Digital surface regularization by normal vector field alignment

David Coeurjolly, Pierre Gueth, Jacques-Olivier Lachaud

Digital surface regularization by normal vector field alignment

David Coeurjolly[1], Pierre Gueth[1], and Jacques-Olivier Lachaud[2]

[1] Université de Lyon, CNRS, LIRIS UMR 5205, F-69622, France
[2] Laboratoire de Mathématiques, CNRS, UMR 5127, University Savoie Mont Blanc, France

Abstract. Digital objects and digital surfaces are isothetic structures *per se*. Such surfaces are thus not adapted to direct visualization with isothetic quads, or to many geometry processing methods. We propose a new regularization technique to construct a piecewise smooth quadrangulated surface from a digital surface. More formally we propose a variational formulation which efficiently regularizes digital surface vertices while complying with a prescribed, eventually anisotropic, input normal vector field estimated on the digital structure. Beside visualization purposes, such regularized surface can then be used in any geometry processing tasks which operates on triangular or quadrangular meshes (e.g. compression, texturing, anisotropic smoothing, feature extraction).

1 Introduction

Objective. This paper addresses the problem of approximating the boundary of an object given as a 3d *binary* image (see Fig. 4 top for an example). Such input data is called a 3d digital object and its surface is called a digital surface. They arise as digitizations of continuous objects and segmentation of 3d images. Compared to gray-level volumetric images, such binary image model is much more complex to handle: the function takes only two values, it is not continuous and of course has no gradient, it is known only at regularly sampled places. For this data, direct use of former methods gives triangulated meshes with poor geometry (staircasing effects, few possible orientations).

Contribution. We propose an original method to construct a piecewise smooth approximation of the boundary of a digital object. It follows a convex variational principle that regularizes the digital surface according to three criteria: First, it must stay close to the input data; second, it should comply to a well-chosen normal vector field $\mathbf{u}$; third, its cells must be as regular as possible. Our method is not iterative and provides excellent piecewise smooth reconstructions when the normal vector field $\mathbf{u}$ is a good approximation of the normal vector field of the original continuous object. This is why we use recent digital normal vector estimators which offer multigrid convergence guarantees [5, 9] and may also detect normal discontinuities [4]. By this way, we are able to take into account the discrete and arithmetic nature of digital objects while using the powerful framework of convex optimization. Experiments show that the output surface is aligned with the prescribed normal vector field and is closer to the underlying continuous object than was the input digitized surface. Sharp features are nicely delineated and the quality of mesh faces is very good.

Related works. First of all, the numerous methods that extract isosurfaces from a volume image or function provide poor results with such input data, since they rely on the exact value of the image / function to determine the position of vertices. This includes marching cubes, dual marching cubes and the many variants designed to reconstruct meshes from implicit surfaces [12, 14, 13]. Standard mesh denoising methods could also be considered for removing staircasing effects of digital surfaces [8, 15–17]. However, they tend either to consider all steps as features or to smooth everything out. Very few approaches take into account the peculiar nature of digital data. In 2d, we may quote early works for digital contour polygonalization, which use digital straightness properties to align digital points onto their estimated tangent line [2]. In 3d, reversible polyhedrization of digital surfaces can be achieved with greedy digital plane segmentation followed by Marching-Cubes sewing [3]. Although they are theoretically reversible, none of these techniques achieve similar visual quality compared to our proposal (see Figure 3) . Note that we experimentally show that our regularized surface is also very close to the original digital one.

Outline. Section 2 gives the necessary notions and notations of digital geometry used throughout the paper. Section 3 presents the variational formulation of our digital surface regularization method. The convex optimization algorithm is described in Section 4. Experiments illustrating the qualities of the method are given in Section 5, as well as a quantitative asymptotic analysis of the surface reconstruction. Limitations and possible extensions of this method are discussed in Section 6.

2 Preliminaries

A 3d binary image is simply a function from $\mathbb{Z}^3$ toward $\{0, 1\}$. The set $Z \subset \mathbb{Z}^3$ of digital points where the function is not null represents the digital object. It is naturally embedded in $\mathbb{R}^3$ as the union of unit cubes centered on these digital points. Its topological boundary is called its digital boundary, and is a special case of digital surface. Since a digital surface is a set of unit squares sewn together, it has a cellular representation in cartesian cubical grid as a set $\mathbb{E}^0$ of $0-$cells (vertices), a set $\mathbb{E}^1$ of $1-$cells (edges), and a set $\mathbb{E}^2$ of $2-$cells (square faces). We do not detail the topological combinatorial structure, *à-la* Alexandrov, associated with subsets of the Cartesian cubic grid (and more specifically digital surfaces). Interested readers may refer to [10]. We just assume in the following that $(\mathbb{E}^0, \mathbb{E}^1, \mathbb{E}^2)$ defines a proper combinatorial 2-manifold without boundaries.

Our objective is to reconstruct a piecewise smooth surface from a digital boundary. Therefore we have in mind that the digital shape comes from some digitization or sampling of a continuous shape X, and our objective is to infer the original shape boundary ∂X with solely its digitization as input. The *multigrid convergence* framework is thus well adapted to evaluate objectively the qualities of our method (e.g. see [10, 11]). For some continuous shape $X \subset \mathbb{R}^3$ and a digitization process parameterized by a sampling grid step h, we will evaluate if our regularized surface $\partial_h^* X$ (in red in Figure 1) is closer to the continuous surface ∂X than the digitized boundary $\partial_h X$ (in orange). And of course, we wish that the finer is the resolution ($h \to 0$) the better is the approximation. Classical notations are recalled in Figure 1. It is already known that, for the Gauss digitization (denoted $\mathtt{G}_h(X)$ in Figure 1), the digitized boundary $\partial_h X$ of a compact subset X of $\mathbb{R}^d$ with positive reach is Hausdorff close to ∂X, with a distance no greater than $\sqrt{d}h/2$ [11]. Hence our regularized surface $\partial_h^* X$ should also be Hausdorff close to ∂X. In Section 5, we show that this is indeed the case, and with a better constant than $\frac{\sqrt{3}}{2}h$.

