularization models based on these three operators also exhibit different performances. For clarity and fairness, we introduce two ℓ_1 -norm normal filtering models using the aforementioned gradient and Laplace operators respectively. The model using the gradient operator is given as

$$\min_{N \in C_N} \left\{ \frac{\alpha}{2} \sum_{\tau} \text{area}(\tau) \|N_{\tau} - N_{\tau}^{in}\|^2 + \sum_e \text{len}(e) \|\mathbf{G}(N)|_e\| \right\}, \quad (8)$$

where \mathbf{G} is the gradient operator defined by Zhang et al. (2015a). The model with the Laplace operator is designed as

$$\min_{N \in C_N} \left\{ \frac{\alpha}{2} \sum_{\tau} \text{area}(\tau) \|N_{\tau} - N_{\tau}^{in}\|^2 + \sum_{\tau} \text{area}(\tau) \|\mathbf{L}(N)|_{\tau}\| \right\}, \quad (9)$$

where \mathbf{L} is the Laplace operator defined by Zheng et al. (2011). Then, we compare the denoising results produced by the normal filtering models (3), (8) and (9). To further show abilities of the above models, we also compare them with the global normal filtering model proposed by Zheng et al. (2011), which is actually the ℓ_2 -norm Laplacian normal filtering model. As we can see in Fig. 4, all of these four models can effectively remove noise. However, the result produced by the total variation normal filtering model (8) suffers staircase effects in smooth regions; see Fig. 4c. Although the ℓ_1 -norm Laplacian model (9) preserves sharp features better than the ℓ_2 -norm Laplacian model proposed by Zheng et al. (2011), both of them blur sharp features in some extent. In contrast, our second order regularization model (3) can produce the satisfactory result without unnatural effects; see Fig. 4b.

Compared to the other two potential operators, our second order operator has the following advantages. First, the ℓ_1 -norm regularization model (3) with our second order operator can overcome the staircase effect problem introduced by the model (8) using the gradient operator. The reason will be explained by one 1D example in the following. Second, the number of face elements related to our second order operator is less than those of the Laplace operator. That is to say, for

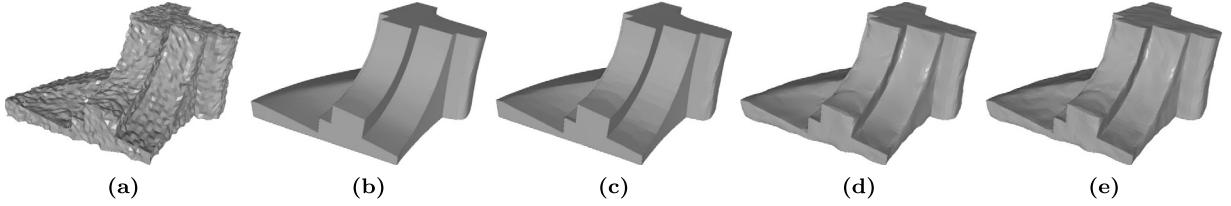


Fig. 4. Denoising results of Fandisk (corrupted by Gaussian noise, standard deviation = 0.2 mean edge length). From left to right: noisy mesh, denoising results produced by our second order regularization model (3), total variation normal filtering model (8), ℓ_1 -norm Laplacian normal filtering model (9), and global bilateral normal filtering model proposed by Zheng et al. (2011), respectively.

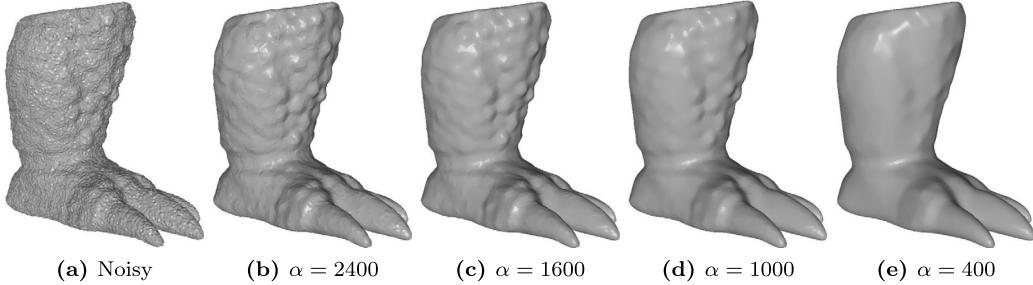


Fig. 5. Denoising results for α with fixed σ . From left to right: noisy mesh (corrupted by Gaussian noise, standard deviation = 0.2 mean edge length) and results with different α .

