Sparsity-aided Variational Mesh Restoration

Sparsity-aided Variational Mesh Restoration *

Martin Huska, Serena Morigi, and Giuseppe Antonio Recupero

Department of Mathematics, University of Bologna, Bologna BO 40126, Italy
martin.huska@unibo.it, serena.morigi@unibo.it,
giuseppe.recupero@studio.unibo.it

Abstract. We propose a variational method for recovering discrete surfaces from noisy observations which promotes sparsity in the normal variation more accurately than ℓ_1 norm (total variation) and ℓ_0 pseudo-norm regularization methods by incorporating a parameterized non-convex penalty function. This results in denoised surfaces with enhanced flat regions and maximally preserved sharp features, including edges and corners. Unlike the classical two-steps mesh denoising approaches, we propose a unique, effective optimization model which is efficiently solved by an instance of Alternating Direction Method of Multipliers. Experiments are presented which strongly indicate that using the sparsity-aided formulation holds the potential for accurate restorations even in the presence of high noise.

Keywords: Non-convex optimization · Surface Denoising · Sparse Variational Formulation

1 Introduction

The goal of a surface denoising algorithm is to remove undesirable noise or spurious information on a 3D mesh, while preserving original features, including edges, creases and corners. The restored surface is a 3D mesh that represents as faithfully as possible a piecewise smooth surface, where edges appear as discontinuities in the normals.

Through time three main numerical approaches have been developed to solve the mesh denoising problem. Initially, linear/nonlinear diffusion equations were proposed in which the evolution of vertices is guided by Partial Differential Equations, see [12, 7]. In general, the isotropic/anisotropic diffusion flows are known to have a strong regularization effect, failing, therefore, in the accurate recovery of sharp mesh features, despite the expedients proposed to preserve the local curvature of the mesh, [8]. More recently, also thanks to the considerable impact in the image processing field, two major challenges have emerged for mesh denoising: data driven and optimization-based methods. Approaches belonging to the former class aim to learn the relationship between noisy geometry and the ground-truth geometry from a training dataset, [13]. Optimization-based mesh denoising methods formulate the mesh restoration as a minimization problem

* Research is supported in part by INDaM-GNCS research project 2020.

and seeks for a denoised mesh that can best fit to the input mesh while satisfying a prior knowledge of the ground-truth geometry and noise distribution. These approches grew their popolarity more and more also thanks to the last sparsity-inducing extraordinary results. This work belongs to this latter class of optimization-based methods.

Assuming an observed noisy triangulated surface with vertex set V^0 be corrupted by additive white Gaussian noise, an estimate $V^* \in \mathbb{R}^{n_V \times 3}$ of the noisy-free vertex set V can be obtained as a solution of the following variational model

$$V^* \in \arg\min_{V \in \mathbb{R}^{n_V \times 3}} \mathcal{J}(V), \qquad \mathcal{J}(V) := \frac{\lambda}{2} \|V - V^0\|_2^2 + \mathcal{R}(V)\,, \tag{1}$$

where $\|v\|_2$ denotes the Frobenius norm of matrix v. $\mathcal{J}(V)$ is the sum of a regularization term $\mathcal{R}(V)$ and a convex smooth (quadratic) fidelity term, defined by the classical regularization parameter λ that controls the trade-off between fidelity to the observation and regularity in the solutions V^* of (1). The regularizer $\mathcal{R}(V)$ encodes a priori knowledge on the solution. In particular, to promote solutions that have piecewise constant normals with sharp discontinuities in the normal map the regularizer can be designed in a way to penalize a measure of the "roughness" or bumpiness (curvature) of a mesh, or, equivalently, to promote sparsity on this measure. A natural bumpiness measure for a surface is the normal deviations, represented by $y_i := \|(\nabla N)_i\|_2$, with $\nabla \in \mathbb{R}^{n_E \times n_T}$ a gradient (linear) operator and $(\nabla N)_i$ the normal variation between two adjacent triangles sharing the i-th edge. The ideal regularizer to induce sparsity on the vector y is the ℓ_0 pseudo-norm, but its combinatorial nature makes the minimization of (1) an NP-hard problem. Nevertheless, in [2] ℓ_0 optimization is directly applied to denoise mesh vertices and in [11] a similar strategy is applied to smooth point clouds. However, even under small amounts of noise and/or with non-uniformly shaped triangles, the strong effect of ℓ_0 can produce spurious overshoots and fold-backs, and the method becomes extremely computationally inefficient. The alternative ℓ_1 norm is the convex relaxation of the ℓ_0 pseudo-norm, and plays a fundamental role in sparse image/signal processing. In [1], ℓ_1-sparsity has been adopted to denoise point sets in a two-phase minimization strategy. However, the ℓ_1 norm tends to underestimate high-amplitude values, thus struggling in the recovery under high-level noise and presents undesired staircase and shrinkage effects. A substantial amount of recent works has argued for classes of sparsity-promoting parametrized nonconvex regularizers in favor of their superior theoretical properties and excellent practical performances [9, 5]. In this direction, the Minimax Concave (MC) penalty $\phi(\cdot; a) : \mathbb{R} \to \mathbb{R}$ see [4], provides a recognized alternative to the ℓ_1 norm and this motivated us to use it in the construction of our regularizer $\mathcal{R}(V)$. The parameter a allows to tune the degree of non-convexity, such that $\phi(\cdot; a)$ tends to ℓ_0 pseudo-norm for $a \to \infty$. The proposed regularizer $\mathcal{R}(V)$ controls sparsity of the normal deviation magnitudes more accurately than the ℓ_1 norm, while mitigating the strong effect and the numerical difficulties of ℓ_0 pseudo-norm. It can handle higher level noise than [2], produce better shaped triangles, while faithful recovering straight or smoothly curved edges.