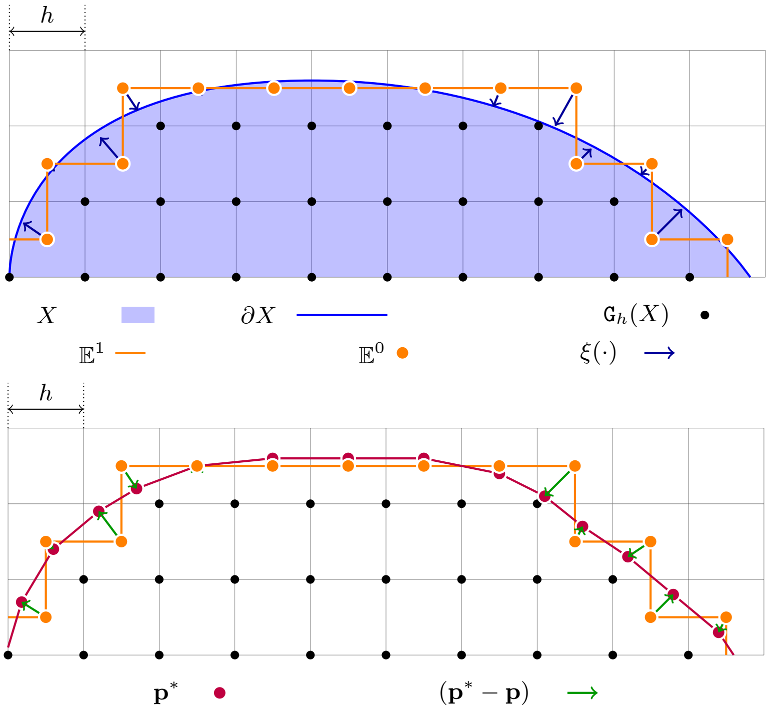

Fig. 1. Illustration of the digitization models and notations in dimension 2. $\xi(\mathbf{y})$ maps $\mathbf{y} \in \mathbb{R}^2 \setminus MA(X)$ to the closest point on ∂X ($MA(X)$ being the medial axis of X).

Furthermore, the normals of the regularized surface $\partial_h^* X$ should also tend toward the normal of the original continuous surface ∂X. We achieve this property by adding as input to our regularization process a normal vector field associated to faces of the input digital surface. By choosing normal estimates given by some multigrid convergent normal estimator (like [5, 9]), our variational formulation makes the normals of the regularized surface $\partial_h^* X$ align with the normals of ∂X.

3 Variational formulation

Our variational model will simply move the vertices $\mathbb{E}^0$ of the input digital surface in order to regularize it. Although the cellular topology of the digital surface is used in the process, the output regularized surface has the same cellular topology as the input digital surface.

In the following, we denote $\mathbf{p} := \{\mathbf{p}_i\}$ the canonical coordinates in $\mathbb{R}^3$ of the $0-$cells $\{\sigma_i^0\}$ in the embedding Euclidean space. Similarly we introduce $\mathbf{p}^* := \{\mathbf{p}_i^*\} \in \mathbb{R}^3$ the set of regularized points coordinates in $\mathbb{R}^3$ associated with each $0-$cell. Furthermore, we denote by $\mathbf{u} := \{\mathbf{u}_k\}$ the input discrete normal vector field associated with the faces of the cubical complex (vector $\mathbf{u}_k$ is the estimated normal of the cell σ_k^2). In the following sections, we provide more details about the structure of these sets allowing us to define a reliable and efficient calculus on the digital surface. We also consider $\mathbf{S}$ as the embedded quadrangulated surface associated with the cubical complex (with vertices $\mathbf{p}$, edges induced by $\mathbb{E}^1$ and faces by $\mathbb{E}^2$). We denote by $\mathbf{S}^*$ the quadrangulated surface with the same structure as $\mathbf{S}$ with vertices $\mathbf{p}^*$. Please note that once vertices are regularized, the quads of $\mathbf{S}^*$ may not be planar anymore. The surface $\mathbf{S}$ is, by definition, $\partial_h X$ if we are considering the digitization of a shape X. The surface $\mathbf{S}^*$ then corresponds to $\partial_h^* X$ in this case.

Before formally defining $\mathbf{p}^*$, let us consider a given point set $\hat{\mathbf{p}} \subset \mathbb{R}^3$ and its associated quadrangulation $\hat{\mathbf{S}}$. We evaluate the energy associated with this point set. First, we want $\hat{\mathbf{p}}$ to be close to $\mathbf{p}$ (*data attachment*). Then, we want the discrete normal vector field associated with $\hat{\mathbf{p}}$ to comply with the input vector field $\mathbf{u}$. Since quads associated with $\hat{\mathbf{S}}$ may not be planar, we place this constraint onto the edges of each quad: we want each edge $\mathbf{d}\hat{\mathbf{p}}_j$ (where $\mathbf{d}\hat{\mathbf{p}}_j := \hat{\mathbf{p}}_{j_0} - \hat{\mathbf{p}}_{j_1}$ with $\partial\sigma_j^1 = \{\sigma_{j_0}^0, \sigma_{j_1}^0\}$, $\sigma_{j_0}^0$ being the tip of edge σ_j^1 and $\sigma_{j_1}^0$ being the origin of edge σ_j^1)[3] to be as orthogonal as possible to each neighboring face normal vector $\mathbf{u}_k$ (*normal vector alignment*). Those constraints are handled by the first and second terms of the following energy function :

$$\mathcal{E}(\hat{\mathbf{p}}) := \alpha \underbrace{\|\hat{\mathbf{p}} - \mathbf{p}\|^2}_{\mathcal{E}_{\text{data}}(\hat{\mathbf{p}})} + \beta \underbrace{\|\mathbf{d}\hat{\mathbf{p}} \cdot \mathbf{u}\|^2}_{\mathcal{E}_{\text{align}}(\hat{\mathbf{p}})} + \gamma \underbrace{\|\hat{\mathbf{p}} - \hat{\mathbf{b}}\|^2}_{\mathcal{E}_{\text{fairness}}(\hat{\mathbf{p}})} . \quad (1)$$

where $\hat{\mathbf{b}}_i$ holds the coordinates of the barycenter of the neighboring vertices of $\mathbf{p}_i$. The last term (*fairness*) ensures that vertices of $\hat{\mathbf{S}}$ are well distributed along the surface: it moves the points onto their tangent planes so that the sampling is

