



# A Unified Surface Geometric Framework for Feature-Aware Denoising, Hole Filling and Context-Aware Completion

Luca Calatroni<sup>1</sup> · Martin Huska<sup>2</sup> · Serena Morigi<sup>2</sup> · Giuseppe Antonio Recupero<sup>1,2</sup>

Received: 30 July 2021 / Accepted: 18 May 2022 / Published online: 22 June 2022  
© The Author(s) 2022

## Abstract

Technologies for 3D data acquisition and 3D printing have enormously developed in the past few years, and, consequently, the demand for 3D virtual twins of the original scanned objects has increased. In this context, feature-aware denoising, hole filling and context-aware completion are three essential (but far from trivial) tasks. In this work, they are integrated within a geometric framework and realized through a unified variational model aiming at recovering triangulated surfaces from scanned, damaged and possibly incomplete noisy observations. The underlying non-convex optimization problem incorporates two regularisation terms: a discrete approximation of the Willmore energy forcing local sphericity and suited for the recovery of rounded features, and an approximation of the  $\ell_0$  pseudo-norm penalty favouring sparsity in the normal variation. The proposed numerical method solving the model is parameterization-free, avoids expensive implicit volume-based computations and based on the efficient use of the Alternating Direction Method of Multipliers. Experiments show how the proposed framework can provide a robust and elegant solution suited for accurate restorations even in the presence of severe random noise and large damaged areas.

**Keywords** Variational surface restoration · Willmore energy · Sparse non-convex optimization · Surface denoising · Surface inpainting · Context-aware mesh completion

## 1 Introduction

In spite of the remarkable progresses achieved in the fields of 3D scanning and 3D printing technologies, the available digitizing techniques often produce defective data samples corrupted by random noise and often subject to a local lack of data. Typically, this is mainly due to occlusions, surface reflection, scanner placement constraints, etc. In the context of digital restoration of cultural heritage art-works, for

instance, the scanned object itself (e.g. the archaeological findings) may be incomplete and damaged due to the fact that some of its (missing) parts have been ruined over time due to wear and tear. In these cases, to facilitate the downstream processing of its digital content, the object shape needs to be denoised and repaired. Generally speaking, desirable properties of a surface repair toolkit shall include:

- *Feature-aware denoising*: the removal of undesirable noise or spurious information from the data, while preserving original features, including edges, creases and corners, with special care on the robustness for defective and incomplete point sets.
- *Smooth hole filling inpainting*: the process of recovering a missing or damaged region in the surface by filling it in a plausible way using available information. The result of a surface-inpainting operation depends on specific application considered. In digital cultural heritage restoration, for instance, surface inpainting is understood as the recovering of the holes in the data or the removal of the scratches/cracks possibly present in the scanned objects. In prototype manufacturing, the goal is shifted

---

✉ Serena Morigi  
serena.morigi@unibo.it

Luca Calatroni  
calatroni@i3s.unice.fr

Martin Huska  
martin.huska@unibo.it

Giuseppe Antonio Recupero  
giuseppe.recupero3@unibo.it

<sup>1</sup> Laboratory I3S, UCA, CNRS, INRIA,  
2000 Route des Lucioles, 06903 Sophia-Antipolis, France

<sup>2</sup> Department of Mathematics, University of Bologna,  
Piazza di Porta San Donato 5, 40126 Bologna, Italy

towards a waterproof virtual reconstruction, so that the related operation is rather interpreted as a smooth hole filling. Either case, all damaged areas should be filled in a seamless way that is minimally distinguishable from their surrounding regions.

- *Context-aware completion:* when a priori knowledge on the missing/damaged parts of the scanned model is known, it is desirable that the completion of the damaged areas occurs by pasting known data—such as template patches—automatically or semi-automatically under user guidance. This allows, for example, to repair a damaged part of an artefact by filling the region of interest with a patch taken from a valid/undamaged region of the model itself or even from other 3D geometric models.

An example of the three geometric tasks processed by the proposed geometric variational framework is illustrated in Fig. 1. The original noisy and incomplete scanned *angel* mesh (see Fig. 1(left)) is denoised while keeping all the holes, see Fig. 1(center, first row). Then, the inpainting tool filled the holes smoothly, as shown in Fig. 1(center, second row) driven by the inpainting mask illustrated on the left. The large damaged region on the head is recovered by replacing a hair curl patch selected from a different, undamaged, mesh, see the recovered mesh in Fig. 1(center, third row). Finally, the completion of the damaged part together with hole filling is performed and illustrated in Fig. 1(right).

We propose a unified approach for these challenging geometric tasks by defining a variational problem encoding *a priori* knowledge of the particular problem (i.e. the mask operators) directly in the cost functional.

Here, it is assumed that a corrupted surface  $S$  embedded into  $\mathbb{R}^3$  and possibly characterized by the presence of a damaged (incomplete) region  $S_D \subset S$ , is represented by a

triangulated mesh  $\mathcal{M}_0 = (V_0, T_0)$  with  $V_0 \in \mathbb{R}^{n_V \times 3}$  being the set of  $n_V$  vertices, and  $T_0 \in \mathbb{R}^{n_T \times 3}$  being the set of  $n_T$  triangle faces, sharing  $n_E$  edges.

The aforementioned three geometry processing tasks are addressed by means of the following unified variational formulation

$$V^* \in \arg \min_V \mathcal{J}(V; M_E),$$

$$\mathcal{J}(V; M_E) := \lambda \chi_{S \setminus S_D} \mathcal{F}(V; V_0) + \mathcal{R}_1(V; M_E) + \mathcal{R}_2(V; M_E^c), \tag{1}$$

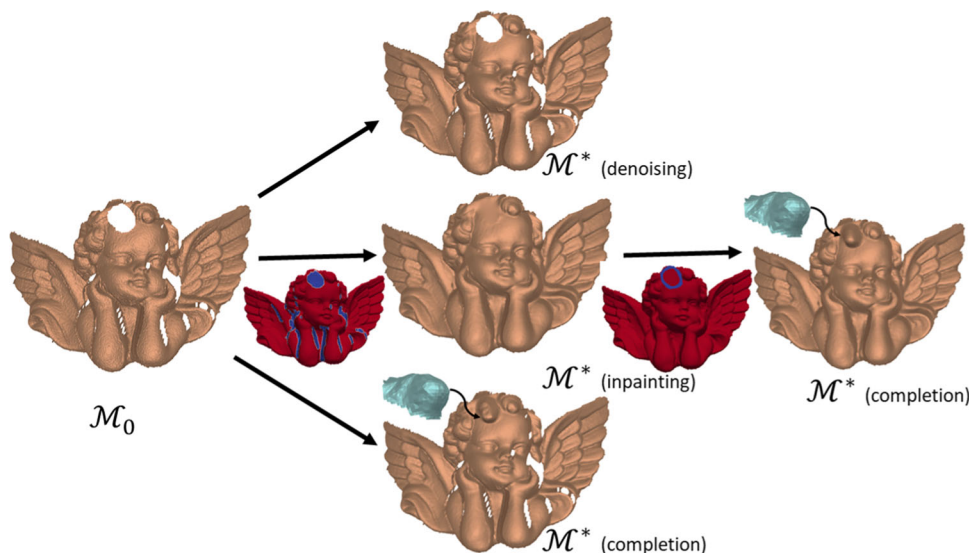
where  $\chi_{S \setminus S_D} : S \rightarrow \{0, 1\}$  denotes the characteristic function of the subset  $S \setminus S_D$ , while the binary mask operators  $M_E \in \{0, 1\}^{n_E}$  and  $M_E^c = \mathbf{1}_{n_E} - M_E$  characterize the specific surface geometry considered. As a result of the discretization on the triangulated mesh, the role of the characteristic function is played by a mask operator  $M_V \in \{0, 1\}^{n_V}$  whose zero values identify the region  $S_D$ .

The set of vertices  $V^*$  solution of the unconstrained optimization problem (1) defines a restored triangulated surface  $\mathcal{M}^* = (V^*, T^*)$  which provides a solution of the three surface geometry tasks, depending on the particular setup considered.

In case of surface completion, the region identified by  $S_D$  is replaced by a given template patch  $\mathcal{P}$  with boundary  $b_{\mathcal{P}}$ . In this case, the only compatibility assumptions required is that the boundary of  $S_D$  in  $\mathcal{M}_0$ , named  $b_0$ , and  $b_{\mathcal{P}}$  have the same number of vertices. If this is not the case, a suitable subdivision process can be preliminarily applied. The template patch  $\mathcal{P}$  can be identified on the object itself as well as on other objects, thus allowing a mesh editing process.

The proposed approach does not need any global or even local 2D parameterization, nor any sophisticated octree data structures to efficiently solve implicit volumetric compu-

**Fig. 1** Applications of the proposed surface geometry framework: incomplete and noisy surface input  $\mathcal{M}_0$  (left); denoised surface  $\mathcal{M}^*$  (center, first row), inpainting mask  $M_V$  and inpainting result  $\mathcal{M}^*$  (center, second row), context-aware completion of the curly hair detail without inpainting (center, third row); context-aware completion result  $\mathcal{M}^*$  with preliminary inpainting (right). The  $S_D$  region is represented in blue in the masks



tations [1,2]. The data are explicitly treated as connected samples of a surface embedded in  $\mathbb{R}^3$ .

The functional  $\mathcal{J}(V; M_E)$  in (1) is characterized by the presence of the sum of two regularization terms: the sparsity-promoting term  $\mathcal{R}_1(V; M_E)$  and the sphericity-inducing penalty  $\mathcal{R}_2(V; M_E^c)$ . Furthermore, a fidelity term  $\mathcal{F}(V; V_0)$ , weighted by the scalar parameter  $\lambda \geq 0$ , is used to control the trade-off between fidelity to the observations and regularity in the solution  $V^*$  of (1).

The regularizer  $\mathcal{R}_1$  favours solutions with piece-wise constant normal map and sharp discontinuities. Such 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 deviation measuring the normal variation between adjacent triangles.

The ideal sparse-recovery term one would like to consider is the non-convex, non-continuous  $\ell_0$  pseudo-norm, but its combinatorial nature makes the minimization of (1) an NP-hard problem. We then rather consider as regularizer  $\mathcal{R}_1(V; M_E)$  a sparsity-promoting parametrized non-convex term, whose form provides an effective control on the sparsity of the normal deviation magnitudes being more accurate than the  $\ell_1$  norm, while mitigating the strong effect and the numerical difficulties of  $\ell_0$  pseudo-norm. Numerical experiments will show its efficiency in handling high levels of noise, producing good-shaped triangles and faithfully recovering straight and smoothly curved edges.