Most of the variational mesh denoising approaches, split the process into two optimization phases - the normal smoothing phase followed by a vertex position update. The second phase suffers from foldovers problems and normal inconsistency, [15]. We propose a one-phase, effective, sparse variational model to directly smooth vertices, while preserving sharp features, keeping the normal consistency, and reducing foldovers problems. An efficient algorithm for minimizing the (non-convex) formulation is proposed which is based on the Alternating Direction Method of Multipliers (ADMM). Numerical experiments show the effectiveness of the proposed method for the solution of several mesh denoising examples.

2 Sparsity-aided Variational Model

Let us assume a surface embedded in $\mathbb{R}^3$, which is approximated by a triangulated mesh (V, T, E), where $V \in \mathbb{R}^{n_V \times 3}$, $V = \{v_i\}_{i=1}^{n_V}$ represents the set of vertices, $T \in \mathbb{R}^{n_T \times 3}$, $T = \{\tau_m\}_{m=1}^{n_T}$ is the set of triangles and $E \in \mathbb{R}^{n_E \times 2}$, $E = \{e_j\}_{j=1}^{n_E}$ is the set of edges. We denote the first disk, i.e. triangle neighbors of a vertex v_i, by $\mathcal{D}(v_i) = \{\tau_m \mid v_i \in \tau_m\}$. Let $\mathcal{N}(V) : \mathbb{R}^{n_V \times 3} \to \mathbb{R}^{n_T \times 3}$ be the mapping that computes the piecewise-constant normal field over the mesh, where the m-th element being the outward unit normal at face $\tau_m = (v_i, v_j, v_k)$, defined as

$$\mathcal{N}_m(V) := \left(\frac{(v_j - v_i) \times (v_k - v_i)}{\|(v_j - v_i) \times (v_k - v_i)\|_2} \right)^T \in \mathbb{R}^3 , \qquad m = 1, \ldots, n_T . \tag{2}$$

Focusing on the recovery of surfaces characterized by piecewise constant normals with sharp discontinuities in the normal map, we propose the following sparsity-inducing variational model to determine solutions V^* which are close to the noisy data V^0 according to the observation model and, at the same time, for which the vector of components $y_i^* = \|(\nabla N^*)_i\|_2$, $i = 1, \cdots, n_E$, is sparse

$$V^* \in \arg \min_{V \in \mathbb{R}^{n_V \times 3}} \mathcal{J}(V; \lambda, a) \tag{3}$$

$$\mathcal{J}(V; \lambda, a) := \frac{\lambda}{2} \|V - V^0\|_2^2 + \sum_{j=1}^{n_E} \phi \left(\left\| (\nabla \mathcal{N}(V))_j \right\|_2 ; a \right) .$$

At the aim to construct a parameterized sparsity-promoting regularizers characterized by tunable degree of non-convexity $a \in \mathbb{R}_+$, the function $\phi(t; a)$ is chosen among the wide class of parameterized, scalar, non-convex penalty functions, which mimic the asymptotically constant behaviour of the ℓ_0 pseudo-norm. In particular we consider one of the most effective representatives, the so-called minimax concave (MC) penalty function, $\phi(t; a) : \mathbb{R} \to \mathbb{R}$, defined as

$$\phi(t; a) = \begin{cases} -\dfrac{a}{2} t^2 + \sqrt{2a}\, t & \text{for } t \in \left[0, \sqrt{2/a} \right) , \\ 1 & \text{for } t \in \left[\sqrt{2/a}, +\infty \right) \end{cases} \tag{4}$$

which, for any value of the parameter a, satisfies the following assumptions:

- $\phi(t;a) \in \mathcal{C}^0(\mathbb{R}) \cap \mathcal{C}^2(\mathbb{R} \setminus \{0\})$
- $\phi'(t;a) \geq 0\,, \qquad \phi''(t;a) \leq 0,\ \forall\, t \in [0,\infty) \setminus \{\sqrt{2/a}\}$
- $\phi(0;a) = 0,\ \inf_t \phi''(t;a) = -a.$

The proposed non-convex penalty plays a key role in controlling, by the parameter $a > 0$, the normal variation more accurately than total variation (TV) and ℓ_0 regularizations, and induces sparsity more effectively.

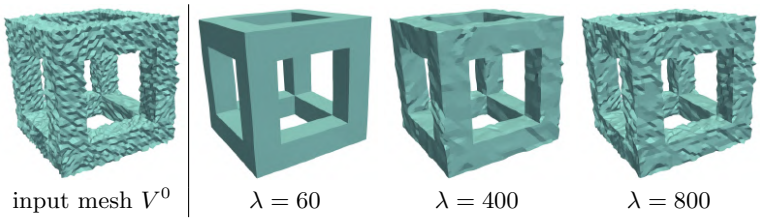

Fig. 1. Example 1: influence of the fidelity parameter λ for fixed $a = 0.8$.

