Pacific Graphics 2016
E. Grinspun, B. Bickel, and Y. Dobashi
(Guest Editors)

Volume 35 (2016), Number 7

Piecewise smooth reconstruction of normal vector field on digital data

David Coeurjolly[1] Marion Foare[2] Pierre Gueth[1] 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

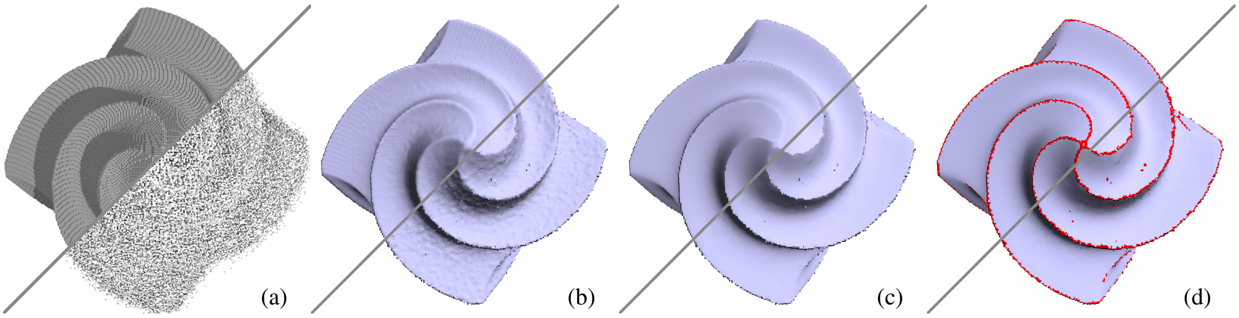


Figure 1: *Piecewise smooth reconstruction of a normal vector field on a digital shape, normal vectors are represented through the flat shading of faces according to illumination (top-left: perfect digitization / down-right: noisy digitization): (a) input digital shape V, (b) input normal vector field* **g** *obtained with digital integral invariant (II) method [CLL14] with $r = 3$, (c) output normal vector field* **u** *and (d) sharp features* **v** *superposed in red. Perfect and noisy digitization results are obtained using the same parameters. Our approach smoothes the input normal vector field except across sharp features, which are precisely delineated, thin, and consistent with the smoothing.*

Abstract

We propose a novel method to regularize a normal vector field defined on a digital surface (boundary of a set of voxels). When the digital surface is a digitization of a piecewise smooth manifold, our method localizes sharp features (edges) while regularizing the input normal vector field at the same time. It relies on the optimisation of a variant of the Ambrosio-Tortorelli functional, originally defined for denoising and contour extraction in image processing [AT90]. We reformulate this functional to digital surface processing thanks to discrete calculus operators. Experiments show that the output normal field is very robust to digitization artifacts or noise, and also fairly independent of the sampling resolution. The method allows the user to choose independently the amount of smoothing and the length of the set of discontinuities. Sharp and vanishing features are correctly delineated even on extremely damaged data. Finally, our method can be used to enhance considerably the output of state-of-the-art normal field estimators like Voronoi Covariance Measure [MOG11] or Randomized Hough Transform [BM12].

Categories and Subject Descriptors (according to ACM CCS): [Computer Graphics]: Modeling - Digital Geometry Processing—Modeling - Shape Analysis

1. Introduction

Processing discrete 3D data to enhance their geometric quality is an important task in many shape modeling and computer graphics applications. For instance, it influences deeply object rendering, shape matching, shape compression, salient part extraction or the stability of numerical simulation performed on object boundary. We focus here on processing digital surfaces, which are boundaries of sets of voxels (i.e. points of $\mathbb{Z}^3$). These data naturally come from the segmentation or binarization of MR, X-ray tomographic or micro-tomographic images. Most of the time, the original shape of interest has a piecewise smooth boundary whose digitization introduces arithmetic artefacts and noise around its boundaries. A typical application is illustrated in Figure 2. Starting from a volumetric tomographic images of a snow sample microstructure (Figure 2-(a)), a binarization process extracts the ice/air interface. Thermo-mechanical properties of the snow sample at this microscale level are related to geometrical properties of the ice/air interface [CLB*00]. Reconstructing smooth ice crystals facets while preserving sharp edges at crystal facets boundaries (Figure 2-(b)) is a crucial step in snow metamorphism modeling [FB08]. Preserving the digital nature of the input surface is important here to alleviate potential approximations when converting it to a triangular mesh.

In this paper, we propose a variational approach to recover such underlying piecewise smooth object. More precisely, we regularize a raw input normal vector field on the digital surface while preserving sharp features defined as loci of high discontinuities in the vector field. Although many works have been proposed for estimating the normal vector field on point clouds or digital shapes [CP05, MOG11, BM12, CLL14], their theoretical guarantees require smoothness on the original shape. These methods are often unreliable around sharp features of the surface. Statistical and voting strategies can then be used to control the anisotropic regularization of the estimation [BM12, ZCL*13]. Note that getting a consistent normal vector field considerably facilitates further post-processing, notably surface reconstruction (as stated in [BTS*14]).

In contrast with former approaches, our method reconstructs a consistent and smooth normal field, and delineates discontinuities of this field at the same time. We use a variational formulation that incorporates the fit to an input normal data, its piecewise regularization, and a penalty on the regularity and length of discontinuities. When input data is an image, this functional is known as Ambrosio-Tortorelli (AT) functional [AT90]. We adapt it to the piecewise smooth reconstruction of a normal vector field. To avoid known difficulties related to the discretization and optimization of AT functional, we reformulate it in a discrete calculus setting. The model is thus able to locate precisely discontinuities as 1-dimensional curves on the digital surface, and to smooth at the same time the input normal field in directions without discontinuity. Experiments show that sharp and vanishing features are correctly delineated even on extremely damaged data.

1.1. Related works

Estimating normals on point clouds. There has been many works published on this topic in the last twenty years. We mention here only recent methods that either present theoretical robustness guarantees or incorporate sharp features in their formulation.

By formalizing and extending former works using the Voronoi diagram for normal estimation (e.g. [ACSTD07]), Mérigot *et. al.* estimate the geometry of point clouds with a covariance matrix analysis of the distance function [MOG11], the so-called Voronoi Covariance Measure (VCM). The VCM matrix is proved robust to Hausdorff perturbation. Another interest of this method is that edges/sharp features in data are recognizable in eigenvalues. Cuel *et. al.* have generalized the VCM to make it resilient to outliers [CLMT15]. However, when data is corrupted, the smoothing parameter affects both the normal estimation and the feature detection, which is either unstable or produces wide and fuzzy features.

Boulch and Marlet follow a statistical approach to estimate normals on point clouds [BM12], called Randomized Hough Transform (RHT). The idea is to consider many random triples of input points and to bin their normal direction into a spherical histogram. Maximal vote gives the local normal. The method offers some (statistical) guarantees and the output normal field is not smoothed around edges. However, the normal field is not perfectly smooth along smooth parts of the shape. Zhang *et. al.* adopt a low-rank representation to segment locally the feature points into subspace [ZCL*13]. This approach handles better point clouds with variable sampling around features, at the price of a very high computation cost and parameter tuning. Liu *et. al.* improve the computation time of the preceding method, essentially by restricting the set of points where heavy computations are done [LZC*15]. Both methods use thresholds to discriminate between feature and common points.