As far as the  $\mathcal{R}_2$  regularization term is concerned, we choose it so as to encode a geometric energy, aimed to force local sphericity in correspondence of rounded regions. We considered here the Willmore energy, which has to be preferred over standard approaches based on mean curvature flow due to its scale invariance nature. Such energy is a quantitative measure of how much a given surface deviates from a round sphere, and it is defined by the following curvature functional

$$E_w(S) = \frac{1}{2} \int_S (h^2 - k) dA, \quad (2)$$

where  $dA$  is the area element, and  $h$  and  $k$  are the mean and Gaussian rigidities, respectively. The Willmore energy  $E_w(S)$  is non-negative and vanishes if and only if  $S$  is a sphere [3]. For compact and closed surfaces, and surfaces whose boundary is fixed up to first order, i.e. positions and normals are prescribed, finding the minima of (2) is equivalent to minimize the Willmore bending energy  $E_h(S) = \frac{1}{2} \int_S h^2 dA$  since the two functionals differ only by a constant (the Euler characteristic of the surface  $S$ ), [4]. In this paper, we present a discrete Willmore energy, which, in contrast to traditional approaches, follows an edge-based discrete formulation.

Compared to the earlier version of this work [5] which focused only on surface denoising, this work further integrates the Willmore energy term in (1) to promote fairness and extends the model usability to more tasks.

From an algorithmic point of view, we solve the (non-convex) problem (1) by means of an Alternating Direction Method of Multipliers (ADMM) scheme. This allows us to split the minimization problem into three more tractable sub-problems. Closed-form solutions for two of these problems can be found, while for the third, non-convex, one different optimization solvers can be used. For this sub-step, we compare standard gradient descent, with heavy ball and Broyden–Fletcher–Goldfarb–Shanno (BFGS) schemes, endowed with suitable backtracking strategy applied to guarantee the convergence to stationary points of the sub-problem considered.

Numerical experiments will demonstrate the effectiveness of the proposed method for the solution of several exemplar mesh denoising, inpainting and completion problems.

The rest of the paper is organized as follows. In Sect. 2, we review the main mathematical approaches related to the three geometry tasks considered. In Sect. 3, the proposed geometric variational model is presented; details on its numerical optimization by means of the ADMM-based scheme are described in Sect. 4. In Sect. 5, we briefly discuss the details on how each of the three task can be realized by solving the optimization problem (1). Experimental results and comparisons are given in Sect. 6. We draw the conclusion in Sect. 7.

## 2 Related Works

Standard numerical approaches solving the mesh denoising problem can be, essentially, divided into three classes. The first class inherits PDE-based techniques from analogous problems arising in image processing and addresses the task by using linear/nonlinear diffusion equations, see, e.g. [6,7], with particular care to preserve local curvature features [8]. Recently, also thanks to their considerable impact in the image processing field, two further major approaches have started to be investigated: 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, see, e.g. [9]. Optimization-based mesh denoising methods formulate the mesh restoration problem as a minimization problem where a denoised mesh best fitting to the input mesh while satisfying a prior knowledge of the ground-truth geometry and noise distribution is sought. These approaches grew their popularity more and more also thanks to rapid development of studies on sparsity-inducing penalties. Among them, penalty terms aimed at approximating the  $\ell_0$  pseudo-norm have been directly applied for denoising mesh vertices in [10] and noisy

point clouds in [11]. However, the strong geometric bias favoured by the use of the  $\ell_0$  pseudo-norm can produce spurious overshoots and fold-backs; hence, these methods may become computationally inefficient, even under small amounts of noise. Alternatively, an  $\ell_1$  penalty can be used. This is very frequent in recent sparse image/signal processing problems as well as in the mesh processing community, see, e.g. [12], where  $\ell_1$ -sparsity was adopted to denoise point sets in a two-phase minimization strategy. As it is well-known, the  $\ell_1$  norm tends to underestimate high-amplitude values, thus struggling in the recovery under high-level noise and presenting undesired staircase and shrinkage artefacts.

In this work, we focus on an optimization-based approach for mesh denoising and exploit a non-convex penalty approximating the  $\ell_0$  pseudo-norm so as to induce sparsity without artefacts and promoting fairness.

As far as the hole filling or repairing problem is concerned, standard approaches range from fourth-order surface diffusion PDE methods [13], to volumetric approaches mainly based on signed distance functions to implicitly represent the surface [14,15]. Other popular non-polygonal methods rely on Radial Basis Functions implicit interpolations [16], and Moving Least Squares [17]. A commonly adopted fairness prior is  $\|\Delta V\|_2^2$ , proposed for smooth hole filling with the so-called least squares meshes, see [18].

Our penalty function also favours smoothness. However, unlike least squares meshes, we adopt a nonlinear curvature measure—the Willmore energy—which leads to a more rounded shape filling.

When some a priori knowledge on the missing part is available, we can do more than simply fill the hole as we can complete/repair the hole geometry with a context-aware template patch, with a minimally distinguishable transition zone. Many efforts have been devoted to the automatic selection of the template patch in the object itself, under a best-matching assumption in a context-sensitive manner [1,2], or by similarity between synthesizing geometry features [19].

Our completion proposal is rather based on the assumption that the patch to be pasted has been pre-selected and placed at the desired position by the user.

### 3 Variational Recovery Model

Solving the variational problem (1) on surfaces requires the definition of the discrete manifold representing the underlying object of interest as well as the discrete approximation of the first-order differential operators involved.

We thus assume  $\mathcal{M} := (V, T)$  to be a triangulated surface (mesh) of arbitrary topology approximating a 2-manifold  $S$  embedded in  $\mathbb{R}^3$ , with  $V = \{v_i\}_{i=1}^n \in \mathbb{R}^{n \times 3}$  being the set of vertices, and  $T \in \mathbb{N}^{n_T \times 3}$  the set of face triangles,  $T = \{\tau_i\}_{i=1}^{n_T}$ . Implicitly, we further denote by  $E \subseteq V \times$

$V \in \mathbb{N}^{n_E \times 2}$ ,  $E = \{e_j\}_{j=1}^{n_E}$  the set of edges. We denote the first disk, i.e. the triangle neighbours of a vertex  $v_i$ , by  $\mathcal{D}(v_i) = \{\tau_m \mid v_i \in \tau_m\}$ . Let  $\mathcal{N} : \mathbb{R}^{n_V \times 3} \rightarrow \mathbb{R}^{n_T \times 3}$  be the mapping computing the piecewise-constant normal field over the triangles of the mesh, where the  $m$ -th element is the outward unit normal at face  $\tau_m = (v_i, v_j, v_k)$ , defined as

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

Notice that the normal vector’s sign depends on the orientation of the face. The desire for consistently oriented normals is that adjacent faces have consistent orientation. Under this discrete setting, the scalar functions  $x, y, z : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$  defined on  $S$  are sampled over the vertices  $v_i = (x_i, y_i, z_i) \in V$  of the mesh  $\mathcal{M}$  and are understood as piecewise linear functions.

We now introduce the discretization of the gradient operator on a 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 sparse matrix  $D \in \mathbb{R}^{n_E \times n_T}$  defined by

$$D_{ij} = \begin{cases} l_i & \text{if } \tau_j \cap \tau_k = e_i, k > j, \\ -l_i & \text{if } \tau_j \cap \tau_k = e_i, k < j, \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where  $l_i = \|e_i\|_2, i = 1, \dots, n_E$  is the length of  $i$ th edge.

The matrix  $D$  can be decomposed as  $D = L\bar{D}$ , with  $L = \text{diag}\{l_1, l_2, \dots, l_{n_E}\}$  being the diagonal matrix of edge lengths, whose values may be updated during the iteration scheme considered, and  $\bar{D} \in \mathbb{R}^{n_E \times n_T}$  an edge-length independent sparse matrix.

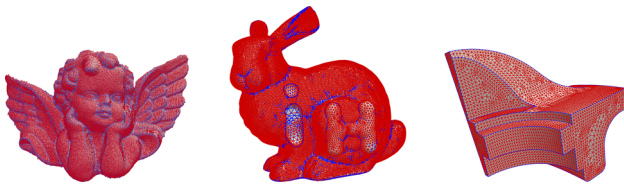
Key ingredients of the proposed formulation (1) are the two operator masks  $M_V$  and  $M_E$ . The role of the mask  $M_E$  is to adapt the recovery according to the surface morphology, while  $M_V$  selects the region to be preserved in the inpainting and completion tasks.

$M_E$  is a sharp detection mask represented by a binary vector  $M_E \in \{0, 1\}^{n_E}$  which has 1s in correspondence with sharp edges. Recalling that the dihedral angle associated with the edge  $e_i$  is the angle between normals to the adjacent triangle faces  $\tau_\ell$  and  $\tau_s$  which share  $e_i$ , we classify  $e_i$  as a sharp edge if the dihedral angle  $\theta_{\ell s} \in [0, 360)$  is greater than a given threshold  $th$ . In formulas

$$(M_E)_i = \begin{cases} 1 & \text{if } (\theta_{\ell s} > th) \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Given  $M_E$ , its complementary mask is the vector  $M_E^c = \mathbf{1}_{n_E} - M_E$ . Figure 2 shows  $M_E$  for three different surface





**Fig. 2** Examples of  $M_E$  mask for three different meshes: blue colours represent values 1, while red colours represent 0 values

meshes, where we empirically set  $th = 30$ , which typically produces good results.

The influence of the choice of the mask  $M_E$  in realizing the denoising task is shown in Fig. 3. The perturbed *sharp sphere* is illustrated in Fig. 3 on the left panel, and the denoised meshes on the right panel, obtained by applying the proposed method under the choice  $M_E = \mathbf{0}_{n_E}$ ,  $M_E = \mathbf{1}_{n_E}$ , in Fig. 3b and e, respectively. The space-variant mask  $M_E$  obtained with  $th = 30$  and illustrated in Fig. 3c is applied to obtain the denoised mesh in Fig. 3d.

Stemming from the consideration by which a general scanned surface is characterized by sharp as well as rounded features, we specify the form of problem (1) to determine solutions  $V^*$  which are close to the given data  $V_0$  according to the observation model,

$$\begin{aligned}
 V^* \in \arg \min_{V \in \mathbb{R}^{n_V \times 3}} \mathcal{J}(V; M_E, \lambda, a) \\
 \mathcal{J}(V; M_E, \lambda, a) := \frac{\lambda}{2} \sum_{i=1}^{n_V} ((M_V)_i (V_i - (V_0)_i))^2 + \\
 + \sum_{j=1}^{n_E} \{ (M_E)_j \phi(\|(DN(V))_j\|_2; a) + \\
 + (M_E^c)_j \|(DN(V))_j\|_2^2 \}, \quad (6)
 \end{aligned}$$

where  $\|\cdot\|_2$  denotes the Frobenius norm. The functional in (6) involves three terms designed to meet three different and competing requirements that arise quite naturally from the intuitive concept of surface recovery: (1) fidelity to the

known data; (2) a parametric (defined in terms of the parameter  $a \in \mathbb{R}_+$ ) discontinuity-preserving smoothing favouring piece-wise constant normals; (3) smooth connection between parts and inside unknown regions. The functional is composed by the sum of smooth convex (quadratic) terms and a non-smooth non-convex regularization term; thus, the functional  $\mathcal{J}$  in (6) is bounded from below by zero, non-smooth and can be convex or non-convex depending on the values of  $M_E$  and  $a$ .