Finally, we introduce the discretization of the gradient operator on the 3D mesh. Since the normal field is piecewise-constant over the mesh triangles, the gradient operator vanishes to zero everywhere but the mesh edges along which it is constant. Therefore, the gradient operator discretization is represented by a global sparse matrix $D \in \mathbb{R}^{n_E \times n_T}$ defined as

$$D_{ij} = \begin{cases} \sqrt{l_i} & \text{if } \tau_j \bigcap \tau_k = e_i\,,\ k > j, \\ -\sqrt{l_i} & \text{if } \tau_j \bigcap \tau_k = e_i\,,\ k < j, \\ 0 & \text{otherwise}\,, \end{cases} \tag{5}$$

where $l_i = |e_i|$, $i = 1, \ldots, n_E$ represents the length of i-th edge.

In Section 3 we describe an iterative optimization algorithm to solve (3).

As the mesh topology does not change during the iterations, the matrix D can be decomposed as $D = L\bar{D}$, with $L = diag\{\sqrt{l_1}, \sqrt{l_2}, \ldots, \sqrt{l_{n_E}}\}$ being the diagonal matrix of edge lengths, updated during the iterations, and $\bar{D} \in \mathbb{R}^{n_E \times n_T}$ an edge-length independent sparse matrix.

3 Numerical solution of the sparse variational model

In this section we provide details of the Alternating Direction Method of Multipliers (ADMM)- based numerical method for the solution of the nonconvex optimization problem (3). Introducing a matrix variable $N \in \mathbb{R}^{n_T \times 3}$ with row components defined in (2), and utilizing the variable splitting technique for $t \in \mathbb{R}^{n_E \times 3}$, $t = DN$, where D is discretized as in (5), the optimization problem (3) is reformulated as

$$\{V^*, N^*, t^*\} \in \arg\min_{V,N,t} \left\{ \frac{\lambda}{2}\|V - V^0\|_2^2 + \sum_{j=1}^{n_E} \phi\left(\|t_j\|_2; a\right) \right\}, \text{ s.t. } \begin{array}{c} t = DN, \\ N = \mathcal{N}(V)\,. \end{array} \tag{6}$$

We define the augmented Lagrangian for (6) as

$$\begin{aligned} \mathcal{L}(V,N,t,\rho_1,\rho_2;\lambda,\beta_1,\beta_2,a) = \frac{\lambda}{2}\|V - V^0\|_2^2 + \\ + \sum_{j=1}^{n_E} \left[\phi\left(\|t_j\|_2; a\right) - \left\langle \rho_{1_j}, t_j - (DN)_j \right\rangle + \frac{\beta_1}{2}\|t_j - (DN)_j\|_2^2 \right] + \\ + \sum_{\substack{m=1 \\ \tau_m=(v_i,v_j,v_k)}}^{n_T} \left[-\left\langle \rho_{2_m}, N_m - \mathcal{N}_m(V) \right\rangle + \frac{\beta_2}{2}\left\|N_m - \mathcal{N}_m(V)\right\|_2^2 \right], \end{aligned} \tag{7}$$

where $\beta_1, \beta_2 > 0$ are scalar penalty parameters, and $\rho_1 \in \mathbb{R}^{n_E \times 3}$, $\rho_2 \in \mathbb{R}^{n_T \times 3}$ represent the matrices of Lagrange multipliers associated with the constraints.

We then consider the following saddle-point problem:

$$\begin{aligned} \text{Find} \quad & (V^*, N^*, t^*, \rho_1^*, \rho_2^*) \in \mathbb{R}^{n_V \times 3} \times \mathbb{R}^{n_T \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_T \times 3} \\ \text{s.t.} \quad & \mathcal{L}(V^*, N^*, t^*, \rho_1, \rho_2) \le \mathcal{L}\left(V^*, N^*, t^*, \rho_1^*, \rho_2^*\right) \le \mathcal{L}\left(V, N, t, \rho_1^*, \rho_2^*\right), \\ & \forall (V, N, t, \rho_1, \rho_2) \in \mathbb{R}^{n_V \times 3} \times \mathbb{R}^{n_T \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_T \times 3} \end{aligned} \tag{8}$$