Normal estimation on digital surfaces. Digital data are specific point clouds whose coordinates are integers (subsets of $\mathbb{Z}^3$) associated to a combinatorial structure. As said in the introduction, such data often come from the binarization or segmentation of 3D images (e.g. thresholding, Bayesian classification, watershed). Formally, given a compact shape $\mathcal{X} \subset \mathbb{R}^3$, its digitization at gridstep h is $\{\mathbf{z} \in \mathbb{Z}^3, h\mathbf{z} \in \mathcal{X}\}$. Subsets of $\mathbb{Z}^3$ can be seen as a collection of cubes, so their topological boundary forms a quadrangulated mesh, called digital surface, whose vertices have (half-)integer coordinates. They are thus approximations of continuous 2-manifolds with a specific isothetic noise model: samples are evenly spaced, quad normals are not informative, even perfect digitizations present an Hausdorff noise of at least h, due to arithmetic approximations. Hence local approaches to normal estimation cannot be convergent toward exact normals when increasing the resolution. Digital variants of Integral Invariants (II) [CLL14] or VCM [CLT14] have convergence guarantees for digitization of smooth shapes, but the radii must be adapted to the sampling grid step. Although robust to noise, these methods do not handle well sharp edges.

Sharp feature extraction. Many works aim at identifying specific geometric zones on shapes, like dominant points, sharp edges, salient parts, gathered under the generic name "feature". Here we are interested in sharp features defined on the original continuous shape boundary as discontinuities in the normal vector field. We present below a representative set of feature detection techniques which serve as benchmarks in Section 3.

For meshes or point clouds, many approaches are based on integral quantities computed in a local neighborhood. For instance, Pauly *et. al.* [PKG03] and Clarenz *et. al.* [CRT04] use Principal Component Analysis on nearby data points. Eigenvalues analysis, sometimes as a function of the neighborhood size, provides a feature score. Mérigot *et. al.* [MOG11] use also a ratio of eigenvalues of the VCM to identify large angle variations. The neighborhood size is used in these methods to regularize possible perturbations on data. Unfortunately it tends also to generate large and diffused features. Park *et. al.* [PLL12] have proposed a Tensor-Voting strategy on local surface patches. Edges are identified with a scale-space analysis of the tensor vote. As shown in [LCL15], this technique does not provide sufficiently robust results on digital surfaces. Mellado *et. al.* [MGB*12] have introduced a fast least square spherical fitting approach to a point cloud to create a multi-scale feature score. Although qualitatively relevant, experiments show that it fails to provide a precise localization of sharp features. Some authors propose a statistical approach to extract feature lines in point clouds and we mention the very recent work of Zhang *et.*

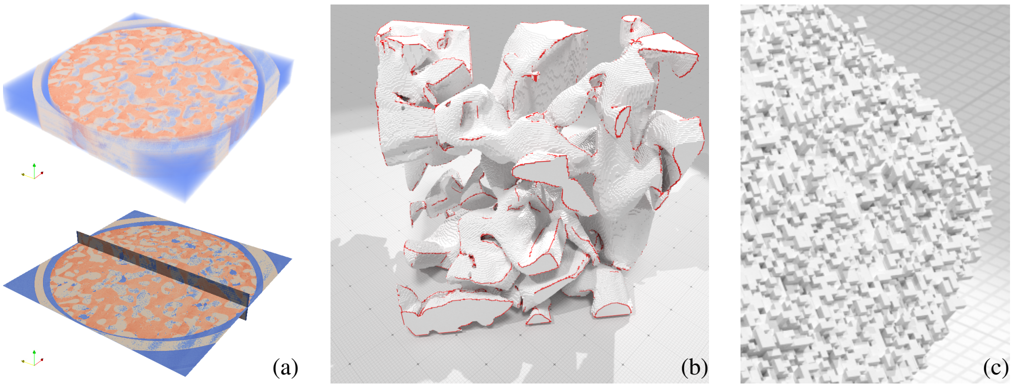


Figure 2: *Analysis of 3D snow micro-structures in Material Sciences: from a volumetric tomographic images of a snow sample (a), we process the ice/air interface geometry (AT feature extraction to determine snow facets in (b)) thermo-mechanical properties of the material (LIRIS - 3SR - MétéoFrance - CEN/CNRM GAME). (c) A close-up showing the noise model of the OctaFlower shape of Fig.1.*

al. [ZGW*16]. Authors identify potential features as points which vary too much from growing faces. Consistency between feature points, lines, and faces is then used to eliminate false positives, and provides a reconstruction of feature lines and not solely a score. However, regularization of point positions and normals, feature detection, and topological reconstruction are performed as a pipeline involving several parameters. Finally, features can be extracted following a spectral analysis of the shape from eigenvalues of the surface Laplacian matrix [GBAL09, SOG09, SLMR14]. In this context, features are characterized by spectral quantities which are locally stable and distinguishable from its neighborhood. However, these features or saliency maps are not related to sharp edges, but more to a behaviour with respect to vibrations.

Very few works exist for extracting sharp features on digital surfaces. We mention the recent approach of Levallois *et. al.* [LCL15], which detects features in scale-space, using multigrid convergence properties of curvature estimation with digital integral invariants. It will be compared to our method in section 3.

Variational models for piecewise smooth reconstruction. The Mumford-Shah (MS) model is a classical variational formulation of image restoration and segmentation [MS89]. It allows the piecewise smooth reconstruction of a possibly perturbated function, while controlling the amount of discontinuities. Notably difficult to solve directly, this formulation has inspired many works in image analysis as well as more tractable variants. Most of them transform the original formulation so that discontinuities are the boundary of regions, and the result is a partition. Since we would like to process shapes with fading sharp features, such relaxations are not suitable. Our approach follows the Ambrosio-Tortorelli (AT) relaxation of MS [AT90], which can handle dangling or fading discontinuities. However, AT is not so easy to solve numerically, and we discuss this issue with more details in Section 2.4.

We also mention also the convex relaxation of MS functional proposed by Alberti *et. al.* [ABDM03], by means of a calibration (some well-chosen vector field). It leads to algorithms targeting global optimum of the MS for scalar functions [PCBC09] and for vectorial functions [SCC12]. However, global optimality is not always guaranteed and these approaches are computationally very demanding (more than 600 seconds for a 128×128 images with 32 greylevels with a fast GPU implementation reported in [PCBC09]). They are not usable on meshes with hundred of thousands faces.

Variational models for mesh denoising. Note that other image processing techniques have proven to be useful in the context of mesh processing. Apart from the ubiquitous spectral approaches, we can mention methods that regularizes meshes while trying to preserve features. Fleishman *et. al.* [FDCO03] propose to use bilateral filtering for this purpose. This approach is not very well adapted to digital surfaces, which are too perturbated at a local scale for this local processing. Along the same lines, Jones *et. al.* [JDD03] smooth a mesh by projecting vertices onto predicted local tangent planes. Both approaches are interesting on meshes whose perturbation is small, about $1/5$ of the average edge length, but would fail on our data. More recently, several authors have considered l_0 (sparsity) or l_1 minimization (total variation) schemes to denoise a mesh while preserving its features [HS13, WYL*14, WZCF15, ZWZD15]. Compressed sensing optimization is used to detect features in [WYL*14]. All these recent approaches denoise triangulated meshes and use vertex positions as input. They achieve impressive results especially on piecewise flat shapes, when the topology of the initial mesh is correct. Although our formulation is specific to normal vector field reconstruction on digital surfaces and does not use vertex positions, reformulating our AT model for mesh processing allowing comparisons with these literature is a challenging future works.

Other works with AT functional. Chambolle and Dal Maso [CDM99, BC00] and Bourdin and Chambolle [BC00] have proposed to use finite elements to solve AT functional for restoring 2D images, with a refinement procedure valid only for planar domains. Shah uses AT functional to determine 2D and 3D shape skeletons [Sha05]. The skeleton is determined by the discontinuity set of the gradient of the squared distance to the boundary of the shape. Hence AT functional can recover smooth skeletons of controled length, even for noisy shape. Shah emphasizes the fact that the skeleton may be thick. In a more recent work, Rosman *et. al.* [RBB*11] have proposed a motion-based segmentation process of a 3D articulated shape into rigid parts. Assuming that motion transformation should vary smoothly into rigid parts and vary suddenly into non-rigid parts, they regularize a map between shapes describing different motions with AT functional to recover a motion estimation. The numerical scheme used in these two works cannot detect thin discontinuities and the resulting reconstructions are smooth everywhere, in opposition to our formulation. Finally, Kee and Kim [KK14] have studied several convex relaxations of AT functional, by using the factorability of some non-convex problems. Although interesting, their best relaxation presents some blurring around edges on given images, as well as some false contours. We discuss more about it in the supplementary material.