preserving sharp features, our second order operator is more robust than the Laplace operator. To intuitively show the above advantages of our second order operator, we give a 1D example. Since our second order operator and the Laplace operator have the same discretization in 1D, we only compare our second order operator with the first order one. Suppose that we observe a 1D signal with the values $\{1, 2, 3, 4, 5, 6\}$ at consecutive locations. In some sense, such a signal can be considered to be smooth. However, with the appropriate parameter setting, the first order model (8) may produce the solution as $\{2, 2, 2, 5, 5, 5\}$ (considering the sum of first order variations of this solution), which has a staircase between 2 and 5. However, if we use the second order model (3) to deal with the observed 1D signal, the solution may be just $\{1, 2, 3, 4, 5, 6\}$. In contrast to the previous solution, there is no staircase effect in the latter solution because the second order variations of this solution vanish everywhere. We believe that this aspect is the main distinguishing feature of the proposed approach which leads to better experimental results.

5. Experimental results and comparisons

We evaluate our mesh denoising method ASO on various kinds of CAD, non-CAD and raw scanned meshes. The tested meshes are corrupted by either synthetic or raw scanned noise. The synthetic noise is generated by a zero-mean Gaussian function with standard deviation proportional to the mean edge length of the ground truth shape. We provide visual and quantitative comparisons of our mesh denoising method to state-of-the-art methods including robust and high fidelity mesh denoising (Yadav et al., 2018b), TV normal filtering (Zhang et al., 2015a), ℓ_0 minimization (He and Schaefer, 2013), and local and global bilateral normal filtering methods (Zheng et al., 2011), abbreviated as RHM, TV, ℓ_0 , LocalBF, and GlobalBF, respectively. For the method RHM, we perform the code provided by its authors; For the other four methods, we have implemented them based on their published articles by using C++. All of the examples are run on a laptop with a Intel i7 core 2.6 GHz processor and 8 GB RAM. For showing the faceting effect, all of the examples are rendered in the flat-shading model.

5.1. Parameter choices

To the best of our knowledge, most mesh denoising methods have parameters, which need to be tuned manually for producing satisfactory results. Our method ASO also has two parameters, i.e., α and σ . α is used to balance the fidelity and smooth terms of the normal filtering model (3), and σ is the kernel width of the Gaussian function in the smooth term of (3).

The parameter α affects the degree of denoising results. Fig. 5 shows results of different α with fixed σ . As we can see in Fig. 5b, if α is too large, the result will be close to the noisy input. The noise cannot be effectively removed. Instead, if α is too small, geometric features and fine details will be blurred, illustrated in Fig. 5e. Beyond that, there exists a range of α for our method ASO producing visually well results, shown in Figs. 5c and 5d. To produce satisfactory results, we empirically set α in the range of [40, 600] for CAD meshes, and [1000, 3500] for non-CAD and scanned meshes. Moreover, α should be set with a smaller value for a higher level of noise, and vice versa.

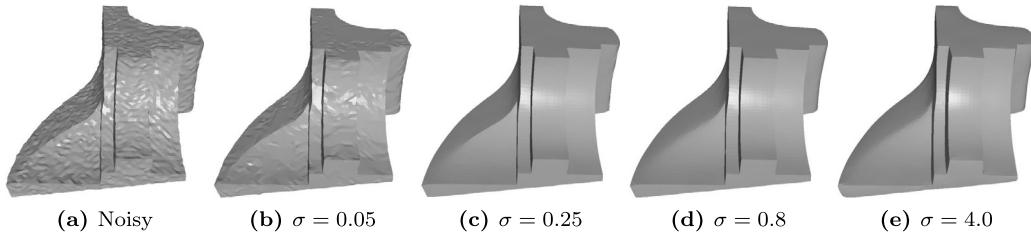


Fig. 6. Denoising results for σ with fixed α . From left to right: noisy mesh (corrupted by Gaussian noise, standard deviation = 0.2 mean edge length) and results with different σ .

The parameter σ controls the feature-preserving ability of our method ASO. Fig. 6 demonstrates some results of different σ with fixed α . If σ is too small, some noise will be left in the result demonstrated in Fig. 6b. The reason is that, some of the noise will be considered as features when σ is too small. As we can see in Figs. 6c, 6d, and 6e, with the increasing of σ , more and more geometric features are smoothed. More specifically, when $\sigma = 0.8$, the shallow edge of the tested mesh is blurred showed in Fig. 6d. When σ is too large, some large-scale features (sharp edges and corners) are blurred; see Fig. 6e.

5.2. Qualitative comparisons and examples

In this subsection, we compare the performance of our mesh denoising method ASO with those of the aforementioned state-of-the-art methods (RHM, TV, ℓ_0 , LocalBF, and GlobalBF) on both CAD and non-CAD meshes as well as raw scanned meshes. For each method, we carefully tune its parameters for producing visually optimal results.