[3] For readers familiar with Discrete Exterior Calculus (DEC), $\mathbf{d}$ is similar to an exterior derivative operator on (triplets of) primal $0-$forms of the cubical complex.

as regular as possible. In sections 3.1, 3.2 and 3.3, we detail the norms involved in each term of this functional.

From this formulation, we define the optimal regularized coordinates $\mathbf{p}^*$ as

$$\mathbf{p}^* := \underset{\hat{\mathbf{p}}}{\operatorname{argmin}}\ \mathcal{E}(\hat{\mathbf{p}})\,. \tag{2}$$

We detail now each energy term in a discrete calculus setting, which allows an efficient minimization of (2) (see Section 4).

3.1 Data attachment term

Since $\mathbf{p}$ contains point coordinates, it can be interpreted as a triplet of maps $\mathbf{p}_x$, $\mathbf{p}_y$ and $\mathbf{p}_z$ from $\mathbb{E}^0$ to $\mathbb{R}$ containing the vertex coordinates in embedding space. Thanks to a numbering of cells in $\mathbb{E}^0$, we use a vector representation of $\mathbf{p}$ as a single column vector concatenating vectors associated with $\mathbf{p}_x$, $\mathbf{p}_y$ and $\mathbf{p}_z$. In other words $\mathbf{p} = [\mathbf{p}_x^T, \mathbf{p}_y^T, \mathbf{p}_z^T]^T$. Data attachement term keeps $\mathbf{p}^*$ close to $\mathbf{p}$ and guarantees the convexity of the problem. It is defined as follows:

$$\mathcal{E}_{\text{data}}(\hat{\mathbf{p}}) := \|\hat{\mathbf{p}} - \mathbf{p}\|^2\,. \tag{3}$$

The norm in (3) is the norm of discrete $\mathbb{E}^0 \mapsto \mathbb{R}^3$ maps and is defined from the scalar products between discrete $\mathbb{E}^0 \mapsto \mathbb{R}$ maps $\langle \bullet, \bullet \rangle_0$:

$$\|\hat{\mathbf{p}} - \mathbf{p}\|^2 = \|\mathbf{a}\|^2 = \langle \mathbf{a}_x, \mathbf{a}_x \rangle_0 + \langle \mathbf{a}_y, \mathbf{a}_y \rangle_0 + \langle \mathbf{a}_z, \mathbf{a}_z \rangle_0\ . \tag{4}$$

To shorten notation in the previous equation, $\mathbf{a}$ is the difference map $(\hat{\mathbf{p}} - \mathbf{p})$.

One can simply consider classical Euclidean scalar products between vectors in $\mathbb{R}^{|\mathbb{E}^0|}$ to define (4). In Section 3.4, we propose an alternative definition which is more consistent with discrete calculus on combinatorial structures.

3.2 Normal vector alignment term

The second term is the most complex one and tends to orthogonalize the direction of each edge with adjacent face normal vectors:

$$\mathcal{E}_{\text{align}}(\hat{\mathbf{p}}) := \|\mathbf{d}\hat{\mathbf{p}} \cdot \mathbf{u}\|^2\,. \tag{5}$$

First let us look at $\mathbf{d}\hat{\mathbf{p}} : \mathbb{E}^1 \mapsto \mathbb{R}^3$ (see definition in Section 3). There exists a linear operator $\mathbf{D} : (\mathbb{E}^0 \mapsto \mathbb{R}^3) \mapsto (\mathbb{E}^1 \mapsto \mathbb{R}^3)$ such that

$$\mathbf{d}\hat{\mathbf{p}} = \mathbf{D}\,\hat{\mathbf{p}}\,. \tag{6}$$

Thanks to the linearization of $\hat{\mathbf{p}}$ as a $3|\mathbb{E}^0|$ vector, such linear operator can be represented as a $3|\mathbb{E}^1| \times 3|\mathbb{E}^0|$ matrix. $\mathbf{d}\hat{\mathbf{p}} \cdot \mathbf{u} : \mathbb{E}^2 \mapsto \mathbb{R}^4$ holds the scalar products for each edge adjacent to all faces in embedding space:

$$(\mathbf{d}\hat{\mathbf{p}} \cdot \mathbf{u})_k = (\mathbf{d}\hat{\mathbf{p}}_{k_0} \cdot \mathbf{u}_k, \mathbf{d}\hat{\mathbf{p}}_{k_1} \cdot \mathbf{u}_k, \mathbf{d}\hat{\mathbf{p}}_{k_2} \cdot \mathbf{u}_k, \mathbf{d}\hat{\mathbf{p}}_{k_3} \cdot \mathbf{u}_k) \tag{7}$$

where $\partial\sigma^2_k = \{\sigma^1_{k_0}, \sigma^1_{k_1}, \sigma^1_{k_2}, \sigma^1_{k_3}\}$ (always four edges in face border).