1.2. Contributions

Our method takes into account the fact that, on a piecewise smooth shape, sharp features lie exactly where normal vectors are not continuous. The AT functional defines two functions, a function v describing the set of discontinuities, a function u that is a smoothed version of the raw input normal field except at places where v indicates a discontinuity. Thanks to discrete calculus formulation, we propose a global optimization approach that minimizes AT functional on digital surfaces. We obtain a robust regularization of the normal vector field and a precise localization of sharp features as a set of edges. As a side product, our method can enhance the output of state-of-the-art anisotropic normal estimators.

We describe our model in Section 2 as well as its numerical implementation. Section 3 presents an experimental evaluation of our method as well as many comparisons with existing techniques. Section 4 concludes with a discussion on the pros and cons of the method as well as possible future directions of research.

2. Variational reconstruction of a piecewise-smooth normal vector field

2.1. Ambrosio-Tortorelli approximation of MS functional

Let Ω be a 2-dimensional compact domain (for instance the plane $\mathbb{R}^2$ or a surface S). Let $g : \Omega \to \mathbb{R}^d$ be the function we wish to approach. The continuous formulation of the Ambrosio-Tortorelli functional can be written as:

$$\mathrm{AT}_\varepsilon(u,v) := \int_\Omega \alpha|u-g|^2 + v^2|\nabla u|^2 + \lambda\varepsilon|\nabla v|^2 + \frac{\lambda}{\varepsilon}\frac{(1-v)^2}{4}d\mathbf{x}, \tag{1}$$

where $u, v \in SBV(\Omega)$ (continuous functions in Ω with bounded variations). Function u is designed to be a piecewise smooth approximation of g, while function v is meant to approach discontinuities. More precisely, the first term expresses the fit to the input function g, the second term tells that u must be smooth except at places where v is zero. The third term measures the smoothness of function v while the last term forces v to stay close to 1. The second term is critical since it decides between a smooth part of the reconstruction ($v \approx 1$) and a discontinuity ($v \approx 0$).

Ambrosio and Tortorelli have proved that functional AT_ε Γ-converges toward the Mumford-Shah functional as ε tends toward 0 [AT90]. This convergence means that discontinuities are approached by stiffening progressively function v around them [Bra98]. At first, a large ε makes the function v be a diffuse approximation of discontinuities. Then, parameter ε is decreased so that the strip where v is close to 0 is made more and more narrow. Ultimately, in the continuous case, the set $\{v = 0\}$ is exactly the set of discontinuities. Using dimensional analysis, ε is a `distance` defining the width of diffusion of discontinuities. $1/\alpha$ is the `area` over which input data is smoothed. The smaller α is, the stronger the smoothing is. Finally, $1/\lambda$ is a `distance`, proportional to the length of discontinuities we allow in the piecewise smooth reconstruction. The smaller λ is, the longer discontinuities are.

At first glance, this functional perfectly fits our purpose: if we define the domain Ω to be our input surface and if we set g as the raw normal vector field, considered as a normal direction field, then minimizing AT_ε with $\varepsilon \to 0$ provides (1) the piecewise smooth approximation u of this input field and (2) the set of discontinuities, *i.e.* sharp features, as $\{v \approx 0\}$. Unfortunately, it is not easy to optimize this energy in the discrete world. We discuss this problem in details in Section 2.4. Just note for now that the optimal v has almost everywhere value 1, and has value 0 only on a set of null 2-Hausdorff measure: singularities are thin. Therefore it is difficult to capture v with a sampling, and most numerical schemes fail.

2.2. Digital formulation of AT functional

Our input data is a set of voxels V (i.e. a subset of $\mathbb{Z}^3$) and the domain Ω will be the boundary of V, seen as a collection of closed unit cubes. Then Ω is decomposed as a 2-dimensional closed cubical complex K, composed of unit cube faces, edges and vertices. Note that we add a notion of adjacency between face elements to get a digital surface [Her92]. From this primal cubical complex, a dual complex is constructed by considering dual vertices at the center of each primal square face, dual edges orthogonal to primal edges joining adjacent primal square face centers and dual faces as umbrellas centered around primal vertices with borders composed of dual edges (see Fig. 3).

Eq. (1) is discretized by considering discrete k-forms as scalars associated with k-cells of K. This approach follows the discrete calculus approach of [GP10] and shares also several characteristics of the discrete exterior calculus scheme [DHLM05], modified to handle cubical cells instead of simplexes. It induces the discretization of linear operators between k-forms as matrices.

The input field g of raw normal vectors is defined by its three dual 0-forms g_1, g_2, g_3 associating to each primal face the three components of the normal vector. Similarly, estimated normal u is defined as three dual 0-forms u_1, u_2, u_3 associating to each primal face the components of the regularized normal vector. Finally v is a primal 0-form associating the feature selection scalar to each primal vertex. This definition allows us to make ε as small as we want (typically $\frac{1}{4}$ of the distance between two voxels), and v become identically 1 everywhere except at some vertices lying on discontinuities where it will be able to retain value 0.

Let $\mathbf{d}_k$ (resp. $\mathbf{d}_{\bar{k}}$) be the primal (resp. dual) discrete exterior derivative operator from primal (resp. dual) k-forms to primal (resp. dual) $k+1$-forms. Discrete inner product between primal (resp. dual) k-forms is denoted by $\langle \cdot,\cdot \rangle_k$ (resp. $\langle \cdot,\cdot \rangle_{\bar{k}}$). The wedge product between a primal 0-form γ and a dual 1-form β is a dual 1-form whose discretization can be computed as

$$\gamma \wedge \beta = \operatorname{diag}(\mathbf{M}\gamma)\,\beta = \operatorname{diag}(\beta)\,\mathbf{M}\gamma,$$

where $\mathrm{diag}(\cdot)$ denotes the matrix whose diagonal elements are the components of the given vector and the operator $\mathbf{M}$ transports a primal 0-form to a dual 1-form, and, for each edge, simply averages the values at the extremities of this edge *without taking into account orientation*. In our setup, it is defined as $\mathbf{M} := \frac{1}{2}|\mathbf{d}_0|$. With u_i as dual 0-forms and v as a primal 0-form, we formulate a digital version of AT_ε as follows:

$$\begin{aligned}\mathrm{AT}^d_\varepsilon(u,v) :=& \alpha \sum_{i=1}^{3} \langle u_i - g_i, u_i - g_i \rangle_{\bar{0}} + \sum_{i=1}^{3} \langle v \wedge \mathbf{d}_{\bar{0}} u_i, v \wedge \mathbf{d}_{\bar{0}} u_i \rangle_{\bar{1}} \\ &+ \lambda\varepsilon \langle \mathbf{d}_0 v, \mathbf{d}_0 v \rangle_1 + \frac{\lambda}{4\varepsilon} \langle 1 - v, 1 - v \rangle_0. \qquad (2)\end{aligned}$$