### 3.1 Sparsity-Inducing Penalty

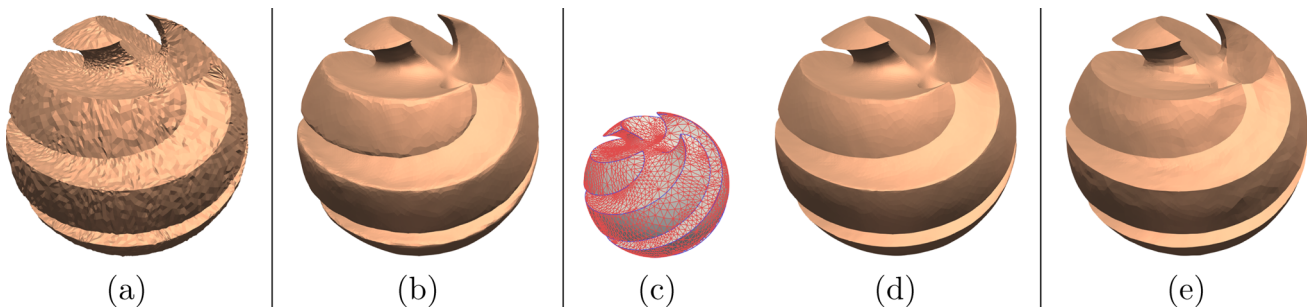
We aim at constructing a parameterized sparsity-promoting regularizer characterized by a tunable degree of non-convexity  $a \in \mathbb{R}_+$  inducing sparsity on the vector of components  $\|(DN)_i\|_2, i = 1, \dots, n_E$ , which represent the normal variation between adjacent triangles sharing the  $i$ -th edge.

A substantial amount of recent works has studied the class of sparsity-promoting parametrized non-convex regularizers, given their provable theoretical properties and practical performances [20,21]. We consider here one of the most effective representative of this class, i.e. the Minimax Concave (MC) penalty  $\phi(\cdot; a) : [0, +\infty) \rightarrow \mathbb{R}$ , introduced in [22] and used previously in [23] applied to  $\|(DN)_i\|_2$  in the context of mesh editing, defined by:

$$\phi(t; a) = \begin{cases} -\frac{a}{2}t^2 + \sqrt{2a}t & \text{for } t \in [0, \sqrt{2/a}), \\ 1 & \text{for } t \in [\sqrt{2/a}, +\infty) \end{cases} \quad (7)$$

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

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



**Fig. 3** Effect of the mask  $M_E$  on the denoising task: original noisy mesh (a); setting  $M_E = \mathbf{0}_{n_E}$  (b); using a space-variant  $M_E$  mask (c)–(d); setting  $M_E = \mathbf{1}_{n_E}$  (e). The perturbed *sharp sphere* on the left panel has been corrupted according to (32) with  $\gamma = 0.15$

We denote by  $\phi'(t; a)$  and  $\phi''(t; a)$  the first-order and second-order derivatives of  $\phi$  with respect to the variable  $t$ , respectively.

The parameter  $a$  allows to tune the degree of non-convexity, such that  $\phi(\cdot; a)$  mimics the asymptotically constant behaviour of the  $\ell_0$  pseudo-norm for  $a \rightarrow \infty$ , while behaves as an  $\ell_1$  regularization term, for values  $a$  approaching to zero. For values of  $a$  in between, the MC penalty function in (7) is a sparsity-inducing penalty which preserves sharp features in normal variations better than  $\ell_0$ -pseudo-norm regularizer, and more accurate than  $\ell_1$  regularizer which tends to produce shrinkage effects.

This motivated us to use it in the construction of the regularizer  $\mathcal{R}_1(V; M_E)$ .

### 3.2 Edge-Based Discretization of the Willmore Energy

Numerical approximations of the Willmore energy in digital geometry processing and geometric modeling are mainly based either on finite element discretization and numerical quadrature [24,25], or on discrete differential geometry approaches ab initio. Discrete isometric bending models, derived from an axiomatic treatment of discrete Laplace operators [26], the discrete conformal vertex-based energy well-defined for simplicial surfaces using circumcircles of their faces [3,27], and the integer linear programming approach [28] all fall into the latter class.

Here, we consider an alternative edge-based discrete approximation of the Willmore energy (2) for open triangulated surfaces  $\mathcal{M}$  represented by polygonal meshes. This energy is a sum over contributions from individual edges

$$E(\mathcal{M}) = \frac{1}{2} \sum_{j=1}^{nE} \|e_j\|^2 \|(DN)_j\|^2, \tag{8}$$

where  $(DN)_j$  measures how the surface ‘‘curves’’ near  $e_j$ . To derive the continuum limit of (8) in the limit of vanishing triangle size, we assume that  $S$  is a two-dimensional manifold of arbitrary topology embedded in  $\mathbb{R}^3$  and parameterized by  $(X, \Omega)$  with  $\Omega \subset \mathbb{R}^2$ , an open reference domain and define

$$X : \Omega \rightarrow S; \quad \xi \mapsto X(\xi),$$

the corresponding coordinate map (that is, the parametrization of  $S$  at a given point). We denote the local coordinates in  $\Omega$  as  $(\xi_1, \xi_2)$ . For a given point  $x \in X(\Omega) \subset S$ , the tangent space  $T_x S$  at  $x$  is spanned by  $\left\{ r_1 := \frac{\partial X(x)}{\partial \xi_1}, r_2 := \frac{\partial X(x)}{\partial \xi_2} \right\}$ , the induced metric is given by  $g_{ij} = r_i \cdot r_j$ , its inverse is denoted by  $g^{ij}$ , so that  $g^{ik} g_{kj} = \delta_i^j$ , or in matrix notation  $[g^{ij}] = [g_{ij}]^{-1}$ , and its determinant is defined as

$$\det(g) \equiv |g| = \frac{1}{2} \epsilon^{ik} \epsilon^{jl} g_{ij} g_{kl} = \frac{1}{2} (g_{ij} g_{kl} - g_{ik} g_{jl}).$$

The second fundamental form  $II : T_x S \times T_x S \rightarrow \mathbb{R}$  is the symmetric bilinear form represented by the coefficients  $L_{ij} = -r_i \cdot \partial_j n$ ,  $1 \leq i, j \leq 2$ .

When the grid size of the triangulation  $\mathcal{M}$  is sent to 0, the energy (8) approximates the Willmore energy as stated by the following Proposition 1.

**Proposition 1** *Let  $S \subset \mathbb{R}^3$  be a two-dimensional manifold,  $\mathcal{M}$  an underlying flat triangulated approximation of  $S$ . Let  $\mathcal{M}_j$  be regular flat triangulated surfaces  $\mathcal{M}_j \subset \mathbb{R}^3$  with  $\text{size}(\mathcal{M}_j) \rightarrow 0$  and  $\mathcal{M}_j \rightarrow S$  for  $j \rightarrow \infty$ . Then, the discrete energy (8) approximates the Willmore energy of  $S$ , i.e.*

$$\lim_{j \rightarrow \infty} E(\mathcal{M}_j) = \frac{1}{2} \int_S (h^2 - k) dS. \tag{9}$$

**Proof** Let us first consider the integrand of (2) in the continuum, with  $h = \kappa_1 + \kappa_2 = \text{tr}(L_k^i)$  being the mean curvature and  $k = \frac{1}{2} \kappa_1 \kappa_2 = \det(L_k^i)$  the Gaussian curvature where  $\kappa_1, \kappa_2$  represent the principal curvatures. The second fundamental form with components  $L_{ij}$  relates with the linear map  $L_i^k$  with respect to the basis of  $T_x S$ , according to the matrix equation:  $[g^{ij}][L_{ij}] = [L_j^i]$ , and we denote  $L_{ij} = \sum_k g_{ik} L_j^k$ ,  $1 \leq i, j \leq 2$ . Following notations in [29], we use the identity

$$g^{ij} g^{kl} = g^{ik} g^{jl} + \epsilon^{il} \epsilon_{mn} g^{mj} g^{nk}$$

in the integrand of (2) as

$$\begin{aligned} h^2 - 2k &= (L_i^i)^2 + \epsilon^{il} \epsilon_{mn} L_l^m L_i^n \\ &= (g^{ik} L_{ik})^2 + \epsilon^{il} \epsilon_{mn} (g^{mj} L_{jl}) (g^{nk} L_{ki}) \\ &= g^{ij} g^{kl} L_{ik} L_{jl}. \end{aligned} \tag{10}$$

Substituting in (10) the Weingarten equations  $\partial_i n = L_i^k r_k$  and  $L_i^k, i = 1, 2$ , we have

$$\begin{aligned} g^{ij} g^{kl} L_{ik} L_{jl} &= L_k^j L_j^k g^{kj} g_{jk} \\ &= L_k^j r_j L_j^k r_k g^{kj} \\ &= \partial_k n \cdot \partial_j n g^{kj} \end{aligned} \tag{11}$$

which is the gradient of the normal vector field. Therefore, replacing (10)–(11) in (2), we get

$$\int_S \partial_k n \cdot \partial_j n g^{kj} dS. \tag{12}$$

For sufficiently fine, non-degenerate tessellations  $\mathcal{M}_j$  approximating  $S$ , we consider a partition of the undeformed surface  $S$  into the disjoint union of diamond-shaped tiles,  $\bar{T}$ , associated with each mesh edge  $e$ . Following Meyer et al.

[30], one can use the barycenter of each triangle to define these regions or, alternatively, the circumcenters. Over such a diamond partition, the integral (12) is defined as the sum over all the diamond tiles, which reads

$$\int_S \partial_k n \cdot \partial_j n g^{kj} dS = \sum_{i=1}^{n_E} \int_{(\tilde{T})_i} |\partial_i n|^2 d\tilde{T}. \tag{13}$$

If the triangles do not degenerate, we can approximate the area of the diamond related to the edge  $e_i$  in  $\mathcal{M}_j$  by  $\|e_i\|^2$ , i.e.  $d\tilde{T} \approx \|e_i\|^2$ , which implies that  $|\partial_i n|^2 d\tilde{T} \approx \|(DN)_i\|^2 \|e_i\|^2$ .  $\square$

The result of the limiting process depends on the triangulations considered. In particular, we assume the triangulations of  $S$  consist of almost equilateral triangles. For our purposes, the discrete Willmore energy will be used based on the observation that (2) is invariant under rigid motions and uniform scaling of the surface, which implies that  $E(S)$  itself is a conformal invariant of the surface  $S$ , see [3].

**Remark 1** Even if the introduced discrete formulation is very simple when compared with the ones introduced in [3,31], it practically produces good results. In order to validate the effective applicability of the proposed discrete Willmore energy, we evaluated  $E(\mathcal{M})$  in (8) on a uniformly tessellated sphere, for decreasing average edge-size  $h = \{0.1208, 0.0308, 0.0076\}$ . The achieved energy values  $E(\mathcal{M}_{h_1}) = 0.1182$ ,  $E(\mathcal{M}_{h_2}) = 0.0075$ ,  $E(\mathcal{M}_{h_3}) = 0.00047$ , tend to zero, as theoretically expected from (2).