Furthermore, there exists a linear operator $\mathbf{U} : (\mathbb{E}^1 \mapsto \mathbb{R}^3) \mapsto (\mathbb{E}^2 \mapsto \mathbb{R}^4)$ such that :

$$\mathbf{d}\hat{\mathbf{p}} \cdot \mathbf{u} = \mathbf{U}\,\mathbf{D}\,\hat{\mathbf{p}}\,. \tag{8}$$

Again, such operator can be represented as a $4|\mathbb{E}^2| \times 3|\mathbb{E}^1|$ matrix in our discrete calculus.

The norm in (5) is the norm of discrete $\mathbb{E}^2 \mapsto \mathbb{R}^4$ maps and is defined from the scalar products between discrete $\mathbb{E}^2 \mapsto \mathbb{R}$ maps $\langle \bullet, \bullet \rangle_2$:

$$\|\mathbf{d}\hat{\mathbf{p}} \cdot \mathbf{u}\|^2 = \|\mathbf{b}\|^2 = \langle \mathbf{b}_{k_0}, \mathbf{b}_{k_0} \rangle_2 + \langle \mathbf{b}_{k_1}, \mathbf{b}_{k_1} \rangle_2 + \langle \mathbf{b}_{k_2}, \mathbf{b}_{k_2} \rangle_2 + \langle \mathbf{b}_{k_3}, \mathbf{b}_{k_3} \rangle_2\ . \tag{9}$$

In the previous equation, $\mathbf{b}$ is a shorthand for $\mathbf{d}\hat{\mathbf{p}} \cdot \mathbf{u}$. Finally, the alignment term is thus simply expressed as

$$\mathcal{E}_{\text{align}}(\hat{\mathbf{p}}) = \|\mathbf{U}\,\mathbf{D}\,\hat{\mathbf{p}}\|^2\ . \tag{10}$$

3.3 Fairness term

The last term is the fairness term which tends to flatten the regularized complex and to distribute the vertex positions with tangential displacements (see Figure 2):

$$\mathcal{E}_{\text{fairness}}(\hat{\mathbf{p}}) := \|\hat{\mathbf{p}} - \hat{\mathbf{b}}\|^2\,. \tag{11}$$

The norm is the same as in (3) and relies on the same scalar product $\langle \bullet, \bullet \rangle_0$. $\hat{\mathbf{b}}_i$ is the barycenter of neighboring vertices to $\mathbf{p}_i$. More formally[4] :

$$\hat{\mathbf{b}}_i := \frac{1}{|\operatorname{link}(\sigma^0_i)|} \sum_{\sigma^0_j \in \operatorname{link}(\sigma^0_i)} \hat{\mathbf{p}}_j\ . \tag{12}$$

This defines $\mathbf{B} : (\mathbb{E}^0 \mapsto \mathbb{R}^3) \mapsto (\mathbb{E}^0 \mapsto \mathbb{R}^3)$ a linear operator from and to $\mathbb{E}^0 \mapsto \mathbb{R}^3$ ($3|\mathbb{E}^0| \times 3|\mathbb{E}^0|$ matrix) allowing us to write $\hat{\mathbf{b}}$, expressed as a linearized column vector of positions, as the matrix-vector multiplication,

$$\hat{\mathbf{b}} := \mathbf{B}\,\hat{\mathbf{p}}\,. \tag{13}$$

Note that matrix $\mathbf{B}$ is sparse which leads to efficient factorization (see Section 4). The fairness term reduces to

$$\mathcal{E}_{\text{fairness}}(\hat{\mathbf{p}}) = \|(\mathbf{I} - \mathbf{B})\hat{\mathbf{p}}\|^2\,, \tag{14}$$

where $\mathbf{I}$ is the identity operator (identity matrix $3|\mathbb{E}^0| \times 3|\mathbb{E}^0|$).

[4] $\operatorname{link}(\sigma^0_i)$ is the link operator on cubical complexes. As a consequence, $0-$cells in this set are *connected* to σ^0_i by a $1-$cell in the complex.

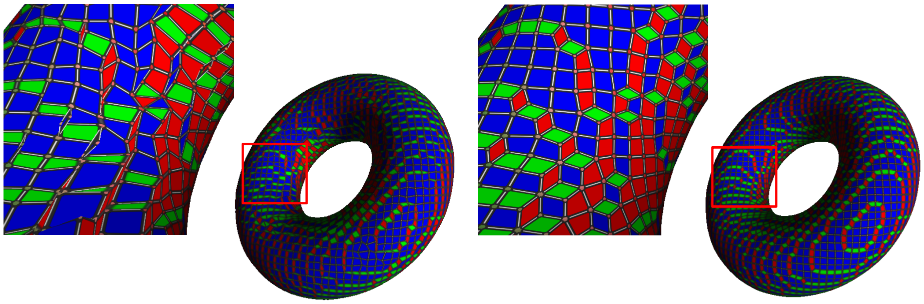

Fig. 2. Impact of the $\mathcal{E}_{\text{fairness}}$ term on the regularized quadrangulation (without on the left, with on the right).

3.4 Scalar products

In previous definitions, scalar products $\langle \bullet, \bullet \rangle_0$ and $\langle \bullet, \bullet \rangle_2$ must be specified. More precisely, we need to specify the metric tensor (definite positive, therefore symmetric matrix) $\mathbf{W}_0$ (resp. $\mathbf{W}_2$) associated with maps $\mathbf{x}$, $\mathbf{y} : \mathbb{E}^0 \mapsto \mathbb{R}$ (resp. maps $\mathbf{u}, \mathbf{v} : \mathbb{E}^2 \mapsto \mathbb{R}$):

$$\langle \mathbf{x}, \mathbf{y} \rangle_0 := \mathbf{x}^T \, \mathbf{W}_0 \, \mathbf{y} \,, \tag{15}$$

$$\langle \mathbf{u}, \mathbf{v} \rangle_2 := \mathbf{u}^T \, \mathbf{W}_2 \, \mathbf{v} \,. \tag{16}$$

A simple choice consists in considering identity matrices for $\mathbf{W}_0$ and $\mathbf{W}_2$. However, specific weights can be set if embedding priors of the digital surface are known (see [6] for a complete discussion and Section 6).