A possibility would be to inject metrics into the discrete calculus, by means of inner products (as in [GP10]). This is difficult in our case since the input data V may be very noisy. We therefore choose identity metrics for inner products. Denoting $\mathbf{A}$ the matrix form of $\mathbf{d}_0$ (the incidence matrix sending vertices toward edges) and $\mathbf{B}$ the matrix form of $\mathbf{d}_{\bar{0}}$ (the incidence matrix sending faces toward edges), we rewrite (2) in matrix/vector form:

$$\begin{aligned}\mathrm{AT}^d_\varepsilon(\mathbf{u},\mathbf{v}) =& \alpha \sum_{i=1}^{3} (\mathbf{u}_i - \mathbf{g})^\mathsf{T} (\mathbf{u}_i - \mathbf{g}) + \sum_{i=1}^{3} \mathbf{u}_i{}^\mathsf{T} \mathbf{B}^\mathsf{T} \mathrm{diag}(\mathbf{M}\mathbf{v})^2 \mathbf{B}\mathbf{u}_i \\ &+ \lambda\varepsilon \mathbf{v}^\mathsf{T} \mathbf{A}^\mathsf{T} \mathbf{A}\mathbf{v} + \frac{\lambda}{4\varepsilon} (\mathbf{1} - \mathbf{v})^\mathsf{T} (\mathbf{1} - \mathbf{v}) \qquad (3) \\ =& \alpha \sum_{i=1}^{3} (\mathbf{u}_i - \mathbf{g})^\mathsf{T} (\mathbf{u}_i - \mathbf{g}) + \sum_{i=1}^{3} \mathbf{v}^\mathsf{T} \mathbf{M}^\mathsf{T} \mathrm{diag}(\mathbf{B}\mathbf{u}_i)^2 \mathbf{M}\mathbf{v} \\ &+ \lambda\varepsilon \mathbf{v}^\mathsf{T} \mathbf{A}^\mathsf{T} \mathbf{A}\mathbf{v} + \frac{\lambda}{4\varepsilon} (\mathbf{1} - \mathbf{v})^\mathsf{T} (\mathbf{1} - \mathbf{v}), \qquad (4)\end{aligned}$$

where column vectors $\mathbf{u}_i$ and $\mathbf{v}$ contain the scalars defining the discrete k-forms u_i and v, given a numbering of the k-cells. All the matrix/vector computations in (3) and (4) correspond to very simple and local operations on the quad mesh, as illustrated on Fig. 3.

2.3. Optimization of digital variant of AT

Energy $\mathrm{AT}^d_\varepsilon$ should have null derivatives with respect to $\mathbf{u}$ and $\mathbf{v}$ at optimum. Consequently we alternately solve the following systems until equilibrium:

$$\forall i \in \{1,2,3\}, \quad \left[\alpha \mathbf{Id} - \mathbf{B}^\mathsf{T} \mathrm{diag}(\mathbf{M}\mathbf{v})^2 \mathbf{B} \right] \mathbf{u}_i = \alpha \mathbf{g}_i, \qquad (5)$$

$$\left[\frac{\lambda}{4\varepsilon} \mathbf{Id} + \lambda\varepsilon \mathbf{A}^\mathsf{T} \mathbf{A} + \mathbf{M}^\mathsf{T} \left(\sum_{i=1}^{3} \mathrm{diag}(\mathbf{B}\mathbf{u}_i)^2 \right) \mathbf{M} \right] \mathbf{v} = \frac{\lambda}{4\varepsilon} \mathbf{1}. \qquad (6)$$

The three linear systems in Eq.(5) are identical and estimate $\{\mathbf{u}_i\}$ for a given $\mathbf{v}$. Eq.(6) is a linear system that resolves $\mathbf{v}$ for fixed $\{\mathbf{u}_i\}$. All involved matrices are symmetric, definite, positive, so we use a Cholesky LL^T factorization to solve the systems. Both systems involve sparse matrix whose number of non-null coefficients is proportional to the number of faces or vertices of the input mesh. They are almost equal for digital surfaces with bounded genus.

Ambrosio and Tortorelli have shown the Γ-convergence of AT_ε toward the MS functional. This means that a sequence of minimizers for decreasing ε tends toward the MS minimizer. Since our problem is convex for fixed $\mathbf{u}$ or fixed $\mathbf{v}$, we compute at each iteration a unique minimum. A result of convex analysis related to block coordinate descent (see [Ber99], Prop. 2.7.1) ensures that the minimizers at each iteration must converge to a stationnary point of $\mathrm{AT}^d_\varepsilon$. Remember that a large ε convexifies the AT energy. Our optimization process thus starts with a relatively large ε (generally 2 in our examples), and after convergence for a given ε, we decrease ε (typically by dividing it by 2) and the process is restarted from this initial guess. We provide further illustration in supplementary material with a visual feedback of this progressive optimization.

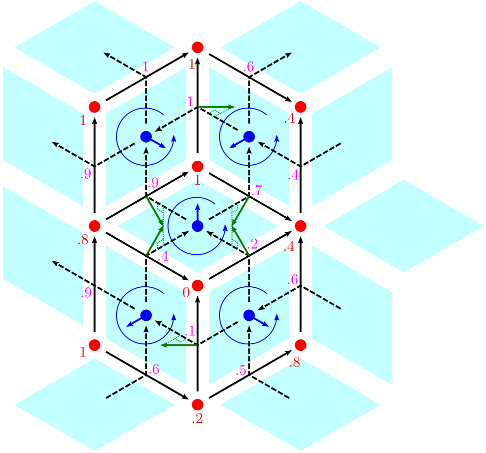


Figure 3: *Illustration of discrete k-forms* **u** *and* **v** *on a simple cubical complex. Primal faces (light blue quad) are oriented to have opposite orientation on a common primal edge (solid black arrow). Primal face and edge orientations induce oriention of dual edges (dotted black arrow). Dual 0-forms* $(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)$ *are attached to dual vertices (blue point), represented as primal faces normals (blue arrow). Primal 0-form* **v** *is attached to primal vertices (red point) and its discrete values are displayed in red. Dual 1-form* **Mv***, appearing in the discrete wedge product, is the average of* **v** *attached to dual edges and displayed in magenta. Dual 1-forms* $(\mathbf{B}\mathbf{u}_1, \mathbf{B}\mathbf{u}_2, \mathbf{B}\mathbf{u}_3)$ *(green arrow) are obtained by simple difference of* $(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)$ *across the two faces touching each edge. The sign is determined by the respective orientation of incident faces with respect to edge orientation.*

```
OPTIMIZE-AT( $(\mathbf{g}_1,\mathbf{g}_2,\mathbf{g}_3)$: $\bar{0}$-forms, $(\alpha,\lambda,\varepsilon_1,\varepsilon_2,\varepsilon_r)$: reals );
Var $(\mathbf{u}_1,\mathbf{u}_2,\mathbf{u}_3)$: $\bar{0}$-forms, $(\mathbf{v},\mathbf{v}')$: 0-forms, $\varepsilon$: real ;
begin
    foreach $i \in \{1,2,3\}$ do $\mathbf{u}_i \leftarrow \mathbf{g}_i$ $\mathbf{v} \leftarrow \mathbf{1}$, $\varepsilon \leftarrow \varepsilon_1$;
    while $\varepsilon \geq \varepsilon_2$ do
        repeat
            $\mathbf{v}' \leftarrow \mathbf{v}$;
            Solve Eq.(5) for all $\mathbf{u}_i$;
            Solve Eq.(6) for $\mathbf{v}$;
        until $\|\mathbf{v} - \mathbf{v}'\|$ is small ;
        $\varepsilon \leftarrow \varepsilon / \varepsilon_r$ ;
    end
    Normalize $(\mathbf{u}_1,\mathbf{u}_2,\mathbf{u}_3)$;
    return $((\mathbf{u}_1,\mathbf{u}_2,\mathbf{u}_3),\mathbf{v})$ ;
end
```

The inner optimization loop (line 1) stops when the 0-form $\mathbf{v}$ becomes stable. In practice, a few iteration is always sufficient. We have used maximum 5 iterations in all our examples.

The input dual 0-forms $(\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3)$ corresponds to the input raw estimation of the normal vector field on the digital surface. In Section 3 and supplementary material, we discuss about the influence of the normal vector estimator on the AT minimization. When the minimization stops, the dual 0-forms $\{\mathbf{u}_i\}$ define the regularized normal vector field and the 0-form $\mathbf{v}$ is a scalar value in $[0,1]$ which is close to zero on features. The final feature extraction evaluated in Section 3 corresponds to the collection of 1-cells such that the values of $\mathbf{v}$ at their vertices are both smaller than $\frac{1}{2}$. This naive thresholding on $\mathbf{v}$ values could be improved but it already leads to very stable and precise results.