### 4 Numerical Solution of the Optimization Problem

In this section, we illustrate the ADMM-based iterative algorithm used to compute the numerical solution of (6).

In order to define the ADMM iteration on triangular mesh surfaces, we first consider a matrix variable  $N \in \mathbb{R}^{n_T \times 3}$  with row components defined as in (3) and resort to the variable splitting technique by defining  $t \in \mathbb{R}^{n_E \times 3}$  as  $t := DN$ , where  $D$  is defined in (4). The optimization problem (6) can be thus reformulated as

$$\begin{aligned} & \{V^*, N^*, t^*\} \\ & \in \arg \min_{V, N, t} \left\{ \frac{\lambda}{2} \sum_{i=1}^{n_V} ((M_V)_i (V_i - (V_0)_i))^2 + \right. \\ & \quad \left. + \sum_{j=1}^{n_E} \left[ (M_E)_j \phi(\|t_j\|_2; a) + (M_E^c)_j \|t_j\|_2^2 \right] \right\}, \\ & \text{s.t. } t = DN, \quad N = \mathcal{N}(V). \end{aligned} \tag{14}$$

We define the augmented Lagrangian functional associated with problem (14) as

$$\begin{aligned} & \mathcal{L}(V, N, t, \rho_1, \rho_2; \lambda, \beta_1, \beta_2, a) \\ & := \frac{\lambda}{2} \sum_{i=1}^{n_V} ((M_V)_i (V_i - (V_0)_i))^2 + \\ & \quad + \sum_{j=1}^{n_E} \left[ (M_E)_j \phi(\|t_j\|_2; a) + (M_E^c)_j \|t_j\|_2^2 - \right. \\ & \quad \left. - \langle \rho_{1_j}, t_j - (DN)_j \rangle + \frac{\beta_1}{2} \|t_j - (DN)_j\|_2^2 \right] + \\ & \quad + \sum_{m=1}^{n_T} \left[ - \langle \rho_{2_m}, N_m - \mathcal{N}_m(V) \rangle \right. \\ & \quad \left. + \frac{\beta_2}{2} \|N_m - \mathcal{N}_m(V)\|_2^2 \right], \end{aligned} \tag{15}$$

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 now consider the following saddle-point problem:

$$\begin{aligned} & \text{Find } (V^*, N^*, t^*, \rho_1^*, \rho_2^*) \in \mathbb{R}^{n_V \times 3} \times \\ & \quad \times \mathbb{R}^{n_T \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_T \times 3} \\ & \text{s.t. } \mathcal{L}(V^*, N^*, t^*, \rho_1, \rho_2) \leq \\ & \quad \leq \mathcal{L}(V^*, N^*, t^*, \rho_1^*, \rho_2^*) \leq \mathcal{L}(V, N, t, \rho_1^*, \rho_2^*), \\ & \forall (V, N, t, \rho_1, \rho_2) \in \mathbb{R}^{n_V \times 3} \times \\ & \quad \times \mathbb{R}^{n_T \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_E \times 3} \times \mathbb{R}^{n_T \times 3}. \end{aligned} \tag{16}$$

An ADMM-based iterative scheme can now be applied to approximate the solution of the saddle-point problem (15)–(16). Initializing to zeros both the dual variables  $\rho_1^{(0)}, \rho_2^{(0)}$  and setting  $N_m^{(0)} = \mathcal{N}_m(V^{(0)})$ ,  $m = 1, \dots, n_T$ , the  $k$ th iteration of the proposed alternating iterative scheme reads:

$$t^{(k+1)} = \arg \min_{t \in \mathbb{R}^{n_E \times 3}} \mathcal{L}(V^{(k)}, N^{(k)}, t; \rho_1^{(k)}, \rho_2^{(k)}), \tag{17}$$

$$N^{(k+1)} = \arg \min_{\substack{N \in \mathbb{R}^{n_T \times 3}, \\ \|N_t\|=1}} \mathcal{L}(V^{(k)}, N, t^{(k+1)}; \rho_1^{(k)}, \rho_2^{(k)}), \tag{18}$$

$$V^{(k+1)} = \arg \min_{V \in \mathbb{R}^{n_V \times 3}} \mathcal{L}(V, N^{(k+1)}, t^{(k+1)}; \rho_1^{(k)}, \rho_2^{(k)}), \tag{19}$$

$$\rho_1^{(k+1)} = \rho_1^{(k)} - \beta_1 (t^{(k+1)} - DN^{(k+1)}), \tag{20}$$

$$\rho_2^{(k+1)} = \rho_2^{(k)} - \beta_2 (N^{(k+1)} - \mathcal{N}(V^{(k+1)})). \tag{21}$$

The updates of Lagrangian multipliers  $\rho_1$  and  $\rho_2$  have closed form. In the following, we show in detail how to solve the three minimization sub-problems (17), (18) and (19) for the primal variables  $t, N$  and  $V$ , respectively.

**Sub-problem for  $t$ .**

The minimization sub-problem for  $t$  in (17) can be explicitly rewritten as:

$$t^{(k+1)} = \arg \min_{t \in \mathbb{R}^{n_E \times 3}} \sum_{j=1}^{n_E} \left[ (M_E)_j \phi(\|t_j\|_2; a) + (M_E^c)_j \|t_j\|_2^2 - \langle \rho_{1j}, t_j - (DN)_j \rangle + \frac{\beta_1}{2} \|t_j - (DN)_j\|_2^2 \right], \tag{22}$$

where we omitted the constant terms in (15). Due to the separability property of  $\phi(\cdot; a)$ , problem (22) is equivalent to  $n_E$  independent, three-dimensional problems for each  $t_j$ ,  $j = 1, \dots, n_E$  in the form

$$t_j^{(k+1)} = \arg \min_{t_j \in \mathbb{R}^3} \left\{ \phi(\|t_j\|_2; a) + \frac{\alpha}{2} \|t_j - r_j^{(k+1)}\|_2^2 \right\}, \tag{23}$$

where

$$r_j^{(k+1)} := \frac{1}{\beta_1 + 2(M_E^c)_j} \left( \beta_1 (DN^{(k)})_j + (\rho_1^{(k)})_j \right)$$

and  $\alpha = \frac{\beta_1 + 2(M_E^c)_j}{(M_E)_j}$ . where we conventionally set  $\frac{x}{0} = 0$ .

Necessary and sufficient conditions for strong convexity of the cost functions in (23) are demonstrated in [32]. In particular, problems (23) are strongly convex if and only if the following condition holds:

$$\frac{\beta_1 + 2(M_E^c)_j}{(M_E)_j} > a, \forall j = 1, \dots, n_E \implies \implies \beta_1 = \varepsilon \max_j \{ (M_E)_j a - 2(M_E^c)_j \}, \text{ for } \varepsilon > 1. \tag{24}$$

We noticed that the sub-problem is always convex when  $t_j$  has associated  $(M_E)_j = 0$ , as it eliminates  $\phi(\cdot; a)$  from the sub-problem.

Whenever (24) holds, the unique minimizers of (23) can be obtained in closed form as

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

where  $v = \frac{\alpha}{\alpha - a}$  and  $\zeta = \frac{\sqrt{2a}}{\alpha - a}$ .

We remark that the condition on  $\beta_1$  in (15) only ensures the convexity conditions (24) of  $t$ -subproblem (23), but does not guarantee convergence of the overall ADMM scheme.

**Sub-problem for  $N$ .** The minimization sub-problem (18) for  $N$  can be reformulated as:

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

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

$$\left( D^T D + \frac{\beta_2}{\beta_1} I \right) N = \frac{\beta_2}{\beta_1} \mathcal{N}(V^{(k)}) + \frac{\rho_2^{(k)}}{\beta_1} + D^T \left( t^{(k+1)} - \frac{1}{\beta_1} \rho_1^{(k)} \right). \tag{25}$$

Since  $\beta_1, \beta_2 > 0$ , the linear system coefficient matrix is sparse, symmetric, positive definite and identical for all three coordinate vectors. The systems can thus be solved efficiently by applying, e.g. a unique Cholesky decomposition. At each iteration, the edge lengths diagonal matrix  $L$  in  $D = L\bar{D}$ , defined in (4), needs to be updated as the vertices  $V$  move to their updated 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 a normal field.

The reconstructed normal map  $N^*$  obtained by solving (14) via the proposed ADMM satisfies the orientation consistency, as proved in [5], thus reducing the foldovers issue. This property is not trivially satisfied by most of the two-stage mesh denoising algorithms (normal smoothing and vertex update). They present the normal orientation ambiguity problem in the vertex updating stage, which provokes ambiguous shifts of the vertex position due to direction inconsistency of the normal vectors [33,34]. In [35], this issue is solved by an orientation-aware vertex updating scheme.

**Sub-problem for  $V$ .** Omitting the constant terms in (15), the sub-problem for  $V$  reads

$$V^{(k+1)} = \arg \min_{V \in \mathbb{R}^{n_V \times 3}} \mathcal{J}_V(V) \\ \mathcal{J}_V(V) := \frac{\lambda}{2} \sum_{i=1}^{n_V} ((M_V)_i (V_i - (V_0)_i))^2 + \sum_{m=1}^{n_T} \left[ \langle \rho_{2m}^{(k)}, \mathcal{N}_m(V) \rangle + \frac{\beta_2}{2} \|N_m^{(k+1)} - \mathcal{N}_m(V)\|_2^2 \right]. \tag{26}$$

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 (GD) algorithm with backtracking satisfying the Armijo condition or using the BFGS algorithm. In order to balance between the slow convergence properties of GD and the high computational costs required to compute the operators involved in the BFGS method, we also considered a heavy-ball type rule, following [36], and its extension with backtracking (covering also non-smooth problems) given in [37]. In particular, the heavy-ball method is a multi-step extension of gradient descent, which, starting from  $\bar{V}^{(0)} = V^{(k)}$ , iterates over  $V$  as follows

$$\bar{V}^{(j+1)} = \bar{V}^{(j)} - \alpha_j \nabla J(\bar{V}^{(j)}) + \delta_j (\bar{V}^{(j)} - \bar{V}^{(j-1)}), \quad j = 1, 2, \dots \tag{27}$$

where  $\alpha_j > 0$  is a step-size parameter and  $0 \leq \delta_j < 1$ . Note that for  $\delta_j = 0$ , (27) reduces to the gradient descent method. In [37], convergence of the scheme above to stationary points is proved in the context of non-convex cost functions as the one in (26), with an extension also to non-smooth scenarios.

All the numerical optimization methods here considered rely on a easily computable formula for the gradient of the functional  $\mathcal{J}_V$  in (26), which is derived in the following.