Fig. 7 demonstrates results on CAD meshes containing both sharp features and smooth regions (including smoothly curved and flat regions). As we can see, TV and ℓ_0 preserve sharp features well. However, TV suffers from staircase effects in smoothly curved regions; see Fig. 7d. This phenomenon is more serious for ℓ_0 , which produces some false edges in smooth regions, shown in Fig. 7e. The reasons are as follows. TV uses low order information of the mesh, which is not suitable to describe piecewise smooth signals. ℓ_0 produces these artifacts for its high sparsity requirement. On the contrary, RHM, LocalBF, and GlobalBF do a good job of recovering smooth regions, but they blur sharp features in varying degrees; see Figs. 7c, 7f, and 7g. The reason is that both LocalBF and GlobalBF as well as RHM cannot clearly distinguish sharp features from noise. As a result, the state-of-the-art methods can either handle sharp features or smooth regions well. Compared to these methods, our method ASO can preserve sharp features and simultaneously recover smooth regions well; see Fig. 7b. Thus, for CAD meshes, the visual comparisons in Fig. 7 show that ASO is noticeably better than all the other compared methods.

In Fig. 8, we show results on non-CAD meshes with fine details. As we can see, our method ASO and all the other state-of-the-art methods can remove noise effectively. However, TV tends to sharpen the fine details, shown in Fig. 8d. Due to the high sparsity requirement, ℓ_0 makes this phenomenon even worse. It flattens a lot of fine details; see Fig. 8e. In contrast, RHM oversmoothes some of the fine details as shown in Fig. 8c. Compared to these methods, ASO, LocalBF, and GlobalBF generate visually better results; see Figs. 8b, 8f, and 8g. However, from the quantitative comparison demonstrated in subsection 5.3, we find errors of ASO are always lower than both LocalBF and GlobalBF. Thus, for non-CAD meshes, ASO produces better results than all the other compared methods.

Fig. 9 shows denoising results on the real scanned mesh. As we can see, our method ASO, TV, and LocalBF produce visually well results; see Figs. 11b, 11d, and 11f. In contrast, RHM and GlobalBF blur the geometric features shown in Figs. 11c and 11g. ℓ_0 suffers from serious staircase effects in smooth regions illustrated in Fig. 11e. Recently, more and more scanned meshes have been captured by Kinect sensors. Thus, we also verify the effectiveness of ASO on meshes produced by Kinect. The results are demonstrated in Fig. 10. As we can see, even for the meshes produced by Kinect with the high level of noise, ASO still can suppress noise effectively while preserving most geometric features.

Our anisotropic second order operator is computed irrespective of size of neighbor triangles that makes our denoising method cannot deal with highly irregular sampling noisy meshes; see subsection 5.4 for more details. However, due to our second order operator is scaled by the mid-line, our mesh denoising method still has some ability to deal with irregular sampling meshes. As we can see in Fig. 11, for the noisy mesh without extremely irregular sampling, our method is better than all the other compared methods.

We also show the robustness of our method against different levels of noise in Fig. 12. As can be seen in Figs. 12a, 12b, and 12c, our method ASO can effectively remove noise while preserving sharp features when the noise level is smaller than the feature size. On the contrary, when the noise level is bigger than the feature size, our method cannot produce the satisfactory result; see Fig. 12d for example.

5.3. Quantitative comparisons

The above qualitative comparisons show that our method ASO can produce visually better results than all the other five state-of-the-art methods. In this subsection, we further compare these methods quantitatively. We use two error metrics,