2.4. Discussion about discrete formulation

There are many alternate ways of discretizing functional (1): some of them should definitely be avoided, while others are acceptable but remain behind our formulation. We sum up in this section our experience on several alternative discretizations.

General comments. The continuous formulation of Ambrosio-Tortorelli involves two functions u and v defined over some domain. When ε tends to 0, the two last terms tends toward the perimeter of the set of discontinuities. The main problem when discretizing this functional comes from the fact that v tends toward a function that cannot be approached by a standard sampling. More precisely v tends toward a function that is almost everywhere 1, meaning except on sets with null Hausdorff-2 measure.

Discrete calculus versus other numerical schemes. A discretization with standard finite differences fails for this functional. Indeed, as noted for instance by Bourdin and Chambolle [BC00], if h is the sampling step, then both ε, h and h/ε must tend toward zero to get a correct approximation of AT_ε, otherwise results are poor. In practice this means that ε should be between 5-10 times greater than h, leading to "features" more than 10 voxels wide. Therefore, several people have proposed to use a finite-element approach to solve this problem [CDM99, BC00] in the case where Ω is a planar square (e.g. an image domain). The mesh should then be adapted progressively to the set of discontinuities, by refining triangles so that edges are tangent to discontinuities. This is required to get a correct approximation of the gradient of u close to discontinuities. Such approaches are complex to implement. We cannot follow this strategy for two reasons. First it would be extremely costly on our input meshes, which have often 1 million faces. Second, our input meshes may be strongly perturbated and do not constitute an accurate tiling of a smooth domain. It would be very difficult to refine such mesh along discontinuities. In supplementary material, we provide a comparison between our AT formulation and classical finite difference or finite element methods on flat domains.

Possible variants in discrete calculus. From (1), it is natural to consider u as dual 0-forms and v as a 0-form. Different formulations could have been considered, for instance with both u and v being 0-forms. However, the crucial point in the formulation is the discretization of the term $v^2|\nabla u|^2$ since it relates the two functions u and v. We observe that this term must be seen as a 1-form living on edges. It is consistent with the fact that the antiderivative of u lives on edges. Form v must thus either be defined on 1-cells or be transported to it. We have chosen u to be a dual 0-form, since the normal field is associated to faces of digital meshes, and v is a 0-form so that it lives in between normals. Form v is transported by simple averaging onto edges as a 1-form.

Γ-convergence towards a unitary normal vector field. In the vectorial case, Focardi *et. al.* proved that the Ambrosio-Tortorelli functional Γ-converges toward a kind of Mumford-Shah functional, where the length term is equal to one to three times the perimeter [FI14]. Optimizing a unitary tangent vector field may be difficult [KCPS13], we prefer a simpler component-wise approach. In smooth parts, each component converges independently and since the input field g is unitary, the overall vector is close to be unitary almost everywhere.

3. Experiments

One of the main originality of AT functional is its ability to regularize the normal vector field and to localize features at the same time. For comparisons to state of the art methods, we evaluate them independently in the following sections. Our approach is parametrized by an input raw normal vector field and α, λ and ε values. In the following experiments, we discuss the impact of these parameters on the regularized field and on the feature extraction. Extended analysis is provided in supplementary material. For some experiments, noisy digital objects are considered. The noise model adapted to digital data consists in flipping the grid point value at p with probability defined by a power law $k^{1+dt(p)}$ for some user-specified $k \in [0,1]$ ($dt(p)$ being the distance of p to the boundary voxels of the original digital shape).

3.1. Feature extraction

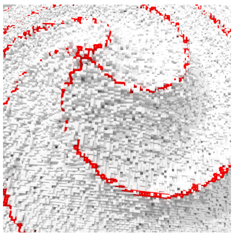
In Figure 4, we compare the feature extraction ability of AT functional with methods working at a single scale like VCM [MOG11] or with methods using a scale-space (Pauly *et al* [PKG03], Mellado *et al* [MGB*12] and Levallois *et al* [LCL15]). Except for the latter one which is specific to digital surfaces, all methods have been originally designed for point cloud processing. The digital surface is thus seen as a point cloud defined by surfel centers. Such point cloud induced by the digitization process can be seen as an Hausdorff sampling of the underlying continuous object, and is consistent with sampling hypotheses usually considered in point cloud processing (see [MOG11] for example).

For the VCM approach [MOG11], each point p of the surface is associated with a *Voronoi Covariance Measure*, which depends on two parameters R and r: the offset radius R dilates the input set while the convolution radius r defines the Voronoi cells that are integrated to smooth the measure. Both parameters allow to control the impact of noise in the input data while preserving geometrical information. To extract a feature score, authors compute a ratio of the eigenvalues of the convolved VCM at each point p: $r(p) := \frac{\lambda_1}{\lambda_0+\lambda_1+\lambda_2}$. The point is considered as a sharp edge when the ratio is greater than a threshold parameter T. Value of $r(p)$ greater than T is thus the feature score of p.

Pauly *et al.* [PKG03] use the eigenvalues of the covariance matrix of the point cloud at each point on the shape surface in a given neighborhood. More precisely, they consider the values of $\tau_i(p) := \frac{\lambda_0}{\lambda_0+\lambda_1+\lambda_2}$ for a range of radii $\{R_i\}_i = 0..n$ ($\lambda_0 \leq \lambda_1 \leq \lambda_2$ being the eigenvalues of the covariance matrix). Since these eigenvalues decrease as the curvature increases, τ_i will be higher on edges than on flat parts of the surface. To enhance this classification, the weight $\omega(p)$ is defined as the number of times τ_i is greater than τ_{max} on the range of radii ($\omega(p) := Card\{\tau_i \geq \tau_{max} \,|\, 0 \leq i < n\}$). This quantity is then used as a feature score.

Mellado *et al* [MGB*12] consider a least squares spherical fitting approach at surface points. The scale-space parameter is the neighborhood size considered in the fitting. Then, following their notations, for each scale t, they fit an algebraic hyper-sphere and get the algebraic offset distance τ between p and the 0-isosurface, the unit normal η and the signed curvature κ of the hyper-sphere. Then, they compute a *geometrical variation* $\nu(p,t)$ at a point p defined from fitted sphere parameters. For a range of scale $t \in [r_{min}, r_{max}]$, the authors define a continuous feature score function $f(p) := \int tanh(\nu(p,t))dt$ that differentiates regions with no geometrical variations from those with high variations.

Finally, Levallois *et al* [LCL15] is based on a scale-space analysis of Integral Invariant based curvature estimators [PWY*07, CLL14]. More precisely, these estimators are defined from the volume, or covariance matrix eigenvalues, of the intersection between a ball of radius R and the object surface. For a given range of radii $[r_{min}, r_{max}]$, an analysis is performed on the estimated values in order to classify each surfel of the digital surface into three classes, Edge part, Smooth part and Flat part (or zero mean curvature part).



In Figures 4, we have considered the same set of parameters for all shapes (with or without noise). Even if some existing approaches provide robust feature selection (VCM or [LCL15]), the global minimization of AT functional allows us to have more precise and thin delineation of sharp features. Furthermore, our features are outputted as a collection of 1-cells (quad edges), compared to a feature score that needs to be post-processed to localize a one-dimensional feature (see zoom on the noisy OctaFlower shape). From the set of parameters chosen for our method in these experiments, we can observe that the smooth edge vanishing on the front part of fandisk object has not been reconstructed by AT compared to [LCL15] or [PKG03]. Changing the parameters would allow us to capture such feature. In supplementary material, we provide more experiments to show the influence of the parameters (α, λ and ε) on the results.