**Proposition 2** *Let  $s_{\tau_m} := \|(v_j - v_i) \times (v_k - v_i)\|_2/2$  be the area of the triangle  $\tau_m = (v_i, v_j, v_k)$  with updated vertices in  $V$ , and  $\mathcal{N}_m(V) = ((v_j - v_i) \times (v_k - v_i))/(2s_{\tau_m})$ . For all triangles  $m = 1, \dots, n_T$ ,*

$$\begin{aligned} \nabla_{v_i} \mathcal{J}_V(V) &= \lambda(M_V)_i (v_i - v_i^0) + \\ &+ \sum_{\tau_m \in \mathcal{D}(v_i)} \frac{1}{2s_{\tau_m}} \left[ \left( \rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)} \right) - \right. \\ &\left. - \left\langle \rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)}, \mathcal{N}_m(V) \right\rangle \mathcal{N}_m(V) \right] \\ &\times (v_k - v_j). \end{aligned} \tag{28}$$

**Proof** The gradient of  $\mathcal{J}_V(V)$  in (26) w.r.t vertex  $v_i \in V$ ,  $i = 1, \dots, n_V$  is nonzero only over the triangles sharing  $v_i$  which are contained in the first disk  $\mathcal{D}(v_i)$ . Therefore, the sum in (26) is reduced to

$$\begin{aligned} \nabla_{v_i} \mathcal{J}_V(V) &= \lambda(M_V)_i (v_i - v_i^0) + \sum_{\tau_m \in \mathcal{D}(v_i)} \tau_m = (v_i, v_j, v_k) \\ &\times \nabla_{v_i} \left( \underbrace{\left\langle \frac{z}{q}, (v_j - v_i) \times (v_k - v_i) \right\rangle}_{g_i} \right), \end{aligned}$$

where  $z = \rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)}$ ,  $q = \|(v_j - v_i) \times (v_k - v_i)\|_2$ , and the third term in (26) reduces to the scalar product  $g_i$  since both  $N_m^{(k+1)}$  and  $\mathcal{N}_m(V)$  have unitary norm. In order to

compute  $\nabla_{v_i}(g_i)$ , we resort on the following two properties, which hold for every constant vectors  $w, u \in \mathbb{R}^3$  and can be easily proved:

1.  $\nabla_{v_i}(\langle w, (v_j - v_i) \times (v_k - v_i) \rangle) = w \times (v_k - v_j)$ ;
2.  $\nabla_{v_i} \left( \left\langle \frac{w}{\|(v_j - v_i) \times (v_k - v_i)\|_2}, u \right\rangle \right) = -\langle w, u \rangle \frac{(v_j - v_i) \times (v_k - v_i) \times (v_k - v_j)}{\|(v_j - v_i) \times (v_k - v_i)\|_2^3}$ . (29)

To evaluate the product rule derivative, we apply property 1, with  $w = z/\|(v_j - v_i) \times (v_k - v_i)\|_2$  for the left-side term constant, while property 2 is applied with  $w = z$  and  $u = (v_j - v_i) \times (v_k - v_i)$  for a right-side term kept constant. Combining the results leads to the explicit formula for  $\nabla_{v_i} g_i$ :

$$\begin{aligned} \nabla_{v_i} g_i(V) &= \frac{\left( \rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)} \right) \times (v_k - v_j)}{\|(v_j - v_i) \times (v_k - v_i)\|_2} \\ &\left\langle \rho_{2_m}^{(k)} - \beta_2 N_m^{(k+1)}, (v_j - v_i) \times (v_k - v_i) \right\rangle \\ &- \frac{\left[ (v_j - v_i) \times (v_k - v_i) \times (v_k - v_j) \right]}{\|(v_j - v_i) \times (v_k - v_i)\|_2^3}, \end{aligned} \tag{30}$$

which reduces to (28). □

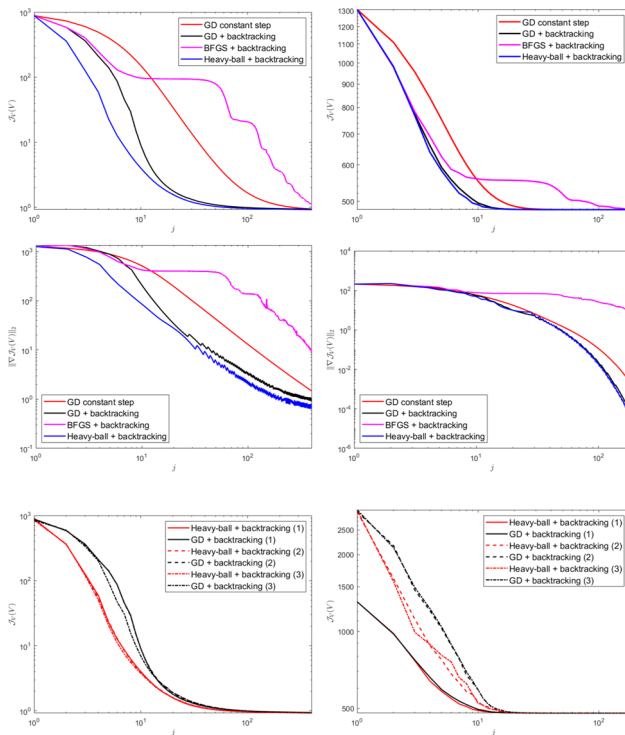
In Fig. 4 (first and second rows), we report the graphs showing both the energy decay and the gradient norm decay for the three different algorithms used, i.e. GD (with and without backtracking), BFGS and heavy-ball with backtracking. The plots are related to the meshes *twelve* (first column) and *block* (second column) as representative of the entire set of meshes analysed in the experimental section.

We remark that the use of Armijo-type backtracking rule is justified by the difficult expression (28), which makes the accurate estimation of the Lipschitz constant  $L_{\mathcal{J}_V}$  of  $\nabla \mathcal{J}_V(V)$  quite challenging. In the proposed strategy, a (typically) initially large step-size  $\alpha_0$  is then reduced depending on whether the following inequality is verified:

$$\mathcal{J}_V(\bar{V}^{(j+1)}) \leq \mathcal{J}_V(\bar{V}^{(j)}) - c_1 \alpha \|\nabla \mathcal{J}_V(\bar{V}^{(j)})\|_2^2 \tag{31}$$

with  $c_1 \in (0, 1)$  and where  $\bar{V}^{(j)}$  denotes the  $j$ -th update of  $V$  given by (27).

From the convergence plots, we notice that upon a manual selection of a sufficiently small constant step-size  $\alpha$  the convergence of plain GD without backtracking is as good as the one of the heavy-ball algorithm combined with backtracking. However, the former choice is problem-dependent; hence, a backtracking strategy automatically adjusting the value of  $\alpha$  to an appropriate size is preferred.



**Fig. 4** First and second rows: plots of the energy  $\mathcal{J}_V$  (first row) and gradient norm (second row) decay for sub-problem (26) with GD (with and without backtracking), heavy-ball update with backtracking and BFGS. Third row: energy decay for GD with backtracking and heavy-ball with backtracking, for three different initializations  $V^{(0)}$ . First column mesh twelve and second column mesh block corrupted by noise level  $\gamma = 0.2$

The graphs in Fig. 4 (third row) show the robustness to the initialization  $\bar{V}^{(0)}$  and show that both GD and heavy-ball with backtracking are consistent, regardless of the chosen initialization. However, the natural and most efficient choice for  $\bar{V}^{(0)}$  is a warm start given by the matrix  $V^{(k)}$  obtained as a solution of the problem (26) in the previous ADMM iteration.

The rigorous analysis of the convergence properties of our proposed three block ADMM scheme following, e.g. [38] is not easy to derive. However, we will provide some evidence of the numerical convergence in Sect. 6.

### 5 A Practical Use of the Geometry Repair Framework

In the following, we provide some details for the practical use of the geometric framework introduced above in view of its application to the three different tasks we are interested in.

*Feature-aware mesh denoising.* The goal of a surface denoising algorithm is to remove undesirable noise or spu-

rious 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 normal field. To achieve this goal, the natural choice is to set in (6)  $M_V = \mathbf{1}_{n_V}$  and define  $M_E$  as in (5), so as to distinguish salient edges from smooth regions. In case of severe noise, the estimate of the mask  $M_E$  may be affected by false edge detections. In such case, we suggest to recompute the edge mask  $M_E$  along the ADMM iterations.

*Smooth hole filling/inpainting.* In contrast to techniques for image inpainting, which make use of the given spatial structure of the data (the regular grid of an image), surfaces lack a natural underlying spatial domain, which brings an additional degree of freedom in the setting of problem. At the same time, vertices’ positions encode both function values and the domain of the function to be reconstructed. The initial mesh  $\mathcal{M}_0 = (V_0, T_0)$  thus has to be set as the original (possibly noisy) incomplete mesh with trivially enclosed and labelled disconnected holes—region  $S_D$ —marked as zeros in  $M_V$ . On the other hand, the mask  $M_E$  can still be defined as in (5), by additionally forcing zero values on the edges in  $S_D$ . The proposed geometric repair algorithm then performs simultaneously denoising, outside the holes, and smooth filling in the internal part of the holes, through the regularizer  $\mathcal{R}_2$ .

*Context-aware completion.* In some applications, smooth filling of holes is not sufficient: this is the case in archaeology and in general cultural heritage applications where the main goal is the reconstruction of a digital twin of a cultural heritage object. Some parts of the original 3D model can be damaged or missing but can be completed by means of characteristic parts taken from the object under consideration or from others. Given the original incomplete mesh  $\mathcal{M}_0$  with a region of interest bounded by a curve  $b_0$  and characterized by vertices  $\bar{V} \subset V_0$  and triangles  $\bar{T} \subset T_0$ , together with a template patch  $\mathcal{P} = (V_{\mathcal{P}}, T_{\mathcal{P}})$ , bounded by a curve  $b_{\mathcal{P}}$ , we build a repaired mesh  $\mathcal{M}^*$  by replacing  $(\bar{V}, \bar{T})$  by  $(V_{\mathcal{P}}, T_{\mathcal{P}})$  and blending the two parts through the proposed variational model.

Note that, in case the region of interest on  $\mathcal{M}_0$  that has to be completed is a hole, then trivially  $(\bar{V}, \bar{T})$  are empty sets.

We assume that the template patch  $\mathcal{P}$  is properly aligned in the correct position and that both polygonals  $b_0$  and  $b_{\mathcal{P}}$  are approximants of oriented, closed, simple curves in  $\mathbb{R}^3$  with the same number of vertices. The correct positioning can thus be performed either automatically (by rigid body transformation algorithms) or through user interaction.

A narrow band around  $b_0$ , named  $strip(b_0)$ , containing at least 2-disk of triangle neighbours adjacent to  $b_0$ , plays the role of  $S_D$ . Hence,  $M_V$  is the characteristic function of  $\mathcal{M}_0 \setminus strip(b_0)$ , i.e. is zeros only on  $strip(b_0)$ .

The operator mask  $M_E$  has values one for each sharp edge in both  $\mathcal{M}_0 \setminus strip(b_0)$  and  $\mathcal{P}$ . According to the user desiderata, the blending can be performed in three different ways: edges in  $strip(b_0)$  all zeros in  $M_E$  to force a smooth joint with the template; edges in  $strip(b_0)$  all ones, to keep a sharp connection; edges in  $strip(b_0)$  defined by the spatially adaptive  $M_E$  in (5) to maintain geometric continuity  $G^0/G^1$  over the blended region.