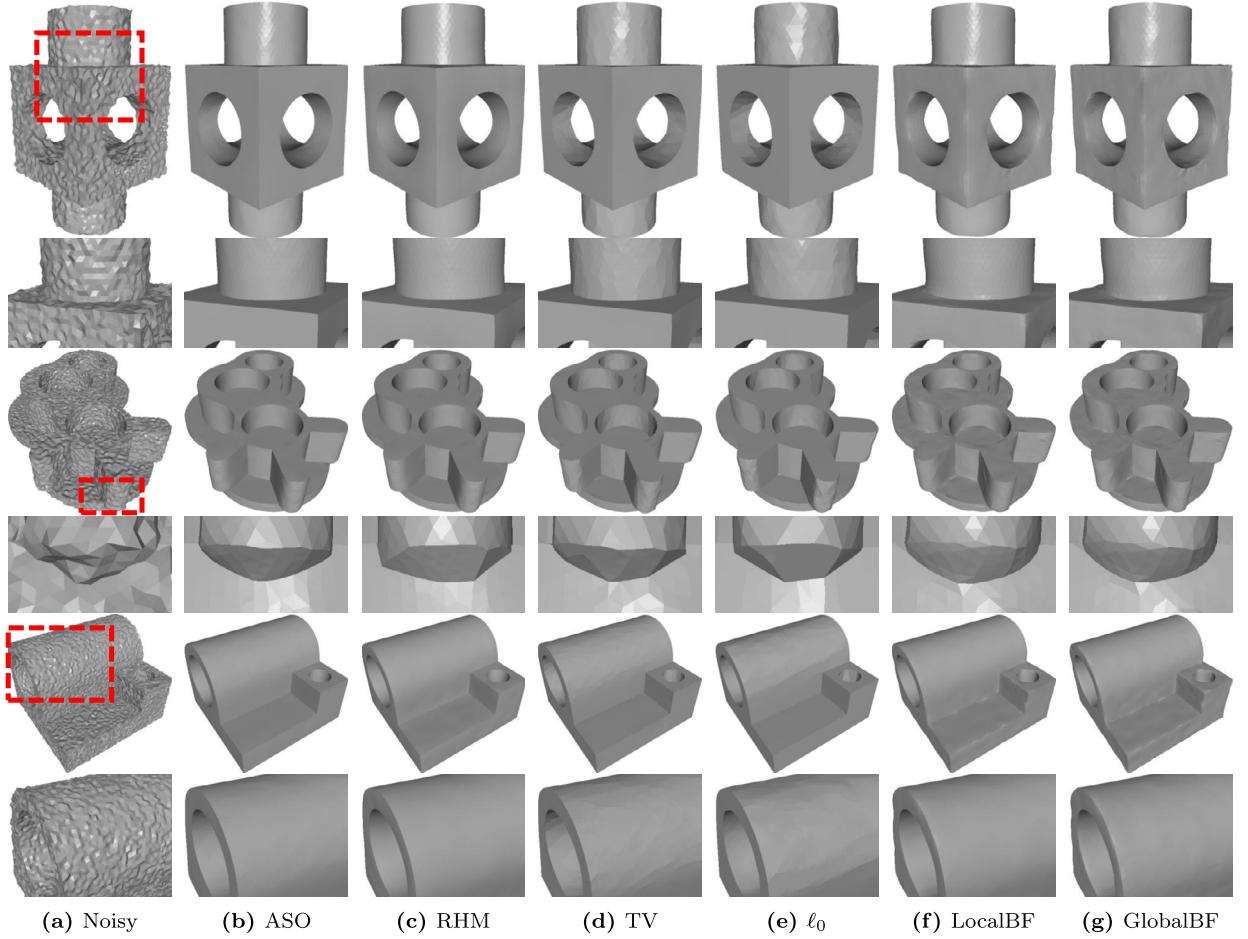


Fig. 7. Comparisons of denoising methods on CAD meshes (corrupted by Gaussian noise, standard deviation = 0.2 mean edge length). From left to right: input noisy meshes, denoising results produced by our method ASO, RHM (Yadav et al., 2018b), TV (Zhang et al., 2015a), ℓ_0 (He and Schaefer, 2013), LocalBF (Zheng et al., 2011), and GlobalBF (Zheng et al., 2011). The even rows are zoomed-in views.

Table 1

Quantitative evaluation results of Figs. 7, 8, and 11 for all the tested mesh denoising methods.

Meshes($ V , F $)	MASE($\times 10^{-3}$), $E_{v,2}(\times 10^{-3})$; Time (in Seconds)					
	ASO	RHM	TV	ℓ_0	LocalBF	GlobalBF
Block(8.8K, 17.6K)	3.90, 1.76 ; 4.02	4.20, 3.20; 14.0	5.10, 1.83; 1.02	5.70, 2.48; 8.19	9.80, 2.75; 1.48	12.5, 3.23; 0.31
Casting(19.4K, 38.8K)	4.50, 0.98 ; 15.3	5.05, 1.38; 27.9	5.30, 1.02; 2.84	6.30, 1.78; 14.5	7.80, 1.06; 0.84	8.70, 1.14; 0.30
Joint(20.9K, 41.8K)	2.30, 1.62 ; 16.7	4.03, 2.39; 39.6	3.20, 2.25; 1.94	3.99, 1.94; 26.4	6.30, 3.56; 8.44	6.40, 2.53; 0.91
Gargoyle(25K, 50K)	18.7 , 1.34; 19.1	46.5, 1.88; 87.7	45.0, 1.63; 5.49	38.8, 2.0; 27.4	19.0, 1.04 ; 1.45	20.0, 1.31; 0.86
Bunny(34.8K, 69.7K)	16.9, 1.48 ; 26.7	26.9, 2.65; 76.9	21.2, 1.87; 6.49	31.4, 3.61; 38.3	17.6, 1.50; 2.97	26.4, 2.10; 1.19
Planck(30.9K, 61.8K)	18.0 , 1.49; 26.1	25.6, 2.66; 97.9	29.0, 1.36 ; 11.9	33.2, 1.89; 56.6	25.8, 2.40; 1.98	20.2, 1.89; 3.94