3.2. Regularization of normal vector field

We then evaluate the quality of the regularized normal vector field. To compare normal vector fields, we use a rendering of quads associated with their normal vector and a specular material. In Figure 5, we consider several noisy versions of the Fandisk object and the estimated normal vector field using VCM, RHT, II and our approach where the input normal vector field is II with radius 4. Both VCM and RHT provide robust and moderately smooth normal vector reconstructions with respect to the noise level. However, edges are also smoothed. Since II has not be designed to be anisotropic with respect to discontinuities, large integration radii lead to smooth reconstruction but all edges disappear (see row "II, $r = 8$"). If the integration is small ($r = 4$ for instance) edges are preserved but noise becomes visible. If we feed the AT functional with this II input estimation (input functional $\mathbf{g}$), we obtain a smooth reconstruction with preserved edges. In supplementary material, we provide additional results when combining RHT or VCM as input to AT functional. We observe that AT functional always improves the input normal vector field while preserving the features.

3.3. Implementation details and timings

Many digital geometry processing tools used in this work are available in the open-source DGtal library [tDGtal]. More precisely, DGtal has optimized implementations of normal vector estimators (II, VCM, RHT) and discrete calculus operators (with Eigen [GJ*10] backend for linear algebra factorization and resolution) on digital surfaces. For the OctaFlower object of Fig. 4 (512^3), the digital surface has 671270 quads. The input normal vector field estimation (using II with $r = 8$) takes 443 seconds. Each inner loop iteration takes approximately 14 seconds (prefactorization, solve for $\{\mathbf{u}_i\}$ and solve for $\mathbf{v}$). The overall AT optimization process ($\varepsilon = [2, 0.25]$, 5 iterations on the inner loop) takes 423 seconds. On the same data and as an indicative basis, VCM ($R_1 = 10$, $r_1 = 10$, $T = 0$) takes 97 seconds. RHT ($r = 6$) For Mellado *et al.*, Pauly *et al.* and Levallois *et al.*, the feature extraction is multiscale and requires local convolutions for a range of radii ($r_{min} = 5$, $r_{max} = 20$ with step 1). Even if local convolutions differ, all processes have similar overall timings (approx. 2400 seconds). Note that these timings correspond to non optimized CPU versions of the algorithms.

4. Discussions and limitations

This paper proposes a new method for the piecewise regularization of normal vector fields onto digital surfaces. Its originality lies in its global variational formulation that addresses at the same time the regularization of normals and the delineation of sharp features as the locii of vector discontinuities. Experiments show that sharp features are correctly localized even on damaged data while keeping smooth normal vectors except across features. Our method can also be used to enhance the normal estimation of several state-of-the-art methods (see also supplementary material for a quantitative evaluation).

Limitations. As many methods developed for this purpose (regularization in presence of sharp features), our formulation requires parameters $(\alpha, \lambda, \varepsilon)$. First note that parameter ε is related to the optimization process, and we have used the same sequence $2, 1, \frac{1}{2}, \frac{1}{4}$ for ε in all experiments. The roles of paremeters α and λ are clearly differentiated: parameter α controls the strength of the smoothing and should be adjusted according to the quality of the input normal vector field, parameter λ controls the length of discontinuities. This parameter can also be used as a filtration scale-space parameter on the extracted feature, allowing scale-space or persistent homology

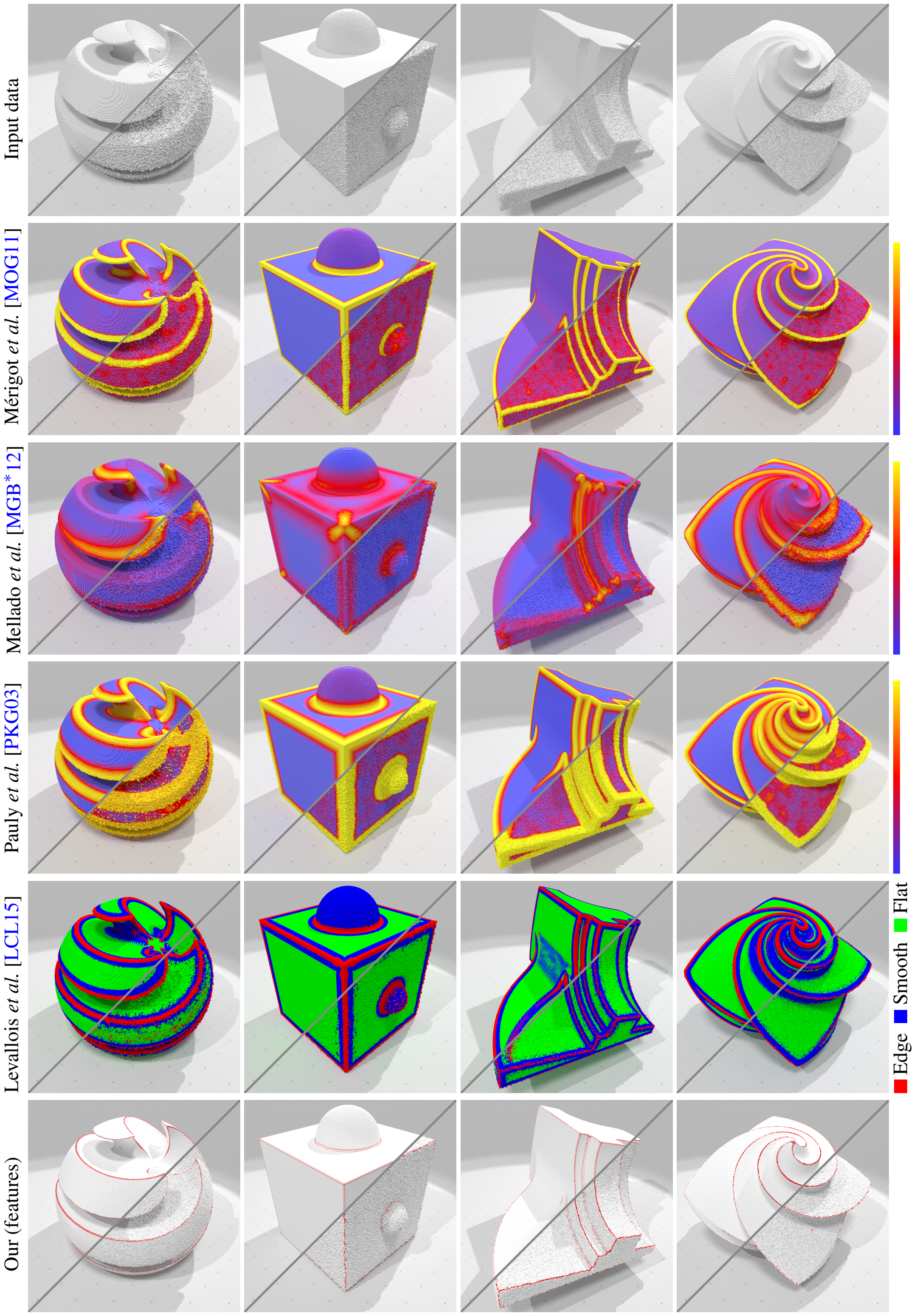


Figure 4: *Evaluation of feature detectors on perfectly digitized shapes. SharpSphere: 256^3 voxels, CubeSphere: 200^3 voxels, Fandisk: 512^3 voxels, OctaFlower: 512^3 voxels. Parameters used for [MOG11]: $R_1 = 10$, $r_1 = 10$, $T = 0$. Parameters used for [MGB*12]: $r_{min} = 5$, $r_{max} = 25$. Parameters used for [PKG03]: $r_{min} = 5$, $r_{max} = 25$, $\tau_{max} = 0.01$. Parameters used for [LCL15]: $r_{min} = 5$, $r_{max} = 20$. Parameters used for our approach: $\alpha = 0.1$, $\lambda = 0.01$, $\varepsilon = [2, 0.25]$ with integral invariant normal vector estimator with radius 8. Bottom-right parts of images correspond to the results with the same parameters and noise level $k = 0.5$.*