The vertices  $V^*$  of the completed surface  $\mathcal{M}^*$  are obtained by minimizing (6), properly initialized with  $V^{(0)} = (V_0 \setminus \bar{V}) \cup V_{\mathcal{P}}$ , while maintaining the connectivity defined by  $T^* = (T_0 \setminus \bar{T}) \cup T_{\mathcal{P}}$ . The connectivity  $T^*$  is automatically achieved as we imposed  $b_0 \equiv b_{\mathcal{P}}$ .

We refer the reader to Fig. 1 for visual representation of the three different tasks performed. Moreover, Sect. 6 offers additional insights.

### 6 Numerical Examples

We validate the proposed geometric framework both qualitatively and quantitatively on a variety of benchmark triangulated surfaces characterized by different sharpness and smoothness features and on some real datasets.

At the aim of a quantitative validation, meshes  $\mathcal{M}_0 = (V_0, T_0)$  have been synthetically corrupted. The noisy vertices in  $V_0$  correspond to underlying noise-free vertices  $V_{GT}$  by the following additive degradation model

$$V_0 = V_{GT} + \eta d, \tag{32}$$

where the product  $\eta d$  accounts for the noise perturbations. Namely,  $\eta \in \mathbb{R}^{n_V}$  is assumed to be at each vertex independently and identically distributed as a zero-mean Gaussian random variable, i.e.  $\eta_i \sim Gauss(0, \sigma^2)$ ,  $i = 1, \dots, n_V$ , with known variance  $\sigma^2$ , and  $d \in \mathbb{R}^{n_V \times 3}$  is a vector field of noise directions with elements  $d_i \in \mathbb{R}^3$ ,  $i = 1, \dots, n_V$ , which can be either random directions or the normals to the vertices. The perturbations are thus characterized by a noise level  $\gamma \in \mathbb{R}_+$  defined by  $\sigma = \gamma \bar{l}$ , with  $\bar{l}$  representing the average edge length.

Quantitative evaluation is done in terms of the following error metrics, which measure the discrepancy of the computed  $V^*$ ,  $N^*$  w.r.t. the noise-free mesh  $V_{GT}$ ,  $N_{GT}$ :

- Mean squared angular error (MSAE)

$$MSAE = \mathbb{E}[\mathcal{L}(N_{GT}, N^*)^2], \tag{33}$$

- $L_2$  vertex to vertex error ( $E_V$ )

$$E_V = \frac{\|V^* - V_{GT}\|_F}{n_V}. \tag{34}$$

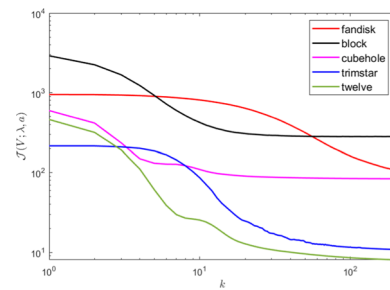


Fig. 5 Empirical convergence of ADMM algorithm for some reconstructed meshes

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

$$k > 200, \quad \|V^{(k+1)} - V^{(k)}\|_2 / \|V^{(k)}\|_2 < 10^{-6}. \tag{35}$$

Figure 5 shows the energy decay curve versus the number of iterations for some of the meshes reported in this section. We observe that for all meshes considered the energy converges to a stationary value. This represents an empirical validation on the numerical convergence of the proposed ADMM-based minimization scheme. Having performed a comparative analysis between inner solvers in Sect. 4, we used the GD algorithm with backtracking for solving the subproblem for  $V$ , with warm start strategy allowing us to restrict to a few number (three in our experiments) of GD iterations while achieving good relative accuracy.

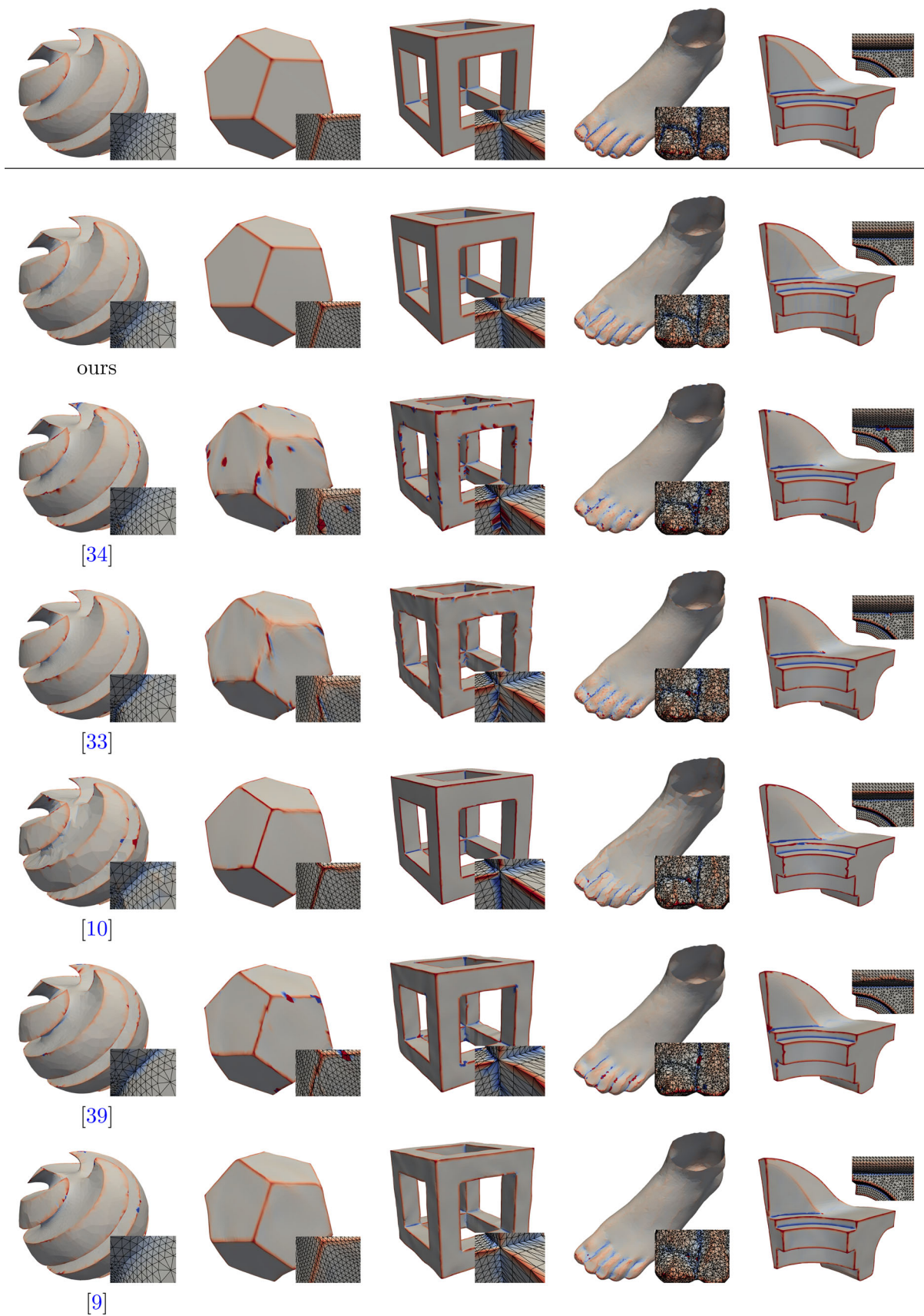
With respect to the comparisons showed with competing approaches for mesh repairing, we remark that most of them are based on hierarchical data structures and combined with various heuristic algorithms. On the contrary, the results presented in the following are directly derived from the solution of the proposed unified mathematical optimization problem and do not require any heuristic post-processing procedures.

All the meshes are rendered in flat shading model and visualized using ParaView software.

**Example 1: feature-aware denoising.** To evaluate the performance of the proposed method for mesh denoising, we compared the results with other state-of-the-art variational methods for mesh denoising, namely the methods introduced in [10,33,34,39], which have been kindly provided by authors of [39] at <https://github.com/bldeng/GuidedDenoising>, and a learning-based approach, presented in [9]. For each method, we show their best results achieved by tuning the corresponding set of parameters.

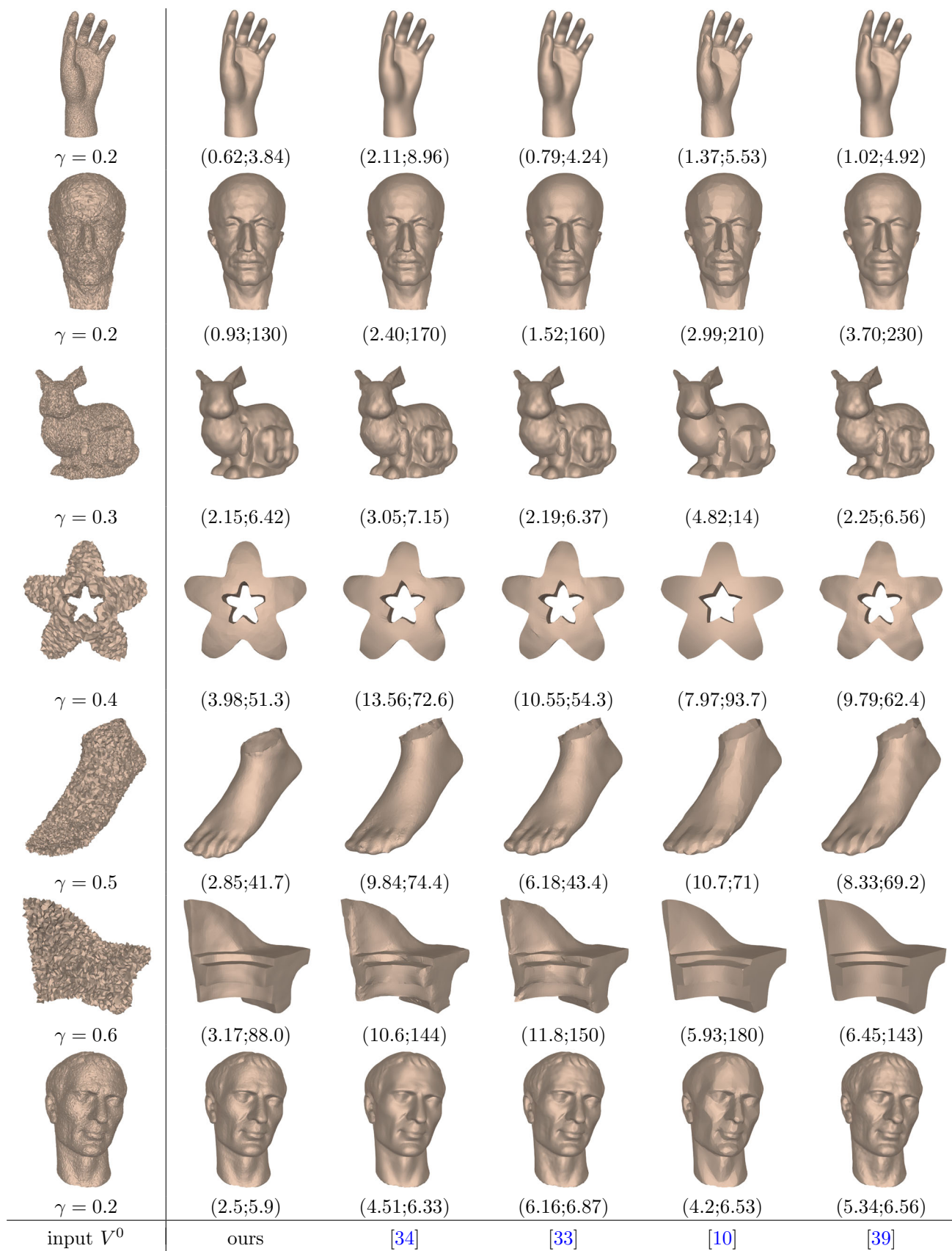
Figure 6 shows the denoised meshes coloured by their mean curvature scalar map, with fixed range, together with zoomed details on mesh edges. From a visual inspection, we notice remarkable overlaps in the denoised meshes obtained from the other compared methods and severe perturbations