mean square angular error (MSAE) and L_2 vertex-based error ($E_{v,2}$), to measure the fidelity of the denoising result to the ground truth shape. Specifically, MSAE is to estimate the mean square angular error between face normals of the ground truth shape and those of the denoising result. $E_{v,2}$ is to measure the positional error between the clean mesh and the denoised one. The definitions of these two metrics can be found in the works (Zheng et al., 2011; Zhang et al., 2015a). Our method ASO is compared to the other state-of-the-art methods by using these two metrics for the examples in Figs. 7, 8, and 11. The results are recorded in Table 1. As can be seen, the values of MSAE of ASO are smaller than the other five compared methods for all the examples. In addition, in the most cases, ASO has the least L_2 vertex-based errors ($E_{v,2}$). These quantitative evaluations show that ASO outperforms the other compared methods.

We also record CPU costs for all of the tested methods in Table 1. As we can see, LocalBF is the fastest method, while RHM is the slowest one. Our method ASO is faster than ℓ_0 , but slower than TV and GlobalBF. Due to the dynamic weighting scheme is adopted in our method, the computational time of our method is still acceptable.

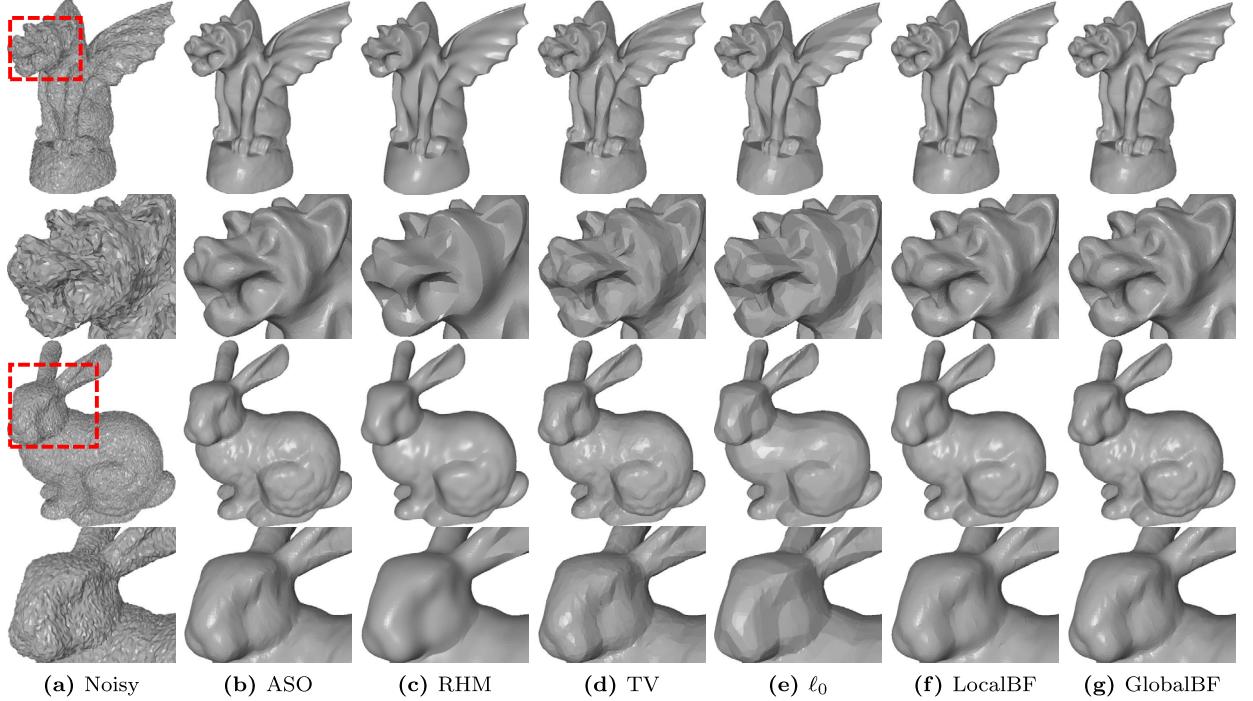


Fig. 8. Comparisons of denoising methods on non-CAD meshes (corrupted by Gaussian noise, standard deviation = 0.2 mean edge length). From left to right: input noisy meshes, denoising results produced by our method ASO, RHM (Yadav et al., 2018b), TV (Zhang et al., 2015a), ℓ_0 (He and Schaefer, 2013), LocalBF (Zheng et al., 2011), and GlobalBF (Zheng et al., 2011). The second and fourth rows are zoomed-in views.