An ADMM-based iterative scheme is applied to approximate the solution of the saddle-point problem (7)–(8). Initializing to zeros both the dual variables $\rho_1^{(0)}$, $\rho_2^{(0)}$ and $N_m^{(0)} = \mathcal{N}_m(V^{(0)})$, $m = 1, \ldots, n_T$, the k-th iteration of the proposed alternating iterative scheme reads as follows:

$$t^{(k+1)} = \arg\min_{t \in \mathbb{R}^{n_E \times 3}} \mathcal{L}(V^{(k)}, N^{(k)}, t; \rho_1^{(k)}, \rho_2^{(k)})\,, \tag{9}$$

$$N^{(k+1)} = \arg\min_{N \in \mathbb{R}^{n_T \times 3}} \mathcal{L}(V^{(k)}, N, t^{(k+1)}; \rho_1^{(k)}, \rho_2^{(k)})\,, \tag{10}$$

$$V^{(k+1)} = \arg\min_{V \in \mathbb{R}^{n_V \times 3}} \mathcal{L}(V, N^{(k+1)}, t^{(k+1)}; \rho_1^{(k)}, \rho_2^{(k)})\,, \tag{11}$$

$$\rho_1^{(k+1)} = \rho_1^{(k)} - \beta_1 \left(t^{(k+1)} - DN^{(k+1)} \right)\,, \tag{12}$$

$$\rho_2^{(k+1)} = \rho_2^{(k)} - \beta_2 \left(N^{(k+1)} - \mathcal{N}\left(V^{(k+1)} \right) \right). \tag{13}$$

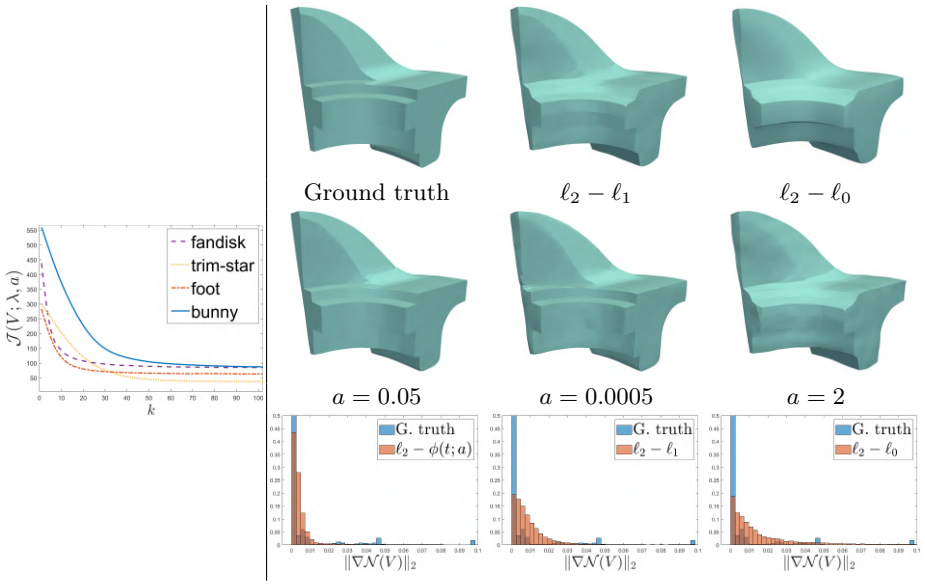

Fig. 2. Example 1: (left) empirical convergence of ADMM algorithm for some reconstructed meshes; (right) sensitivity to the penalty.

The updates of Lagrangian multipliers ρ_1 and ρ_2 have closed form, while the solutions to the remaining subproblems will be described in detail in the following sections.

Subproblem for t. Omitting the constant terms in (7), we can rewrite the subproblem (9) as

$$t^{(k+1)} = \arg\min_{t\in\mathbb{R}^{n_E\times 3}} \sum_{j=1}^{n_E} \left[\phi\left(\|t_j\|_2\,;a\right) - \left\langle \rho_{1_j}, t_j - (DN)_j \right\rangle + \frac{\beta_1}{2}\|t_j - (DN)_j\|_2^2 \right]. \tag{14}$$

Due to the separability property of $\phi(\cdot;a)$, problem (14) is equivalent to n_E three-dimensional problems for each t_j, $j=1,\ldots,n_E$ in form

$$t_j^{(k+1)} = \arg\min_{t_j\in\mathbb{R}^3} \left\{ \frac{1}{\beta_1}\phi\left(\|t_j\|_2\,;a\right) + \frac{1}{2}\|t_j - r_j^{(k+1)}\|_2^2 \right\}, \tag{15}$$

where $r_j^{(k+1)} := \left(DN^{(k)}\right)_j + \frac{1}{\beta_1}\left(\rho_1^{(k)}\right)_j$.

Necessary and sufficient conditions for strong convexity of the cost functions in (15) are demonstrated in [3]. In particular, the problems in (15) are strongly convex if and only if the following condition holds:

$$\beta_1 > a \implies \beta_1 = \varepsilon a, \quad \text{for} \quad \varepsilon > 1. \tag{16}$$

Under the assumption (16), the unique minimizers of (15) can be obtained in closed form as

$$t_j^{(k+1)} = \min(\max(\nu - \zeta/\|r_j\|_2, 0), 1)\, r_j \ ,$$

where $\nu = \dfrac{\beta_1}{\beta_1 - a}$ and $\zeta = \dfrac{\sqrt{2a}}{\beta_1 - a}$. We remark that the condition on β_1 in (7) only ensures the convexity conditions (16) of t-subproblem (15), but does not guarantee convergence of the overall ADMM scheme.