Since we consider triplets or quadruple of respectively $0-$ and $2-$forms in previous formulations (due to the linearization of positions into a single vector and the linearization of the four dot products associated with each face edges), we need to extend scalar products of vector is $\mathbb{R}^{|\mathbb{E}^0|}$ to vectors in $\mathbb{R}^{3|\mathbb{E}^0|}$(respectively to vectors in $\mathbb{R}^{4|\mathbb{E}^2|}$). We simply define

$$\bar{\mathbf{W}}_0 := \begin{bmatrix} \mathbf{W}_0 & 0 & 0 \\ 0 & \mathbf{W}_0 & 0 \\ 0 & 0 & \mathbf{W}_0 \end{bmatrix}, \bar{\mathbf{W}}_2 := \begin{bmatrix} \mathbf{W}_2 & 0 & 0 & 0 \\ 0 & \mathbf{W}_2 & 0 & 0 \\ 0 & 0 & \mathbf{W}_2 & 0 \\ 0 & 0 & 0 & \mathbf{W}_2 \end{bmatrix}. \tag{17}$$

4 Energy minimization

Since (1) is convex, we can compute $\mathbf{p}^*$ as the unique solution to

$$\nabla_{\hat{\mathbf{p}}} \mathcal{E}(\mathbf{p}^*) = \alpha \; \nabla_{\hat{\mathbf{p}}} \mathcal{E}_{\text{data}}(\mathbf{p}^*) + \beta \; \nabla_{\hat{\mathbf{p}}} \mathcal{E}_{\text{align}}(\mathbf{p}^*) + \gamma \; \nabla_{\hat{\mathbf{p}}} \mathcal{E}_{\text{fairness}}(\mathbf{p}^*) = 0 \,. \tag{18}$$

From (3), (10) and (14) gradients of each energy term can be expressed as follows:

$$\nabla_{\hat{\mathbf{p}}}\mathcal{E}_{\text{data}}(\hat{\mathbf{p}}) = 2\bar{\mathbf{W}}_0(\hat{\mathbf{p}} - \mathbf{p})\,, \tag{19}$$

$$\nabla_{\hat{\mathbf{p}}}\mathcal{E}_{\text{align}}(\hat{\mathbf{p}}) = 2\,\mathbf{D}^T\,\mathbf{U}^T\,\bar{\mathbf{W}}_2\,\mathbf{U}\,\mathbf{D}\,\hat{\mathbf{p}}\,, \tag{20}$$

$$\nabla_{\hat{\mathbf{p}}}\mathcal{E}_{\text{fairness}}(\hat{\mathbf{p}}) = 2(\mathbf{I} - \mathbf{B})^T\bar{\mathbf{W}}_0(\mathbf{I} - \mathbf{B})\hat{\mathbf{p}}\,. \tag{21}$$

Weighting up all these gradients in Eq. (18) leads to the following linear system in $\mathbf{p}^*$:

$$\mathbf{R}\,\mathbf{p}^* = \alpha\bar{\mathbf{W}}_0\mathbf{p}\,, \tag{22}$$

with

$$\mathbf{R} := \Big(\alpha\bar{\mathbf{W}}_0 + \beta\,\mathbf{D}^T\,\mathbf{U}^T\,\bar{\mathbf{W}}_2\,\mathbf{U}\,\mathbf{D} + \gamma(\mathbf{I} - \mathbf{B})^T\bar{\mathbf{W}}_0(\mathbf{I} - \mathbf{B})\Big)\,. \tag{23}$$

As long as weights are strictly positive, the operator $\mathbf{R}$ is a $3|\mathbb{E}^0| \times 3|\mathbb{E}^0|$ matrix, which is symmetric definite-positive (see Appendix A). The linear system (22) is thus efficiently solved using classical linear algebra solvers [7]. In our experiments, we have used the following rules to balance the energy terms:

$$\beta = 1,\ 0 < \alpha \lll \gamma \lll \beta\,.$$

We choose parameters with different order of magnitudes to sequence the effect of the minimization of the three energy terms of Eq.(23). First, we wish to build a smooth surface leading to the highest value for β, the alignment term. Second, the α energy term is necessary to achieve uniqueness of the optimization problem. Its lowest order of magnitude ensures that the result is very close to the set of solutions of the alignment term alone. Finally, the user has freedom for the γ energy term, depending on the desired regularity of the output quadrangulation (see Fig. 2).

5 Experiments

In this section, we evaluate the quality of the regularization. We have considered two different normal vector fields as input: the first one is given by Integral Invariant approaches [5] and is known to be multigrid convergent. The second correspond to a piecewise smooth reconstruction of the normal vector field using [4]. This approach performs a normal vector field regularization while detecting and preserving sharp features. The discrete operators have been implemented using the DGtal DEC package [1] with Eigen backend for the linear algebra solver [7].

In Figure 4 we first illustrate the overall regularization on a `Standford-bunny` object using an Integral Invariant based normal vector field. In Figure 5, we show that using an anisotropic, piecewise smooth normal vector field from [4], the regularized surface is able to capture sharp features. In Figure 6, we demonstrate the robustness of the regularization in presence of noise. Note that beside the fact that the input normal vector field is robust for the alignment term, the fairness term allows us to obtain a smooth quadrangulation even in this case.

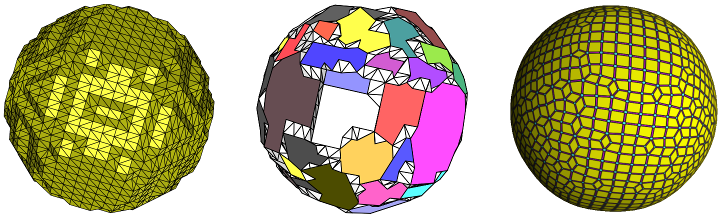

Fig. 3. Regularization example on a digital sphere ($r = 10$). From left to right the original Marching-Cubes surface, the simplified one using [3] and our regularization.