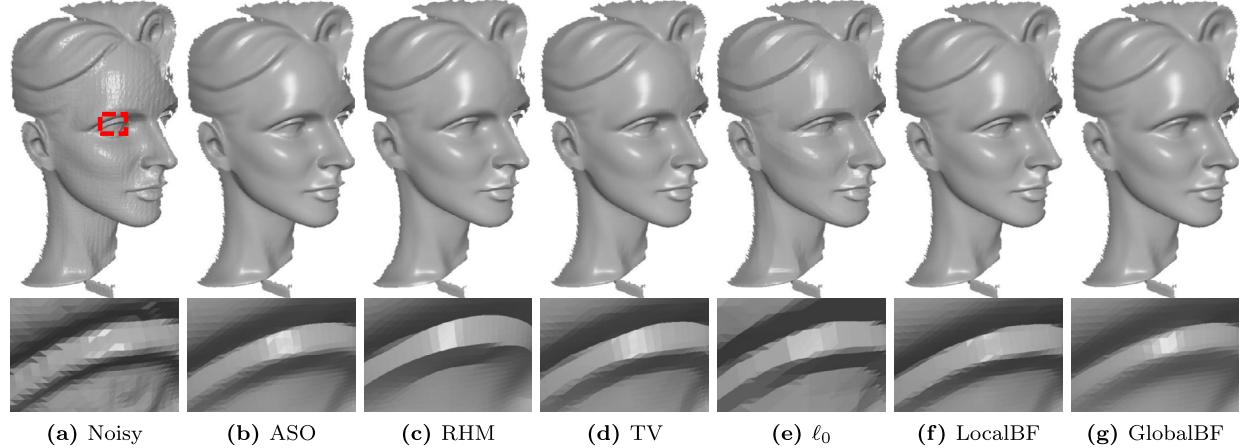


Fig. 9. Comparison of denoising methods on the real scanned mesh. From left to right: input noisy mesh, denoising results produced by our method ASO, RHM (Yadav et al., 2018b), TV (Zhang et al., 2015a), ℓ_0 (He and Schaefer, 2013), LocalBF (Zheng et al., 2011), and GlobalBF (Zheng et al., 2011). The second row is the zoomed-in view.

5.4. Limitation

Although the effectiveness of our mesh denoising method has been verified by the above experiments, it also has the limitation that it cannot deal with the noisy mesh with extremely irregular sampling; see Fig. 13 for example. However, for this case, no methods can be found producing satisfactory results.

6. Conclusion

In this paper, we present a mesh denoising method based on a novel anisotropic second order normal filtering model. The proposed normal filtering model can preserve sharp features and simultaneously recover smooth regions as well as prevent unnatural effects. We solve the problem by variable splitting and augmented Lagrangian method. Extensive experiments on



Fig. 10. Denoising results of scanned meshes captured by Kinect sensors. The first row shows raw scanned meshes, while the second row shows denoising results produced by our method ASO.