Subproblem for N. Gathering the non-constant terms w.r.t. N in (7), we can reformulate (10) as

$$N^{(k+1)} = \arg\min_{N \in \mathbb{R}^{n_T \times 3}} \left\{ \frac{\beta_1}{2} \|t^{(k+1)} - DN\|_2^2 + \left\langle \rho_1^{(k)}, DN \right\rangle - \left\langle \rho_2^{(k)}, N \right\rangle + \frac{\beta_2}{2} \left\| N - \mathcal{N}\left(V^{(k)}\right) \right\|_2^2 \right\} \ ,$$

for which the first optimality conditions lead to the following three linear systems, one for each spatial coordinate of $N \in \mathbb{R}^{n_T \times 3}$

$$\left(D^T D + \frac{\beta_2}{\beta_1} I \right) N = \frac{\beta_2}{\beta_1} \mathcal{N}\left(V^{(k)}\right) + \frac{\rho_2^{(k)}}{\beta_1} + D^T \left(t^{(k+1)} - \frac{1}{\beta_1} \rho_1^{(k)} \right) \ . \tag{17}$$

Since $\beta_1, \beta_2 > 0$, the linear system coefficient matrix is sparse symmetric, positive definite and identical for all three coordinate vectors, therefore, the system can be solved by applying a unique Cholesky decomposition. At each iteration, the edge lengths diagonal matrix L in $D = L\bar{D}$, needs to be updated as the vertices V move to their new position. For large meshes, an iterative solver warm-started with the solution of the last ADMM iteration, is rather preferred. A normalization is finally applied as N represents the normal field.

Subproblem for V. Omitting the constant terms in (7), the subproblem for V reads as

$$V^{(k+1)} = \arg\min_{V \in \mathbb{R}^{n_V \times 3}} \{\mathcal{J}_V(V)\}$$

$$\mathcal{J}_V(V) = \tfrac{\lambda}{2} \|V - V^0\|_2^2 + \textstyle\sum_{m=1}^{n_T} \left[\left\langle \rho_{2_m}^{(k)}, \mathcal{N}_m(V) \right\rangle + \tfrac{\beta_2}{2} \left\| N_m^{(k+1)} - \mathcal{N}_m(V) \right\|_2^2 \right] . \tag{18}$$

The functional $\mathcal{J}_V(V)$ is proper, smooth, non-convex and bounded from below by zero. A minimum can be obtained applying the gradient descent algorithm with backtracking satisfying the Armijo condition or using the BFGS method. For the experimental section we used the gradient descent algorithm for efficiency. The partial derivative of (18) w.r.t vertex $v_i \in V$, $i = 1, \ldots, n_V$ reduces the sum in (18) to the sum over the first disk $\mathcal{D}(v_i)$ and is given as

$$\nabla_{v_i}\mathcal{J}_V(V) = \lambda(v_i - v_i^0) + \sum_{\substack{\tau_m\in\mathcal{D}(v_i)\\ \tau_m=(v_i,v_j,v_k)}} \left[\frac{\left(\rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)}\right)\times(v_k - v_j)}{\|(v_j - v_i)\times(v_k - v_i)\|_2} \right.$$
$$\left. - \frac{\left\langle \rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)}, (v_j - v_i)\times(v_k - v_i)\right\rangle [(v_j - v_i)\times(v_k - v_i)\times(v_k - v_j)]}{\|(v_j - v_i)\times(v_k - v_i)\|_2^3} \right], \tag{19}$$

which simplifies as follows, for all triangles $m = 1, \ldots, n_T$,

$$\nabla_{v_i}\mathcal{J}_V(V) = \lambda(v_i - v_i^0) +$$
$$\sum_{\tau_m\in\mathcal{D}(v_i)} \frac{\left[\left(\rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)}\right) - \left\langle \rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)}, \mathcal{N}_m(V)\right\rangle \mathcal{N}_m(V)\right]}{2s_{\tau_m}} \times (v_k - v_j),$$

with $s_{\tau_m} := \|(v_j - v_i)\times(v_k - v_i)\|_2/2$ the area of triangle τ_m with updated vertices in V, and $\mathcal{N}_m(V) = ((v_j - v_i)\times(v_k - v_i))/(2s_{\tau_m})$.

The convergence of our proposed three block ADMM scheme is not easy to derive relying on the results presented so far, see [14]. However, we will provide some evidence of the numerical convergence in the experimental section.

Many two-phase mesh denoising algorithms present the normal orientation ambiguity problem in the vertex updating phase, which provokes ambiguous shifts of the vertex position due to direction inconsistency of the normal vectors [16, 10]. In [6] this issue is solved by an orientation aware vertex updating scheme which considers the parallelism of the normal determined by triangle vertices to a given normal vector.

Proposition 1. *The reconstructed normal map N^* obtained by solving* (6) *via the proposed ADMM, satisfies the orientation consistency.*

Proof. The vertex update is computed by solving the ADMM sub-problem (18). The second term in function (18) induces for each triangle τ_m the orthogonality between the triangle normal $\mathcal{N}_m$ and ρ_2, that is, it minimizes the volume of the parallelepiped defined by the edges ρ_2, $(v_j - v_i)$ and $(v_k - v_i)$ thus imposing ρ_2 to lie in the triangle plane generated by $(v_j - v_i)$ and $(v_k - v_i)$. The third term in the objective function (18) penalizes the discrepancy between the restored normal ($N_m^{(k+1)}$) and the triangle face normal obtained by its updated vertices, both in orientation and in magnitude.