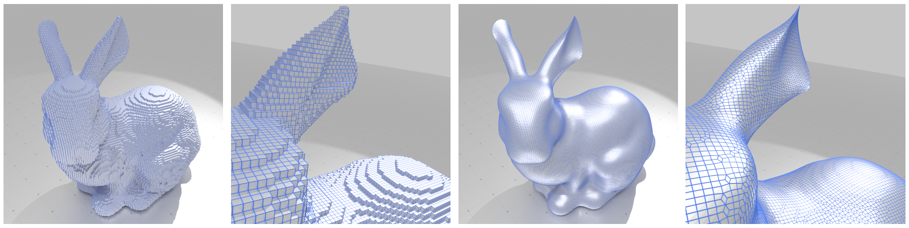

Fig. 4. Regularization example on a 128^3 `Stanford-bunny`. From left to right, input digital surface and regularized surface using the input normal vector field from [5] ($r = 6$).

If not specified, $\alpha = 10^{-3}$, $\beta = 1$ and $\gamma = 10^{-2}$ values have been used in these tests.

We also evaluate the asymptotic behavior of the regularization operator. As discussed in Section 2, we know that the Hausdorff distance between $\mathbf{S}$ and ∂X for some smooth shape X is in $O(h)$. We experimentally observe that $\mathbf{S}^*$ has the same asymptotic behavior and is even closer to the original surface than $\mathbf{S}$. We have considered asymptotic plane and sphere objects in Figure 7 for various h (abscissa) tending to zero. In the first row, we show that the normal vector field $\mathbf{u}^*$ of the regularized surface $\mathbf{S}^*$ seems to converge to the input field $\mathbf{u}$ as h tends to zero[5]. The second row shows the proximity of $\mathbf{p}$ (i.e. the vertices of $\mathbf{S} = \partial_h X$) to ∂X (Theorem 1 of [11]). The $O(h)$ convergence speed is thus experimentally confirmed. In the third row, we evaluate the distance between $\mathbf{p}^*$ (the vertices of $\mathbf{S}^*$) and its closest point on ∂X. We also observe an experimental convergence speed in $O(h)$. Finally, the last row compares the two approximations. In this case, we see that $\mathbf{p}^*$ is experimentally closer to ∂X than $\mathbf{p}$.

[5] Since quads of $\mathbf{S}^*$ are not coplanar anymore, each vector of $\mathbf{u}^*$ is defined by averaging the two normal vectors of the quad triangles.

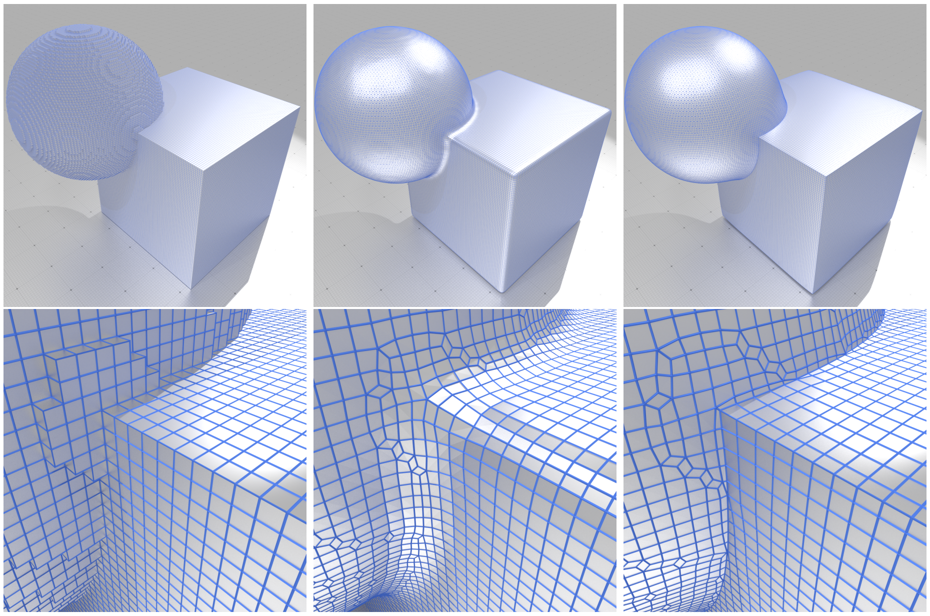

Fig. 5. Isotropic vs anistropic normal vector field as input data. *From left to right*: the original object, the regularization with Integral Invariant based normal vector field [5] ($r = 4$), the regularization after a piecewise smoothing of the same normal vector field using [4] ($\epsilon = [2, 0.25]$, $\lambda = 0.03$, $\alpha = 0.006$ and the Integral Invariant normal vector field with $r = 4$).

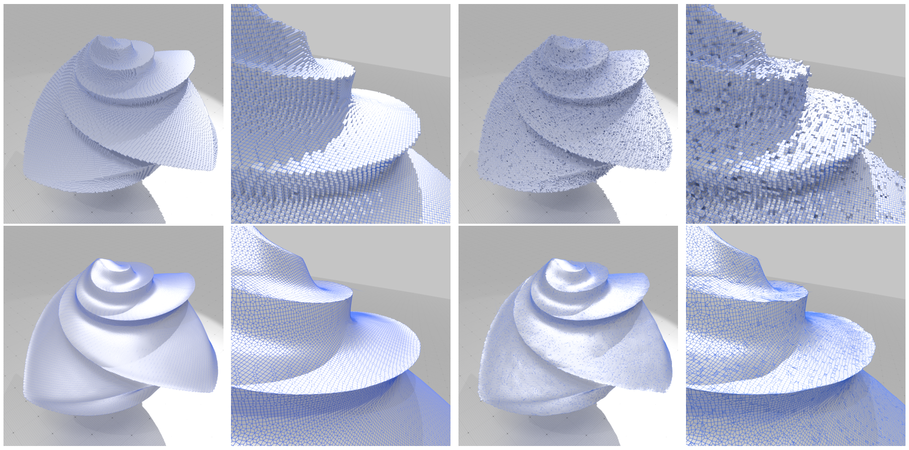

Fig. 6. Regularization examples without (*left*) or with noise *(right)* (`Octaflower`, 256^3). A highly specular surface has been used to highlight the smoothness of the reconstruction (normal vectors are given by the face geometry). The input normal vector field is given by [4].