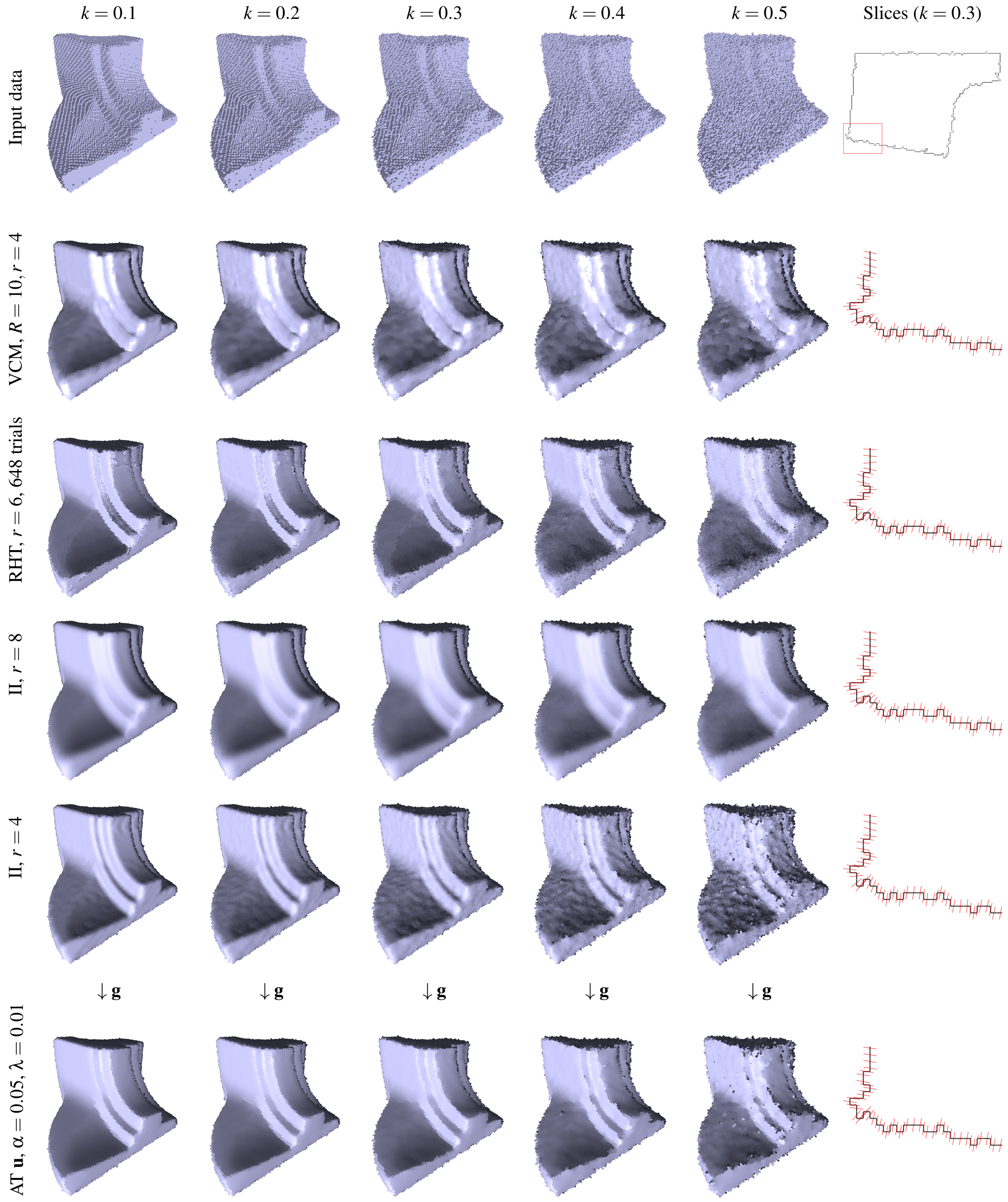


Figure 5: *Normal vector field comparison for Fandisk object* 128^3 *with increasing noise* k*. Compared normal estimators are VCM [MOG11], RHT [BM12], II with two radii [CLL14], and our model AT with input normal given by II with* $r = 4$*. A slice at the top of fandisk displays normal directions for each method.*

based analyses. Note that we have used the same α and λ in all presented experiments. The supplementary material also confirms the relative invariance of the method to large ranges of parameters.

Another limitation is that the method requires a combinatorial surface structure (digital surface), and a lot of the robustness of the method is related to this topological information. Hence the method cannot be used as is on an arbitrary cloud of points. Last, the method performs a global analysis of possible discontinuities on the surface. This analysis is thus performed at a given scale (a balance between α and λ) and not in an adaptive way. Therefore the distinction between sharp or smooth feature is related to an angle variation and it cannot directly handle well variable noise over the input normal vector field.

Finally, our discrete AT formulation is not convex, so our alternate optimization procedure, while being convergent, may stop on a local minima. Our experiments nevertheless show that the multiscale approach (decreasing sequence of ε) limits this issue. Other convex relaxations of AT model (like in [KK14]) would be interesting to cast in a discrete calculus framework. The classical convex relaxation of MS model of [ABDM03,PCBC09,SCC12] seems for now too computationnaly costly for practical geometry processing.

Future works. First it would be interesting to introduce metrics in our discrete calculus formulation in order to remove some metrication artefacts introduced by our staircase surfaces. One possibility is to inject the projected area of quads along the current estimated normal into the hodge star operator $\star$. We intend to see if some results can be enhanced with this new metric.

It is clear also that our method is rather straightforwardly extensible to triangulated meshes. DEC [DHLM05] offers exactly the tools to transpose this formulation to such meshes. This reformulation to meshes would allow us to address specific mesh processing tasks such as mesh reconstruction with optimization of the vertex position, similarly to [HS13,WYL*14,WZCF15,ZWZD15].

A more challenging task would be to transpose this work on point clouds. Approximated Laplacian operators exist on such data and should help in designing an AT functional over point clouds.

Our piecewise smooth regularization model does not require at all that $\mathbf{g}$ (and $\mathbf{u}$) be a normal vector field. In fact, the AT functional is suited to the piecewise reconstruction of any function. We intend to check its ability to regularize arbitrary functions, vector fields or tensors, in problems involving unknown discontinuities. Last, this model is fully compatible with user-interaction either for selecting smooth zones (forcing $\mathbf{v}$ to be one in these places) or for delineating manually some sharp features (forcing $\mathbf{v}$ to be zero).

Acknowlegments

This work has been partly funded by DigitalSnow ANR-11-BS02-009 research grant and CoMeDiC ANR-15-CE40-0006 research grant.

References

[ABDM03] Alberti G., Bouchitté G., Dal Maso G.: The calibration method for the mumford-shah functional and free-discontinuity problems. *Calculus of Variations and Partial Differential Equations 16*, 3 (2003), 299–333. 3, 10

[ACSTD07] Alliez P., Cohen-Steiner D., Tong Y., Desbrun M.: Voronoi-based variational reconstruction of unoriented point sets. In *Symposium on Geometry processing* (2007), vol. 7, pp. 39–48. 2

[AT90] Ambrosio L., Tortorelli V. M.: Approximation of functional depending on jumps by elliptic functional via t-convergence. *Communications on Pure and Applied Mathematics 43*, 8 (1990), 999–1036. 1, 2, 3, 4

[BC00] Bourdin B., Chambolle A.: Implementation of an adaptive finite-element approximation of the mumford-shah functional. *Numerische Mathematik 85* (2000), 609–646. 3, 6

[Ber99] Bertsekas D. P.: Nonlinear programming. 5

[BM12] Boulch A., Marlet R.: Fast and robust normal estimation for point clouds with sharp features. *Computer Graphics Forum 31*, 5 (2012), 1765–1774. 1, 2, 9

[Bra98] Braides A.: *Approximation of Free-Discontinuity Problems*. Lecture Notes in Mathematics, Vol. 1694, Springer-Verlag, pp. 89–91, 1998. 4

[BTS*14] Berger M., Tagliasacchi A., Seversky L., Alliez P., Levine J., Sharf A., Silva C.: State of the art in surface reconstruction from point clouds. In *EUROGRAPHICS star reports* (2014), vol. 1, pp. 161–185. 2