4 Numerical Examples

We validate the proposed method both qualitatively and quantitatively on a variety of benchmark triangulated surfaces characterized by different sharpness

and smoothness features. The noisy meshes have been synthetically corrupted following the degradation model

$$v_i^0 = v_i + c_i d_i \ , \qquad i = 1, \ldots, n_V \ , \tag{20}$$

where $c_i \in \mathbb{R}$ is Gaussian noise distributed with zero mean and standard deviation $\sigma = \gamma \bar{l}$ where $\bar{l}$ is the average edge length and $\gamma \geq 0$ represents the noise level. The vectors d_i, $i = 1, \ldots, n_V$ can be random directions or the vertex normal itself. All the meshes are rendered in flat-shading model.

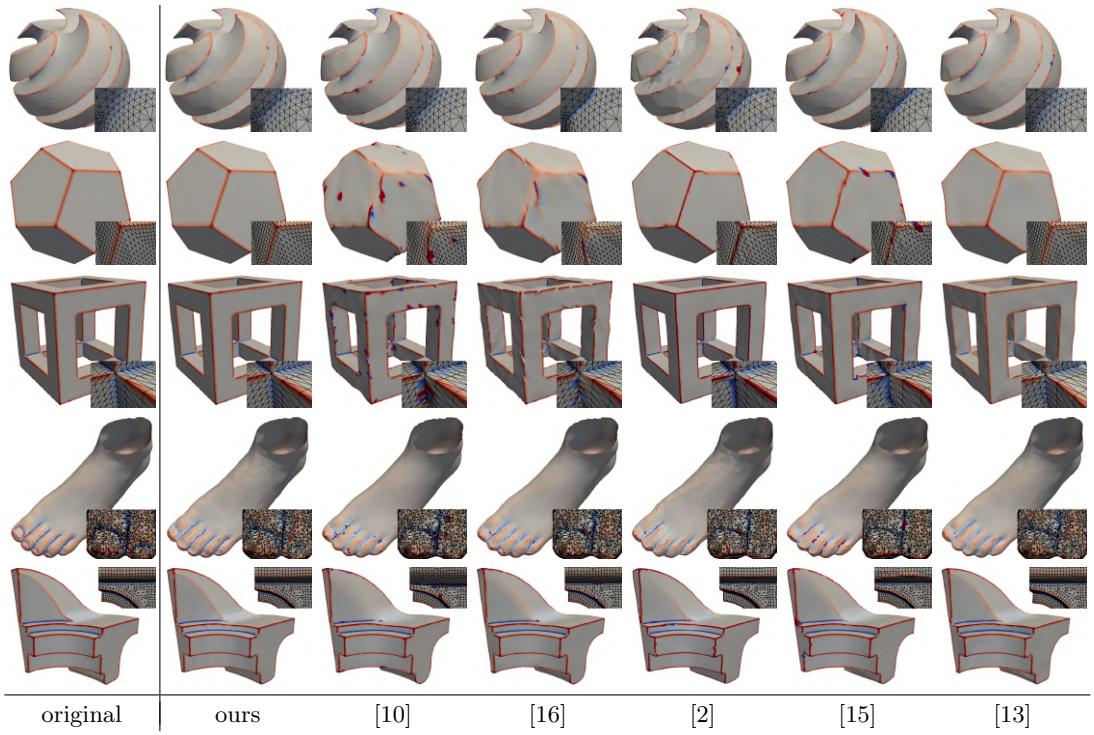

Fig. 3. Example 2: Denoising results from input meshes corrupted by noise levels $\gamma = \{0.15, 0.3, 0.3, 0.2, 0.2\}$, from top to bottom.

Example 1 presents various aspects of our algorithm, and illustrates the sparsity-promoting benefits introduced by the penalty $\phi(\cdot; a)$ with respect to ℓ_1 (TV) and ℓ_0 penalty terms. In Example 2 the performance of the proposed method is compared with some other variational methods for mesh denoising, namely [10, 16, 2, 15], which have been kindly provided by authors of [15] at `https://github.com/bldeng/GuidedDenoising`, and a learning-based approach [13]. For each method, we show their best results we achieved by tuning the corresponding set of parameters.

The quantitative evaluation regards the following error metrics, which measure the discrepancy of the computed V^*, N^* to the noisy-free mesh V_{GT}, N_{GT}:

- **Mean squared angular error (MSAE)** $MSAE = \mathbb{E}[\angle(N_{GT}, N^*)^2]$,
- L_2 **vertex to vertex error** (E_V) $E_V = \frac{\|V^* - V_{GT}\|_F}{n_V}$.

For all the tests, the iterations k of the ADMM algorithm are stopped as soon as either of the two following conditions is fulfilled:

$$k > \mathrm{TH}_1 = 200\,, \qquad \left\|V^{(k+1)} - V^{(k)}\right\|_2 / \left\|V^{(k)}\right\|_2 < \mathrm{TH}_2 = 10^{-6}. \tag{21}$$

Fig. 2(left) shows the energy evolution curve in terms of the number of iterations for some of the meshes reported in this section, which returns the empirical convergence of the proposed ADMM algorithm.