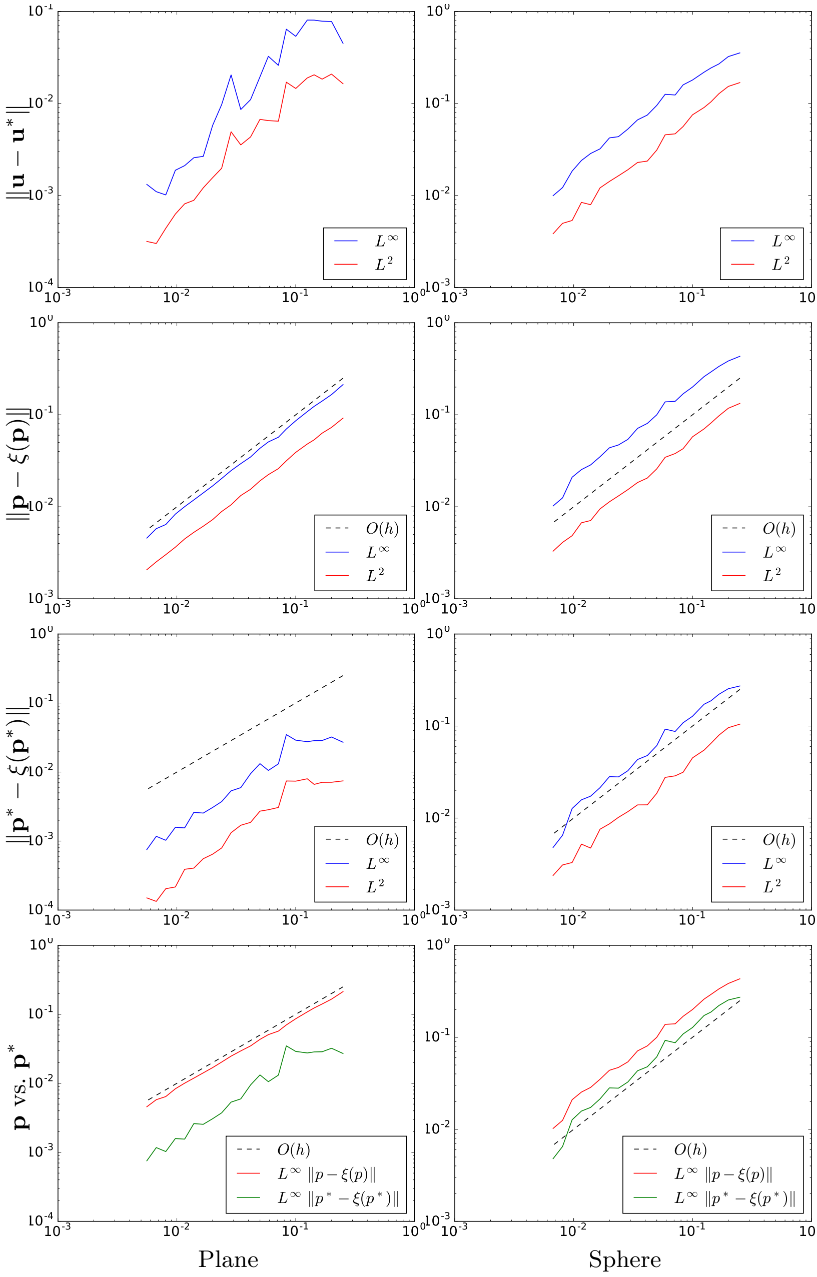

Fig. 7. Convergence results for plane and sphere with $\alpha = 10^{-5}h^2, \beta = 1, \gamma = 0$. Errors ($y-$axis) are given for both the L^2 and L^∞ norms. The grid resolution h (in abscissa) in the range $[6 \cdot 10^{-3}, 0.2]$. As h tends to zero, convergence graphs must be read from right to left.

6 Conclusion and future works

In this article, we have proposed a variational approach to regularize a digital surface. The regularized surface is consistent with respect to an input normal vector field, and has a smooth embedding. If the input normal vector field is piecewise smooth (*i.e.* with singularities), the regularization preserves these features. Finally, we have experimentally demonstrated that the regularized vertices are closer to the underlying continuous object than the Gauss digitization in a multigrid convergent framework.

Future works are twofold: First, similarly to Theorem 1 of [11], a formal proximity result is needed between $\hat{\mathbf{p}}$ and ∂X. At this point, we have been able to derived a formal proof for an L^2 proximity in $O(h)$ (average error). Further developments are required to obtain a L^∞ proximity in $O(h)$ (worst-case). Secondly, we would like to exploit the regularity and smoothness of the regularized surface in various geometry processing problems. Last, we would like to study the influence of metrics $\bar{\mathbf{W}}_0$ and $\bar{\mathbf{W}}_2$ in the quality of reconstruction. For now we have only used identity matrices, but we believe that metrics based on estimated local areas would improve results.