[CDM99] Chambolle A., Dal Maso G.: Discrete approximation of the mumford-shah functional in dimension two. *Mathematical Modelling and Numerical Analysis 33*, 4 (1999), 651–672. 3, 6

[CLB*00] Coléou C., Lesaffre B., Brzoska J.-B., Ludwig W., Boller E.: Three-dimensional snow images by x-ray microtomography. *Annals of Glaciology 32*, 1 (2001-01-01T00:00:00), 75–81. 2

[CLL14] Coeurjolly D., Lachaud J.-O., Levallois J.: Multigrid convergent principal curvature estimators in digital geometry. *Computer Vision and Image Understanding 129* (2014), 27 – 41. Special section: Advances in Discrete Geometry for Computer Imagery. 1, 2, 7, 9

[CLMT15] Cuel L., Lachaud J.-O., Mérigot Q., Thibert B.: Robust geometry estimation using the generalized voronoi covariance measure. *SIAM Journal on Imaging Science 8*, 2 (2015), 1293–1314. 2

[CLT14] Cuel L., Lachaud J.-O., Thibert B.: Voronoi-based geometry estimator for 3d digital surfaces. In *Proc. Int. Conf. on Discrete Geometry for Computer Imagery (DGCI'2014), Sienna, Italy* (2014), Barcucci E., Frosini A., Rinaldi S., (Eds.), vol. 8668 of *Lecture Notes in Computer Science*, Springer International Publishing, pp. 134–149. 2

[CP05] Cazals F., Pouget M.: Estimating differential quantities using polynomial fitting of osculating jets. *Computer Aided Geometric Design 22*, 2 (2005), 121–146. 2

[CRT04] Clarenz U., Rumpf M., Telea A.: Robust feature detection and local classification for surfaces based on moment analysis. *IEEE transactions on visualization and computer graphics 10*, 5 (2004), 516–24. 2

[DHLM05] Desbrun M., Hirani A. N., Leok M., Marsden J. E.: Discrete exterior calculus. *arXiv preprint math/0508341* (2005). 4, 10

[FB08] Flin F., Brzoska J.-B.: The temperature-gradient metamorphism of snow: vapour diffusion model and application to tomographic images. *Annals of Glaciology 49*, 1 (2008), 17–21. 2

[FDCO03] Fleishman S., Drori I., Cohen-Or D.: Bilateral mesh denoising. *ACM Transactions on Graphics (TOG) 22*, 3 (2003), 950–953. 3

[FI14] Focardi M., Iurlano F.: Asymptotic analysis of ambrosio–tortorelli energies in linearized elasticity. *SIAM Journal on Mathematical Analysis 46*, 4 (2014), 2936–2955. 6

[GBAL09] Gebal K., Bærentzen J. A., Aanæs H., Larsen R.: Shape analysis using the auto diffusion function. *Computer Graphics Forum 28*, 5 (2009), 1405–1413. 3

[GJ*10] Guennebaud G., Jacob B., et al.: Eigen v3. http://eigen.tuxfamily.org, 2010. 7

[GP10] Grady L. J., Polimeni J.: *Discrete calculus: Applied analysis on graphs for computational science*. Springer, 2010. 4, 5

[Her92] Herman G. T.: Discrete Multidimensional Jordan Surfaces. *Computer Vision, Graphics, and Image Processing 54*, 6 (Nov. 1992), 507–515. 4

[HS13] He L., Schaefer S.: Mesh denoising via l_0 minimization. *ACM Transactions on Graphics (TOG) 32*, 4 (2013), 64. 3, 10

[JDD03] Jones T. R., Durand F., Desbrun M.: Non-iterative, feature-preserving mesh smoothing. *ACM Transactions on Graphics (TOG) 22*, 3 (2003), 943–949. 3

[KCPS13] Knöppel F., Crane K., Pinkall U., Schröder P.: Globally optimal direction fields. *ACM Transactions on Graphics (TOG) 32*, 4 (2013), 59. 6

[KK14] Kee Y., Kim J.: A convex relaxation of the ambrosio–tortorelli elliptic functionals for the mumford-shah functional. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 4074–4081. 4, 10

[LCL15] Levallois J., Coeurjolly D., Lachaud J.-O.: Scale-space feature extraction on digital surfaces. *Computers & Graphics 51* (2015), 177 – 189. International Conference Shape Modeling International. 2, 3, 6, 7, 8

[LZC*15] Liu X., Zhang J., Cao J., Li B., Liu L.: Quality point cloud normal estimation by guided least squares representation. *Computers & Graphics 51* (2015), 106 – 116. International Conference Shape Modeling International. 2

[MGB*12] Mellado N., Guennebaud G., Barla P., Reuter P., Christophe S.: Growing least squares for the analysis of manifolds in scale-space. *Computer Graphics Forum 31*, 5 (2012), 1691–1701. 2, 6, 7, 8

[MOG11] Mérigot Q., Ovsjanikov M., Guibas L.: Voronoi-based curvature and feature estimation from point clouds. *Visualization and Computer Graphics, IEEE Transactions on 17*, 6 (2011), 743–756. 1, 2, 6, 8, 9

[MS89] Mumford D., Shah J.: Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics 42*, 5 (1989), 577–685. 3

[PCBC09] Pock T., Cremers D., Bischof H., Chambolle A.: An algorithm for minimizing the mumford-shah functional. In *2009 IEEE 12th International Conference on Computer Vision* (2009), IEEE, pp. 1133–1140. 3, 10

[PKG03] Pauly M., Keiser R., Gross M.: Multi-scale Feature Extraction on Point-Sampled Surfaces. *Computer Graphics Forum 22*, 3 (sep 2003), 281–289. 2, 6, 7, 8

[PLL12] Park M. K., Lee S. J., Lee K. H.: Multi-scale tensor voting for feature extraction from unstructured point clouds. *Graphical Models 74*, 4 (jul 2012), 197–208. 2

[PWY*07] Pottmann H., Wallner J., Yang Y., Lai Y., Hu S.: Principal curvatures from the integral invariant viewpoint. *Computer Aided Geometric Design 24*, 8-9 (2007), 428–442. 7

[RBB*11] Rosman G., Bronstein M. M., Bronstein A. M., Wolf A., Kimmel R.: Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In *Scale Space and Variational Methods in Computer Vision*. Springer, 2011, pp. 725–736. 3

[SCC12] Strekalovskiy E., Chambolle A., Cremers D.: A convex representation for the vectorial mumford-shah functional. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (2012), IEEE, pp. 1712–1719. 3, 10

[Sha05] Shah J.: Skeletons of 3d shapes. In *Scale Space and PDE Methods in Computer Vision*. Springer, 2005, pp. 339–350. 3

[SLMR14] Song R., Liu Y., Martin R. R., Rosin P. L.: Mesh saliency via spectral processing. *ACM Trans. Graph. 33*, 1 (2014), 1–17. 3

[SOG09] Sun J., Ovsjanikov M., Guibas L.: A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion. *Computer Graphics Forum 28*, 5 (jul 2009), 1383–1392. 3

[tDGtal] DGtal: Digital Geometry tools, algorithms library: http://dgtal.org. 7

[WYL*14] 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 (2014), 18. 3, 10

[WZCF15] Wu X., Zheng J., Cai Y., Fu C.-W.: Mesh denoising using extended rof model with l1 fidelity. *Computer Graphics Forum 34*, 7 (2015), 35–45. 3, 10

[ZCL*13] Zhang J., Cao J., Liu X., Wang J., Liu J., Shi X.: Point cloud normal estimation via low-rank subspace clustering. *Computers & Graphics 37*, 6 (2013), 697–706. 2

[ZGW*16] Zhang Y., Geng G., Wei X., Zhang S., Li S.: A statistical approach for extraction of feature lines from point clouds. *Computers & Graphics 56* (2016), 31 – 45. 3

[ZWZD15] 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 (2015), 873–886. 3, 10