**Fig. 6** Examples of denoising: results of noisy-free input meshes (first row) corrupted by noise levels  $\gamma = \{0.15, 0.3, 0.3, 0.2, 0.2\}$ , from left to right





**Fig. 7** Examples of denoising: comparison of our denoising framework with related works on meshes synthetically corrupted by noise levels  $\gamma$ . Reported metrics: ( $MSAE \times 10^{-2}$ ;  $E_V \times 10^{-6}$ )

of the triangle shapes in the reconstructed meshes. To further demonstrate how robust our approach is w.r.t. to increasing noise perturbation, in Fig. 7 we reported qualitative and quantitative results for noise levels  $\gamma = \{0.2, 0.2, 0.3, 0.4, 0.5, 0.6\}$  - from top to bottom. In the last row, the mesh has been corrupted by arbitrary perturbations on the noise directions ( $d_i$ ) in (32). Below each recovered surface, we report the quantitative evaluations according to the two error metrics ( $MSAE \times 10^2, E_V \times 10^6$ ). Both quantitatively and qualitatively the results confirm the effectiveness of the proposed variational model in preserving sharp features while smoothly recovering rounded parts. 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 [11] which is the slowest, while it is comparable to the other compared methods.

To improve the estimation of mask  $M_E$  for severe noise, we dynamically updated the edge mask  $M_E$  every three ADMM iterations.

**Example 2: hole filling/inpainting.** We applied our geometric framework for the recovery of various meshes  $\mathcal{M}_0$  which exhibit holes or damaged parts. Fig.2 illustrates the basic workflow for the inpainting task on *angel* mesh which takes as input the original eventually noisy mesh  $\mathcal{M}_0$  (Fig. 2, left) and the inpainting mask  $M_V$ , which can be of arbitrary topology, in the figure the holes to be filled are marked as 0 in  $M_V$  and blue coloured. The recovery of *angel* mesh using smooth hole filling is illustrated in Fig. 2(second row).

Figure 8 (first row) shows the challenging *Igea* mesh which presents a deep groove on the left side of the mouth and a shallower one on the right cheek. Our geometric framework was able to inpaint the shallower hole perfectly, while the deep one was filled in a satisfactory, even if not complete, way. This is justified by the different contribution of Willmore vs sparsity-inducing penalties. The latter acts more strongly with respect to the former, especially for high levels of noise. Hence, adding suitable weights to the two penalties could overcome to this disparity.

The data set *minerva* shown in Fig. 8 (second row, first column) presents a few holes caused by the scanner acquisition, in the head and under the nose. Moreover, a vertical strip has been intentionally added to the inpainting region  $S_D$  in order to remove the groove provoked by the gluing of the two parts of the *minerva's* face. This dataset has been provided by ENEA, Bologna, Italy, and acquired by a VIVID laser scanner. The dataset presents inherent noise due to the optical acquisition system. The result of repairing the damaged geometry and filling surface holes is illustrated in Fig. 8c.

In Fig. 8 (third row), the inpainting framework has been applied to repair a *shard* from neolithic pottery received by the CEPAM laboratory (CNRS France), obtained by fusion of more fragments. The inpainting region, shown in Fig.8

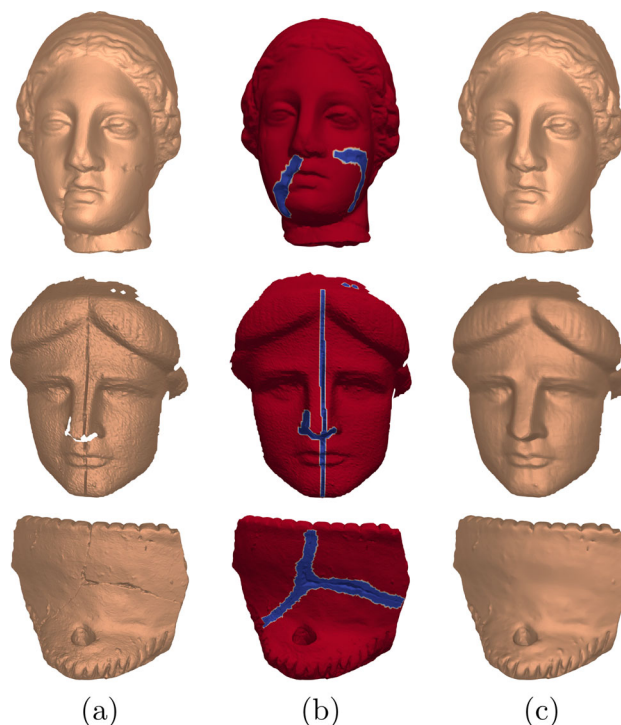


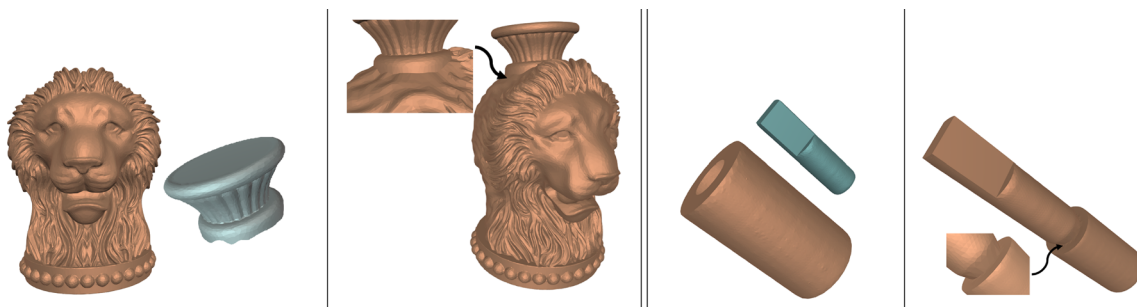
Fig. 8 Examples of surface inpainting: (a) original damaged object; (b) inpainting mask  $M_V$ ; (c) inpainted surface

(third row, second column), has been intentionally imposed to eliminate obvious fractures between joined fragments.

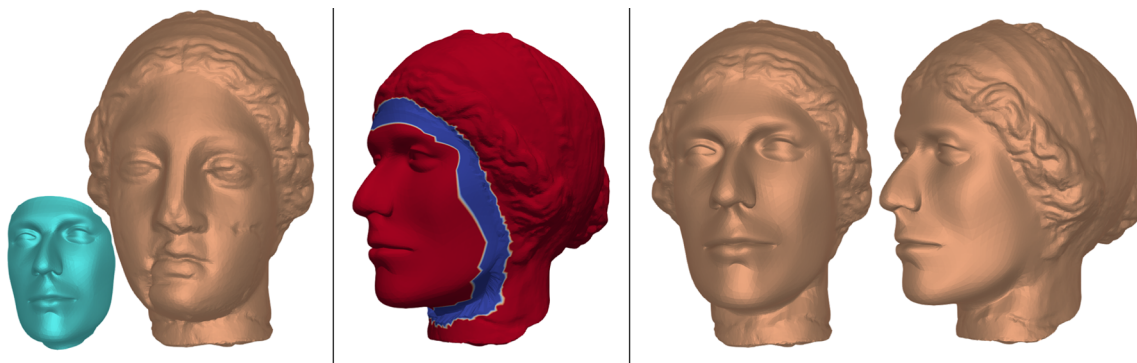
**Example 3: context-aware completion.** We finally applied context-aware completion as an editing tool for seamless object fusion. Completion results for the meshes *lion*, *screw-driver*, and *igea* are illustrated in Figs. 9 and 10. The templates  $\mathcal{P}$  smoothly complete the original surfaces.

A critical aspect in context-aware completion is the continuity imposed in the joint region, which we denoted by  $strip(b_0)$ . Conditions for geometric continuity between parametric surfaces are well assessed, while for meshes a rigorous treatment on this topic is still missing. In our framework, according to the user's desiderata, the template  $\mathcal{P}$  can be joined to  $\mathcal{M}_0$ , both smoothly, by setting  $M_E(strip(b_0)) \equiv 0$ , in a sharp manner by setting  $M_E(strip(b_0)) \equiv 1$ , or in a blended fashion by simply using the  $M_E$  mask of one of the two meshes (or even a combination of them). Therefore, imposing different continuity conditions for  $strip(b_0)$  means to define in a different way the mask  $M_E$  in correspondence to the  $strip(b_0)$ .