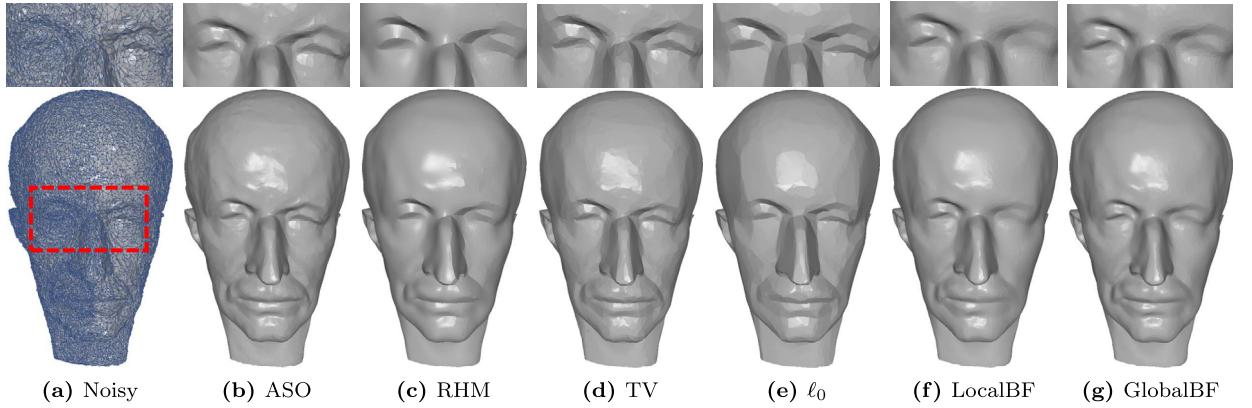


Fig. 11. Comparison of denoising methods on the irregular sampling mesh (corrupted by Gaussian noise, standard deviation = 0.2 mean edge length). From left to right: input noisy mesh, denoising results produced by our method ASO, RHM (Yadav et al., 2018b), TV (Zhang et al., 2015a), ℓ_0 (He and Schaefer, 2013), LocalBF (Zheng et al., 2011), and GlobalBF (Zheng et al., 2011). The top row is the zoomed-in view.

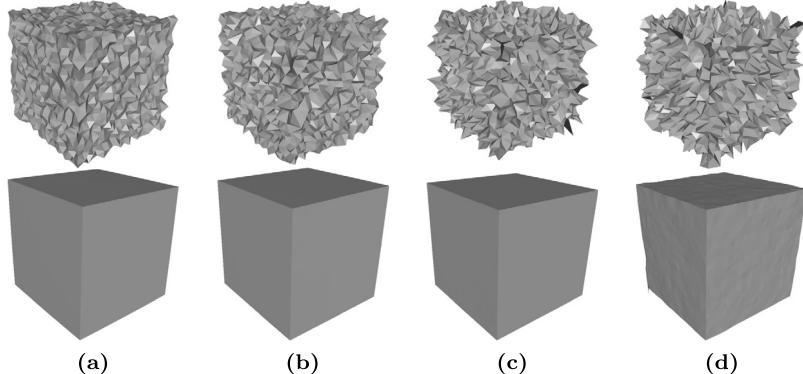


Fig. 12. Denoising results of Cube corrupted by different levels of noise. The first row shows noisy meshes (corrupted by Gaussian noise, standard deviation = 0.3, 0.5, 0.7, and 0.9 mean edge length, respectively). The second row shows the corresponding results produced by our method ASO.

a variety of meshes show that our mesh denoising method outperforms the other five state-of-the-art methods both visually and quantitatively.

Several problems are left open. First, the anisotropic second order operator can be extended to define over point clouds. Second, this second order operator can be used in other geometry processing applications, such as mesh segmentation, repairing, and simplification.

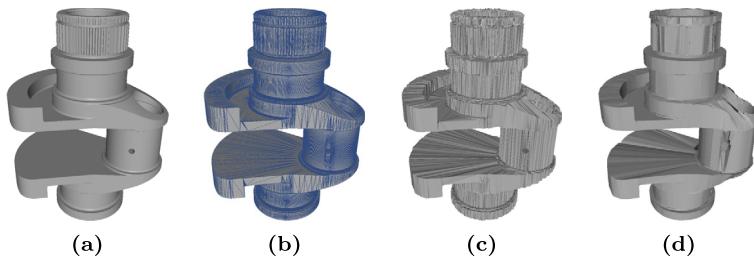


Fig. 13. A failed result produced by our method with extremely irregular sampling. From left to right: clean mesh, wireframe mode of the clean mesh, noisy mesh, and result produced by our method ASO.

Acknowledgements

The authors would like to thank three anonymous reviewers for their constructive suggestions for improving the manuscript. Zheng Liu's research is supported by National Natural Science Foundation of China (No. 61702467). Zhong Xie's research is supported by National Natural Science Foundation of China (No. 41671400) and National Key Research and Development Program (No. 2018YFB0505500).