Example 1 We illustrate how the model parameters λ and a influence the result quality. The value for λ depends on the amount of noise: the smaller the noise the bigger has to be λ. In Fig.1, for a fixed noise level $\gamma = 0.3$ and d_i being the normal at v_i, the amount of noise removed from the `cube_hole` mesh is less for increasing λ values. The sensitivity to the penalty function in the recovery of a corrupted `fandisk` mesh (noise level $\gamma = 0.2$) is illustrated in Fig. 2(right). In the first row the noisy free mesh is shown together with the denoised meshes obtained by the $\ell_2 - \ell_1$ and the $\ell_2 - \ell_0$ models, respectively. The results present remarkable losses of sharp features and creases. In the second row the $\ell_2 - \phi(t; a)$ model is applied with optimal $a = 5 \times 10^{-2}$, $a = 5 \times 10^{-4}$ as for $a \to 0$, $\phi(t; a)$ behaves like the ℓ_1- penalty (TV), and finally with $a = 2$, since for $a \to \infty$ the penalty approaches to ℓ_0- penalty. The value of a in $\phi(\cdot; a)$ allows to tune the degree of non-convexity and thus the degree of flatness reconstruction, while preserving features. Fig. 2 (third row) shows the histograms of the sparsified measure $\|\nabla \mathcal{N}(V)\|_2$ for our model $\ell_2 - \phi(t; a)$ (left), $\ell_2 - \ell_1$ model (center), and the $\ell_2 - \ell_0$ model (right).

Example 2 In this example we compare our method with other state of-the-art methods. Fig. 3 shows the denoised meshes colored by their mean curvature scalar map, with fixed range, and zoomed details on mesh edges. Compared to the visually better denoising results obtained by our method, remarkable overlaps appear on the other results and severe shaped triangle perturbations are introduced in the reconstructed meshes. To further demonstrate the robustness to noise, in Fig. 4 we test our sparsity-inducing variational framework on increasing levels of noise $\gamma = \{0.2, 0.3, 0.4, 0.5, 0.6\}$ from top to bottom, and in the last two rows also for arbitrary noise directions (d_i) in (20) and a real 3D scanned data, respectively. Below each synthetic result, we report the quantitative evaluations according to the error metrics ($MSAE \times 10^2$,$E_V \times 10^6$). Both quantitatively and qualitatively the results confirm the effectiveness of our sparse variational proposal. Finally, we can comment on the efficiency of our algorithm which computational time is, on average, one order less than the $\ell_2 - \ell_0$ denoising method, and comparable to the other tested methods.