A typical example is shown in Fig. 11(left panel) where a synthetically created hole on the *fandisk* mesh  $\mathcal{M}_0$  is filled with a similar corner patch—template  $\mathcal{P}$  cyan coloured. In the right panel, we report details onto the completion area  $\mathcal{M}_0 \cup \mathcal{P}$  (a), output  $\mathcal{M}^*$  for  $M_E(strip(b_0)) \equiv 0$  (b),  $\mathcal{M}^*$  for  $M_E(strip(b_0)) \equiv 1$  (c),  $\mathcal{M}^*$  for  $M_E(strip(b_0))$  estimated from dihedral angles (d). Note that the initial

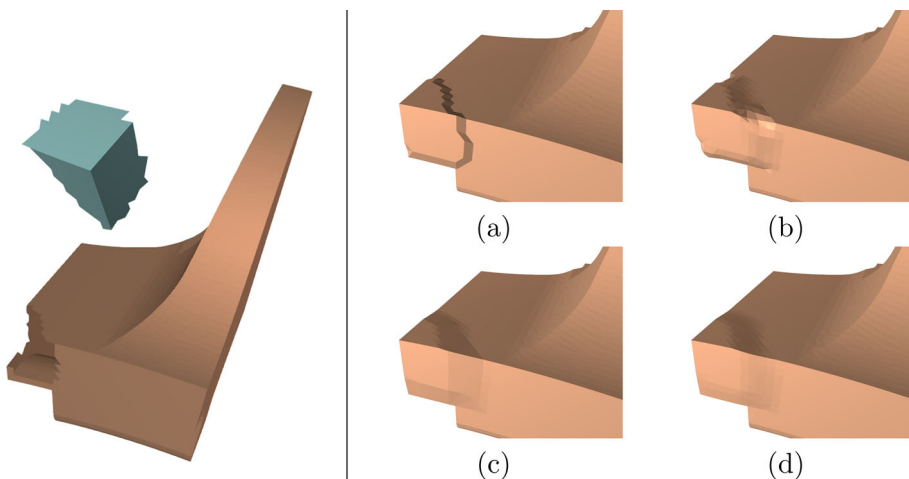


**Fig. 9** Examples of context-aware completion:  $\mathcal{M}_0$  with template patch  $\mathcal{P}$ , output  $\mathcal{M}^*$  for two different incomplete meshes



**Fig. 10** Example of context-aware completion:  $\mathcal{M}_0$  with template patch  $\mathcal{P}$  (left); mask  $M_V$  (middle); output  $\mathcal{M}^*$  from two different camera points of view (right)

**Fig. 11** Examples of different continuity conditions for  $strip(b_0)$  in  $M_E$  for context-aware completion:  $\mathcal{M}_0$  with a hole bounded by  $b_0$  and template  $\mathcal{P}$  (left). Right, zoom onto the completion area  $\mathcal{M}_0 \cup \mathcal{P}$ , i.e.  $V^{(0)}$  (a), output  $\mathcal{M}^*$  for  $M_E(strip(b_0)) \equiv 0$  (b),  $\mathcal{M}^*$  for  $M_E(strip(b_0)) \equiv 1$  (c),  $\mathcal{M}^*$  for  $M_E(strip(b_0))$  estimated from dihedral angles (d)



boundary  $b_{\mathcal{P}}$  was larger than  $b_0$  and slightly shifted. Nevertheless, the feature-adaptive regularization perfectly respects the continuity of  $strip(b_0)$ , as illustrated in Fig. 11d, while a smooth mask—Fig. 11b—destroys the sharp edges, and a non-smooth joint—Fig. 11c—creates artefact features.

### 7 Conclusions

We presented a novel geometric framework for denoising, inpainting and context-based completion for the recovery

of damaged and incomplete scanned data. In contrast to volumetric approaches which use complex data structures and sophisticated procedures, we formulate the solution of the three tasks in terms of a single variational problem which is parameterization-free and normal consistent. The proposed approach is intended to repair damaged and incomplete meshes resulting from range scanning as well as for all modeling operations aiming at replacing damaged or missing parts of the surface. Future investigations will focus on the study of the theoretical convergence of the proposed numerical algorithm, which minimizes a functional



that is spatially variant and characterized by a convex-non-convex structure: it varies spatially from being convex (due to the presence of the Willmore energy) to non-convex (MC penalty) according to the mask operator  $M_E$ . Nevertheless, the algorithm demonstrates empirical convergence and very satisfying practical performance. The encouraging results can further be extended to the completion of missing part of objects and template patches with boundaries of different topology in order to validate the process on more realistic cases. Finally, a future direction will be to couple the proposed approach with image inpainting models favouring the completion of texture-like regions.

**Acknowledgements** Research is supported in part by INDAM-GNCS research project 2020. LC and GR acknowledge the support received by the Laboratory I3S, the UCA JEDI Academy on Complex Systems and the project Arch-AI-story at UCA. LC acknowledges D. Binder, L. Cassard and V. L. Coli (CEPAM, CNRS) for providing the data used for the examples reported in Fig. 8.

**Funding** Open access funding provided by Alma Mater Studiorum - Università di Bologna within the CRUI-CARE Agreement.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Sharf, A., Alexa, M., Cohen-Or, D.: Context-based surface completion. *ACM Trans. Graph.* **23**(3), 878–887 (2004)
- Park, S., Guo, X., Shin, H., Qin, H.: Surface completion for shape and appearance. *Vis. Comput.* **22**(3), 168–180 (2006)
- Bobenko, A.I., Schröder, P.: Discrete willmore flow. SGP'05. Eurographics Association, Goslar, DEU (2005)
- Willmore, T.J.: Surfaces in conformal geometry. *Ann. Global Anal. Geom.* **18**(3), 255–264 (2000)
- Huska, M., Morigi, S., Recupero, G.A.: Sparsity-aided variational mesh restoration. In: Elmoataz, A., Fadili, J., Quéau, Y., Rabin, J., Simon, L. (eds.) *Scale Space and Variational Methods in Computer Vision*, pp. 437–449. Springer, Cham (2021)
- Tasdizen, T., Whitaker, R., Burchard, P., Osher, S.: Geometric surface processing via normal maps. *ACM Trans. Graph.* **22**(4), 1012–1033 (2003)
- Lysaker, M., Osher, S.: Xue-Cheng Tai: Noise removal using smoothed normals and surface fitting. *IEEE Trans. Image Process.* **13**(10), 1345–1357 (2004)
- 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 (2012)
- Wang, P.-S., Liu, Y., Tong, X.: Mesh denoising via cascaded normal regression. *ACM Trans. Graph.* **35**(6) (2016)
- He, L., Schaefer, S.: Mesh denoising via l0 minimization. *ACM Trans. Graph.* **32**(4) (2013)
- Sun, Y., Schaefer, S., Wang, W.: Denoising point sets via l0 minimization. *Comput. Aided Geom. Des.* **35–36**, 2–15 (2015)
- Avron, H., Sharf, A., Greif, C., Cohen-Or, D.:  $\ell_1$ -sparse reconstruction of sharp point set surfaces. *ACM Trans. Graph.* **29**, 135–113512 (2010)
- Clarenz, U., Diewald, U., Dziuk, G., Rumpf, M., Rusu, R.: A finite element method for surface restoration with smooth boundary conditions. *Comput. Aided Geom. Des.* **21**(5), 427–445 (2004)
- Davis, J., Marschner, S.R., Garr, M., Levoy, M.: Filling holes in complex surfaces using volumetric diffusion. In: *Proceedings. First International Symposium on 3D Data Processing Visualization and Transmission*, pp. 428–441 (2002)
- Argudo, O., Brunet, P., Chica, A., Vinacua, À.: Biharmonic fields and mesh completion. *Graph. Models* **82**, 137–148 (2015)
- Casciola, G., Lazzaro, D., Montefusco, L., Morigi, S.: Fast surface reconstruction and hole filling using positive definite radial basis functions. *Numer. Algor.* **39**, 289–305 (2005)
- Wang, J., Oliveira, M.M.: Filling holes on locally smooth surfaces reconstructed from point clouds. *Image Vis. Comput.* **25**(1), 103–113 (2007)
- Sorkine-Hornung, O., Cohen-Or, D.: Least-squares meshes. *Proc. Shape Model. Int. SMI* **2004**, 191–199 (2004)
- Harary, G., Tal, A., Grinspun, E.: Context-based coherent surface completion. *ACM Trans. Graph.* **33**(1) (2014)
- Lanza, A., Morigi, S., Sgallari, F.: Convex image denoising via non-convex regularization with parameter selection. *J. Math. Imaging Vis.* **56**(2), 195–220 (2016)
- Soubies, E., Blanc-Féraud, L., Aubert, G.: A unified view of exact continuous penalties for  $\ell_2$ - $\ell_0$  minimization. *SIAM J. Optim.* **27**(3), 2034–2060 (2017)
- Zhang, C.-H.: Nearly unbiased variable selection under minimax concave penalty. *Ann. Stat.* **38**(2), 894–942 (2010)
- Huska, M., Lanza, A., Morigi, S., Sgallari, F.: Convex non-convex segmentation of scalar fields over arbitrary triangulated surfaces. *J. Comput. Appl. Math.* **349**, 438–451 (2019)
- Clarenz, U., Diewald, U., Dziuk, G., Rumpf, M., Rusu, R.: A finite element method for surface restoration with smooth boundary conditions. *Comput. Aided Geom. Des.* **21**, 427–445 (2004)
- Hari, L.P., Givoli, D.J.: Computation of open willmore-type surfaces. *Appl. Numer. Math.* **37**, 257–269 (2001)
- Bergou, M., Wardetzky, M., Harmon, D., Zorin, D., Grinspun, E.: Discrete quadratic curvature energies. In: *ACM SIGGRAPH 2006 Courses. SIGGRAPH '06*, pp. 20–29. Association for Computing Machinery, New York, NY, USA (2006)
- Bobenko, A.I.: A conformal energy for simplicial surfaces. *Combinat. Comput. Geom.* **52**, 133–143 (2005)
- Schoenemann, T., Masnou, S., Cremers, D.: On a linear programming approach to the discrete willmore boundary value problem and generalizations. In: Boissonnat, J.-D., Chenin, P., Cohen, A., Gout, C., Lyche, T., Mazure, M.-L., Schumaker, L. (eds.) *Curves and Surfaces*, pp. 629–646. Springer, Berlin, Heidelberg (2012)
- Seung, H.S., Nelson, D.R.: Defects in flexible membranes with crystalline order. *Phys. Rev. A* **38**, 1005–1018 (1988)
- Meyer, M., Desbrun, M., Schröder, P., Barr, A.H.: Discrete differential-geometry operators for triangulated 2-manifolds. In: Hege, H.-C., Polthier, K. (eds.) *Visualization and Mathematics III*, pp. 35–57. Springer, Berlin, Heidelberg (2003)



31. Schmidt, B., Fraternali, F.: Universal formulae for the limiting elastic energy of membrane networks. *J. Mech. Phys. Solids* **60**(1), 172–180 (2012)
32. Huska, M., Lanza, A., Morigi, S., Selesnick, I.: A convex-nonconvex variational method for the additive decomposition of functions on surfaces. *Inver. Prob.* **35**(12), 124008 (2019)
33. Zheng, Y., Fu, H., Au, O.K., Tai, C.: Bilateral normal filtering for mesh denoising. *IEEE Trans. Visualiz. Comput. Graph.* **17**(10), 1521–1530 (2011)
34. Sun, X., Rosin, P.L., Martin, R., Langbein, F.: Fast and effective feature-preserving mesh denoising. *IEEE Trans. Vis. Comp. Graph.* **13**(5), 925–938 (2007)
35. Liu, Z., Lai, R., Zhang, H., Wu, C.: Triangulated surface denoising using high order regularization with dynamic weights. *SIAM J. Sci. Comp.* **41**(1), 1–26 (2019)
36. Zavriev, S.K., Kostyuk, F.V.: Heavy-ball method in nonconvex optimization problems. *Comput. Math. Model.* **4**, 336–341 (1993)
37. Ochs, P., Chen, Y., Brox, T., Pock, T.: ipiano: Inertial proximal algorithm for non-convex optimization. *SIAM J. Imag. Sci.* **7**(2), 1388–1419 (2014)
38. Wang, Y., Yin, W., Zeng, J.: Global convergence of admm in non-convex nonsmooth optimization. *J. Sci. Comput.* **78**(12), 29–63 (2019)
39. Zhang, W., Deng, B., Zhang, J., Bouaziz, S., Liu, L.: Guided mesh normal filtering. *Comput. Graph. Forum* **34**(7), 23–34 (2015)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.