References

- Bajaj, C., Xu, G., 2003. Anisotropic diffusion of surfaces and functions on surfaces. *ACM Trans. Graph.* 22, 4–32.
- Chen, C.Y., Cheng, K.Y., 2005. A sharpness dependent filter for mesh smoothing. *Comput. Aided Geom. Des.* 22, 376–391.
- Desbrun, M., Meyer, M., Schröder, P., Barr, A.H., 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In: Proceedings of the 26th Annual Conference Computer Graphics and Interactive Techniques, pp. 317–324.
- Desbrun, M., Meyer, M., Schröder, P., Barr, A.H., 2000. Anisotropic feature-preserving denoising of height fields and bivariate data. In: Proceedings of the 2000 Graphics Interface, pp. 145–152.
- Fleishman, S., Drori, I., Cohen-Or, D., 2003. Bilateral mesh denoising. *ACM Trans. Graph.* 22, 950–953.
- He, L., Schaefer, S., 2013. Mesh denoising via ℓ_0 minimization. *ACM Trans. Graph.* 32, 1–8.
- Hildebrandt, K., Polthier, K., 2010. Anisotropic filtering of non-linear surface features. *Comput. Graph. Forum* 23, 391–400.
- Hu, Y., Jacob, M., 2012. Higher degree total variation (hdtv) regularization for image recovery. *IEEE Trans. Image Process.* 21, 2559–2571.
- Jones, T.R., Durand, F., Desbrun, M., 2003. Non-iterative, feature-preserving mesh smoothing. *ACM Trans. Graph.* 22, 943–949.
- Li, X., Zhu, L., Fu, C.W., Heng, P.A., 2018. Non-local low-rank normal filtering for mesh denoising. *Comput. Graph. Forum* 37, 155–166.
- Lu, X., Chen, W., Schaefer, S., 2017. Robust mesh denoising via vertex pre-filtering and ℓ_1 -median normal filtering. *Comput. Aided Geom. Des.* 54.
- Sun, X., Rosin, P., Martin, R., Langbein, F., 2007. Fast and effective feature-preserving mesh denoising. *IEEE Trans. Vis. Comput. Graph.* 13, 925–938.
- Tasdizen, T., Whitaker, R., Burchard, P., Osher, S., 2002. Geometric surface smoothing via anisotropic diffusion of normals. In: Proceedings of the Conference on Visualization, pp. 125–132.
- Taubin, G., 1995. A signal processing approach to fair surface design. In: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pp. 351–358.
- Taubin, G., 2001. Linear Anisotropic Mesh Filters. IBM Research Report RC22213(W0110-051). IBM TJ. Watson Research.
- Wang, C., 2006. Bilateral recovering of sharp edges on feature-insensitive sampled meshes. *IEEE Trans. Vis. Comput. Graph.* 12, 629–639.
- Wang, P.S., Liu, Y., Tong, X., 2016. Mesh denoising via cascaded normal regression. *ACM Trans. Graph.* 35, 232.
- Wang, R., Yang, Z., Liu, L., Deng, J., Chen, F., 2014. Decoupling noise and features via weighted ℓ_1 -analysis compressed sensing. *ACM Trans. Graph.* 33, 18.
- Wang, Y., Yang, J., Yin, W., Zhang, Y., 2008. A new alternating minimization algorithm for total variation image reconstruction. *SIAM J. Imaging Sci.* 1, 248–272.
- Wu, C., Tai, X.C., 2010. Augmented Lagrangian method, dual methods, and split Bregman iteration for rof, vectorial tv, and high order models. *SIAM J. Imaging Sci.* 3, 300–339.
- Wu, C., Zhang, J., Duan, Y., Tai, X.C., 2012. Augmented Lagrangian method for total variation based image restoration and segmentation over triangulated surfaces. *J. Sci. Comput.* 50, 145–166.
- Wu, X., Zheng, J., Cai, Y., Fu, C.W., 2015. Mesh denoising using extended rof model with ℓ_1 fidelity. *Comput. Graph. Forum* 34, 35–45.
- Xu, L., Lu, C., Xu, Y., Jia, J., 2011. Image smoothing via ℓ_0 gradient minimization. *ACM Trans. Graph.* 30, 174.
- Yadav, S., Reitebuch, U., Polthier, K., 2018a. Mesh denoising based on normal voting tensor and binary optimization. *IEEE Trans. Vis. Comput. Graph.* 24, 2366–2379.
- Yadav, S., Reitebuch, U., Polthier, K., 2018b. Robust and high fidelity mesh denoising. *IEEE Trans. Vis. Comput. Graph.*, 1.
- Yagou, H., Ohtake, Y., Belyaev, A., 2002. Mesh smoothing via mean and median filtering applied to face normals. In: Geom. Model. and Proc., 2002. Proceedings, pp. 124–131.
- Yang, J., Yin, W., Yin, Z., Wang, Y., 2009. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM J. Imaging Sci.* 2, 569–592.
- Zhang, H., Wu, C., Zhang, J., Deng, J., 2015a. Variational mesh denoising using total variation and piecewise constant function space. *IEEE Trans. Vis. Comput. Graph.* 21, 873–886.
- Zhang, J., Deng, B., Hong, Y., Peng, Y., Qin, W., Liu, L., 2018. Static/dynamic filtering for mesh geometry. *IEEE Trans. Vis. Comput. Graph.*, 1–14.
- Zhang, W., Deng, B., Zhang, J., Bouaziz, S., Liu, L., 2015b. Guided mesh normal filtering. *Comput. Graph. Forum* 34, 23–34.
- Zheng, Y., Fu, H., Au, O.C., Tai, C.L., 2011. Bilateral normal filtering for mesh denoising. *IEEE Trans. Vis. Comput. Graph.* 17, 1521–1530.