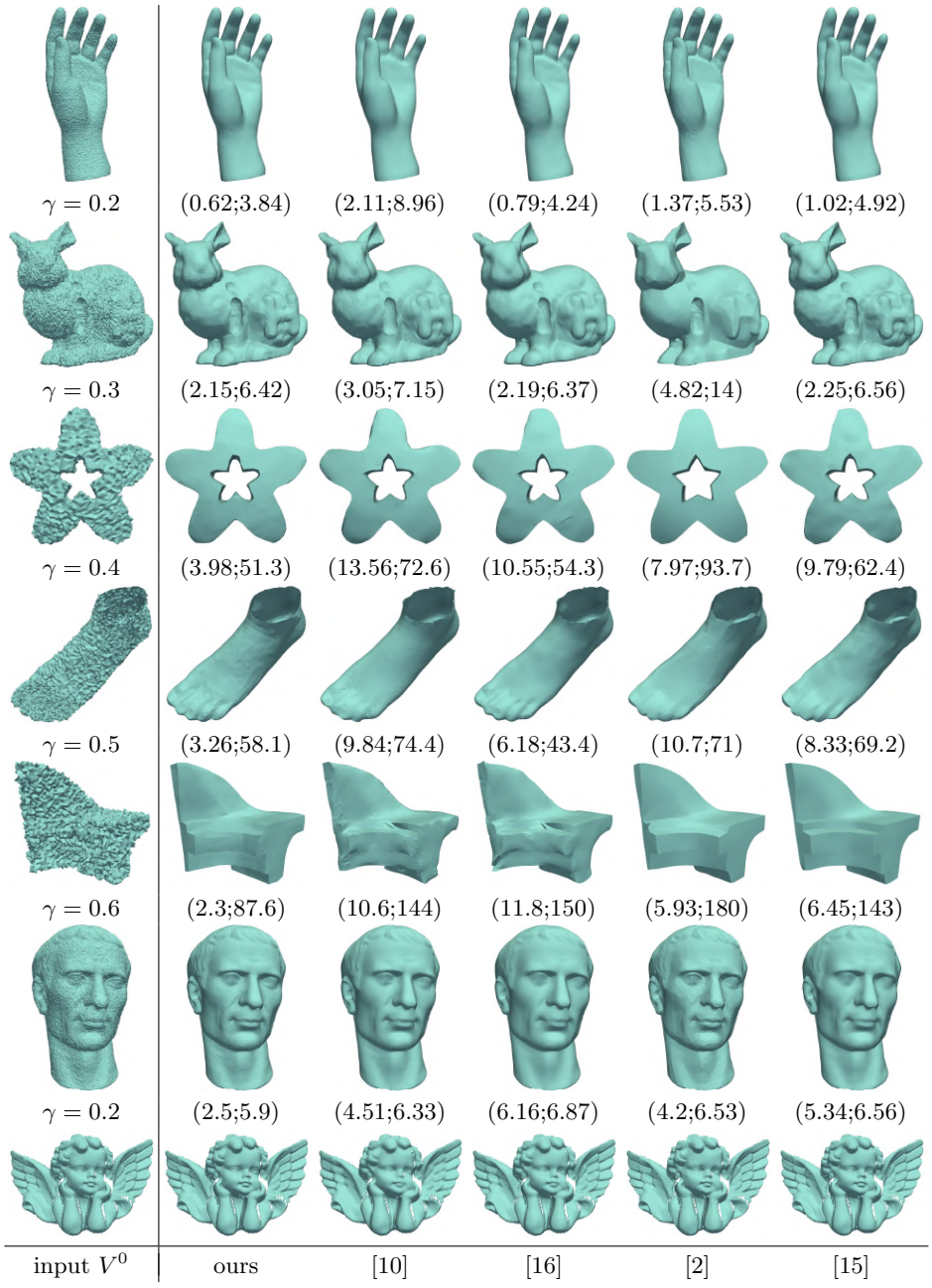

Fig. 4. Example 2: Comparison of our method with related works.

5 Conclusion

The proposed single-phase variational method is capable to restore sharp edges and creases from a noisy triangulated surface in a significantly better way than other two-phases variational methods which rely on ℓ_0 and ℓ_1 regularizers. This is achieved by introducing a parameterized sparsity inducing penalty with a parameter a which allows for promoting a fair smoothing of the normal field to be reconstructed. One of the future investigations will be aimed at a rigorous theory to derive convexity conditions in the convex non-convex framework which would lead to the well-known convex optimization benefits.

References

1. Avron, H., Sharf, A., Greif, C., Cohen-Or, D.: ℓ_1-sparse reconstruction of sharp point set surfaces. ACM Trans. Graph. **29**(5) (Nov 2010)
2. He, L., Schaefer, S.: Mesh denoising via l0 minimization. ACM Trans. Graph. **32**(4) (Jul 2013)
3. Huska, M., Lanza, A., Morigi, S., Selesnick, I.: A convex-nonconvex variational method for the additive decomposition of functions on surfaces. Inverse Problems **35**(12), 124008 (2019)
4. Huska, M., Lanza, A., Morigi, S., Sgallari, F.: Convex non-convex segmentation of scalar fields over arbitrary triangulated surfaces. J. Computational and Applied Mathematics **349**, 438–451 (2019)
5. Lanza, A., Morigi, S., Sgallari, F.: Convex image denoising via non-convex regularization. In: Aujol, J.F., Nikolova, M., Papadakis, N. (eds.) Proc. SSVM, pp. 666–677. Springer International Publishing (2015)
6. Liu, Z., Lai, R., Zhang, H., Wu, C.: Triangulated surface denoising using high order regularization with dynamic weights. SIAM J. Sci. Comp. **41**(1), B1–B26 (2019)
7. Lysaker, M., Osher, S., Xue-Cheng Tai: Noise removal using smoothed normals and surface fitting. IEEE Transactions on Image Processing **13**(10), 1345–1357 (2004)
8. Morigi, S., Rucci, M., Sgallari, F.: Nonlocal surface fairing. In: Bruckstein, A.M., ter Haar Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (eds.) SSVM. pp. 38–49. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)
9. Nikolova, M.: Energy minimization methods. In: Scherzer, O. (ed.) Handbook of Mathematical Methods in Imaging, chap. 5, pp. 138–186. Springer (2011)
10. Sun, X., Rosin, P.L., Martin, R., Langbein, F.: Fast and effective feature-preserving mesh denoising. IEEE Trans. on Vis. and Comp. Graph. **13**(5), 925–938 (2007)
11. Sun, Y., Schaefer, S., Wang, W.: Denoising point sets via l0 minimization. Computer Aided Geometric Design **35-36**, 2 – 15 (2015)
12. Tasdizen, T., Whitaker, R., Burchard, P., Osher, S.: Geometric surface processing via normal maps. ACM Trans. Graph. **22**(4), 1012–1033 (Oct 2003)
13. Wang, P.S., Liu, Y., Tong, X.: Mesh denoising via cascaded normal regression. ACM Trans. Graph. **35**(6) (Nov 2016)
14. Wang, Y., Yin, W., Zeng, J.: Global convergence of admm in nonconvex nonsmooth optimization. J Sci Comput **78**(12), 29–63 (2019)
15. Zhang, W., Deng, B., Zhang, J., Bouaziz, S., Liu, L.: Guided mesh normal filtering. Comput. Graph. Forum **34**(7), 23–34 (Oct 2015)
16. Zheng, Y., Fu, H., Au, O.K., Tai, C.: Bilateral normal filtering for mesh denoising. IEEE Trans. on Visualization and Computer Graphics **17**(10), 1521–1530 (2011)