A Details on operator R

The $\mathbf{R}$ operator defined in (23) is a sum of three symmetric matrices,

$$\alpha\bar{\mathbf{W}}_0 + \beta\,\mathbf{D}^T\,\mathbf{U}^T\,\bar{\mathbf{W}}_2\,\mathbf{U}\,\mathbf{D} + \gamma(\mathbf{I}-\mathbf{B})^T\bar{\mathbf{W}}_0(\mathbf{I}-\mathbf{B})\,. \tag{24}$$

The first one is a positive definite matrix since it stands for a scalar product. From (5) and (11), one can clearly see that $\mathcal{E}_{\text{align}} \geq 0$ and $\mathcal{E}_{\text{fairness}} \geq 0$ as they are calculated as the sum of positive or null terms. Noting that $\mathcal{E}_{\text{align}}(\hat{\mathbf{p}}) = \hat{\mathbf{p}}^T\,\mathbf{D}^T\,\mathbf{U}^T\,\bar{\mathbf{W}}_2\,\mathbf{U}\,\mathbf{D}\,\hat{\mathbf{p}}$ and $\mathcal{E}_{\text{fairness}}(\hat{\mathbf{p}}) = \hat{\mathbf{p}}^T(\mathbf{I}-\mathbf{B})^T\bar{\mathbf{W}}_0(\mathbf{I}-\mathbf{B})\hat{\mathbf{p}}$, the second and third matrices are therefore positive semi-definite. Therefore we have $\forall \mathbf{x} \in \mathbb{R}^{3|\mathbb{E}^0|} \setminus \{0\}$:

$$\mathbf{x}^T\left(\alpha\bar{\mathbf{W}}_0\right)\mathbf{x} > 0\,, \tag{25}$$

$$\mathbf{x}^T\left(\beta\,\mathbf{D}^T\,\mathbf{U}^T\,\bar{\mathbf{W}}_2\,\mathbf{U}\,\mathbf{D}\right)\mathbf{x} \geq 0\,, \tag{26}$$

$$\mathbf{x}^T\left(\gamma(\mathbf{I}-\mathbf{B})^T\bar{\mathbf{W}}_0(\mathbf{I}-\mathbf{B})\right)\mathbf{x} \geq 0\,. \tag{27}$$

Assuming $\alpha > 0$, $\beta > 0$ and $\gamma > 0$, we have $\mathbf{x}^T\,\mathbf{R}\,\mathbf{x} > 0$ and $\mathbf{R}$ is positive definite. $\mathbf{R}$ is thus invertible and efficient inversion algorithms exist (*e.g.* using LDL^T Cholesky factorization [7]).

References

1. DGtal: Digital geometry tools and algorithms library, http://dgtal.org

2. Braquelaire, J., Vialard, A.: Euclidean paths: A new representation of boundary of discrete regions. Graphical Models and Image Processing 61(1), 16–43 (1999), http://dx.doi.org/10.1006/gmip.1999.0488
3. Coeurjolly, D., Dupont, F., Jospin, L., Sivignon, I.: Optimization schemes for the reversible discrete volume polyhedrization using Marching Cubes simplification. In: International Conference on Discrete Geometry for computer Imagery. pp. 413–424. LNCS, Springer Verlag (Oct 2006), http://liris.cnrs.fr/publis/?id=2431
4. Coeurjolly, D., Foare, M., Gueth, P., Lachaud, J.: Piecewise smooth reconstruction of normal vector field on digital data. Comput. Graph. Forum 35(7), 157–167 (2016), http://dx.doi.org/10.1111/cgf.13013
5. Coeurjolly, D., Lachaud, J., Levallois, J.: Multigrid convergent principal curvature estimators in digital geometry. Computer Vision and Image Understanding 129, 27–41 (2014), http://dx.doi.org/10.1016/j.cviu.2014.04.013
6. Grady, L.J., Polimeni, J.: Discrete calculus: Applied analysis on graphs for computational science. Springer (2010)
7. Guennebaud, G., Jacob, B., et al.: Eigen v3. http://eigen.tuxfamily.org (2010)
8. He, L., Schaefer, S.: Mesh denoising via l_0 minimization. ACM Transactions on Graphics (TOG) 32(4), 64 (2013)
9. Jacques-Olivier Lachaud, D.C., Levallois, J.: Modern Approaches to Discrete Curvature, chap. Robust and Convergent Curvature and Normal Estimators with Digital Integral Invariants. No. 2184 in LNM, Springer International Publishing (2017), to appear
10. Klette, R., Rosenfeld, A.: Digital Geometry: Geometric Methods for Digital Picture Analysis. Series in Computer Graphics and Geometric Modelin, Morgan Kaufmann (2004)
11. Lachaud, J., Thibert, B.: Properties of gauss digitized shapes and digital surface integration. Journal of Mathematical Imaging and Vision 54(2), 162–180 (2016), http://dx.doi.org/10.1007/s10851-015-0595-7
12. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. In: ACM siggraph computer graphics. vol. 21, pp. 163–169. ACM (1987)
13. Ohtake, Y., Belyaev, A.G.: Dual/primal mesh optimization for polygonized implicit surfaces. In: Proceedings of the seventh ACM symposium on Solid modeling and applications. pp. 171–178. ACM (2002)
14. Schaefer, S., Warren, J.: Dual marching cubes: Primal contouring of dual grids. In: Computer Graphics and Applications, 2004. PG 2004. Proceedings. 12th Pacific Conference on. pp. 70–76. IEEE (2004)
15. Wang, R., Yang, Z., Liu, L., Deng, J., Chen, F.: Decoupling noise and features via weighted l_1-analysis compressed sensing. ACM Transactions on Graphics (TOG) 33(2), 18 (2014)
16. Wu, X., Zheng, J., Cai, Y., Fu, C.W.: Mesh denoising using extended rof model with l1 fidelity. Computer Graphics Forum 34(7), 35–45 (2015)
17. Zhang, H., Wu, C., Zhang, J., Deng, J.: Variational mesh denoising using total variation and piecewise constant function space. IEEE transactions on visualization and computer graphics 21(7), 873–886 (2015)