Eurographics Symposium on Geometry Processing 2020
Q. Huang and A. Jacobson
(Guest Editors)

Volume 39 (2020), Number 5

Interpolated corrected curvature measures for polygonal surfaces

J.-O. Lachaud[†][1] and P. Romon[2] and B. Thibert[3] and D. Coeurjolly[4]

[1]Université Savoie Mont Blanc, France
[2]Université Gustave Eiffel, France
[3]Université Grenoble Alpes, France
[4]Université de Lyon, CNRS, France

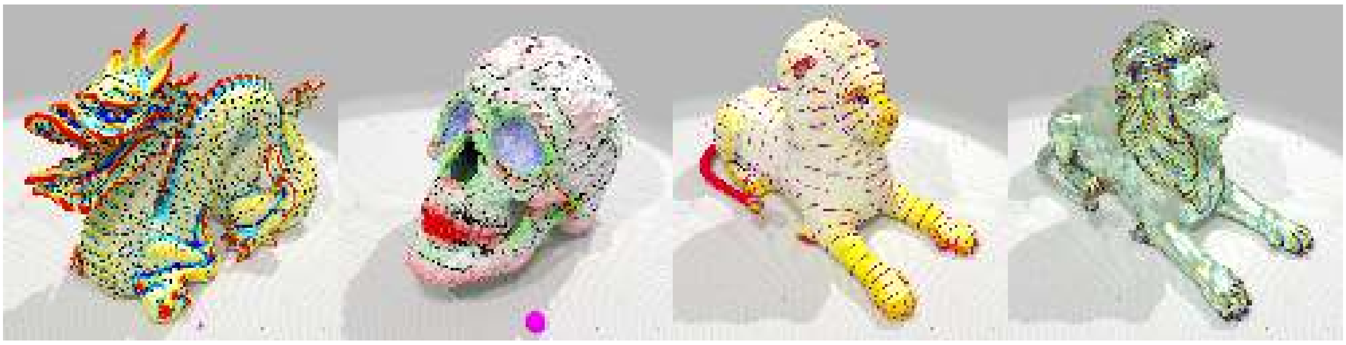


Figure 1: Left: *mean curvature estimation.* Middle left: *convex, concave and saddle parts estimation on noisy data.* Middle right: *second principal curvature and direction estimation on coarse "Lion" mesh.* Right: *mean curvature extraction on the same "Lion" mesh geometry but with normal vector field given by normal map.*

Abstract

A consistent and yet practically accurate definition of curvature onto polyhedral meshes remains an open problem. We propose a new framework to define curvature measures, based on the Corrected Normal Current*, which generalizes the normal cycle: it uncouples the positional information of the polyhedral mesh from its geometric normal vector field, and the user can freely choose the* corrected *normal vector field at vertices for curvature computations. A smooth surface is then built in the Grassmannian $\mathbb{R}^3 \times \mathbb{S}^2$ by simply interpolating the given normal vector field. Curvature measures are then computed using the usual Lipschitz–Killing forms, and we provide closed-form formulas per triangle. We prove a stability result with respect to perturbations of positions and normals. Our approach provides a natural scale-space for all curvature estimations, where the scale is given by the radius of the measuring ball. We show on experiments how this method outperforms state-of-the-art methods on clean and noisy data, and even achieves pointwise convergence on difficult polyhedral meshes like digital surfaces. The framework is also well suited to curvature computations using normal map information.*

CCS Concepts

• ***Computing methodologies*** → ***Shape analysis; Mesh geometry models;*** • ***Theory of computation*** → ***Computational geometry;***

1. Introduction

Context. This paper presents a new approach to define and estimate the curvatures over triangulated or polygonal meshes. Curvature is a differential property of 2-manifolds embedded in the usual 3D space, and is well defined for smooth enough surfaces. More precisely, we are interested in estimating at any position on the mesh its mean and Gaussian curvatures, but also its two principal curvatures and principal directions.

It is well known that curvature carries many information about the local shape geometry: convexity, concavity and saddle regions,

[†] This work has been partly funded by CoMeDiC ANR-15-CE40-0006 research grant.

unfoldability, approximation, sharp features, umbilics, etc. Several geometry processing applications exploit heavily curvature information, for instance shape matching, surface parameterization, classification, or artistic representation.

Although well-defined for C^2-smooth surfaces, extending curvature to polyhedral surface is not trivial, and certainly not unique. This has led to a considerable number of techniques to *approximate* curvature information on such data. We can roughly divide them into two categories: local fitting of a smooth simple surface, and integration of a local quantity related to some curvature integral. We discuss some representative approaches in the related works paragraph. We can already point out that few of them present theoretical guarantees, and even less offer quantified guarantees in the case of position/normal/sampling perturbations. Furthermore, as demonstrated by the extensive experimental evaluation of [VVP*16], their respective practical accuracy depend on the shape type, mesh sampling, alignment of edges, noise in position, normal accuracy and estimation, neighborhood size; and none of them is a clear winner as is.

In contrast, few approaches have tried to *define* curvature in a consistent way on more general subsets of the space than smooth 2-manifolds, exploiting a kind of natural continuity between smooth shapes and their polygonal approximation. Such works were carried in the mathematical community and lead essentially to two different models: the *normal cycle* [Win82, Fu93] and the *varifold* [Alm66]. These approaches were used concretely in computer science for curvature estimation much later (see related works below). Their common theoretical feature is to define curvature as *measures* instead of trying to define curvature *pointwise*. From a practical point of view, they require at least a radius parameter for integrating measures, they are more robust to noise than former methods but tend to be less accurate on "good" data.

Contribution. We propose here to exploit an extension of the normal cycle, called *corrected normal current* [LRT19], and to define classical curvature quantities as ratios of the induced corrected curvature measures. This generic framework defines curvatures over piecewise smooth surfaces, with a corrected unitary normal vector field that is smooth over smooth parts and may not be continuous on discontinuity parts. This framework was proven stable wrt curvature measures on smooth surfaces under hypotheses. To get easily computable formulas [LRT19] focused on the case of polyhedral mesh with constant per face corrected normals.

The present paper brings this idea one step forward by considering linearly interpolated corrected normal vector field and presents the following features:

- a new formalism and definitions of curvatures measures / quantities from uncoupled geometry/tangent bundle information;
- stability results with respect to noise either in the geometry or the normal vector field;
- closed form formulas for triangles and quads allowing fast estimation on meshes, with more accurate estimates than the state-of-the-art approaches;
- with adequate corrected normal vector field, this method achieves convergence on difficult data like Schwarz lantern or digital surfaces.

Related works. We review here representative methods to extract the whole curvature information, and not approaches devoted to only mean curvature information (like the ones using the Laplacian) or Gaussian curvature (like angle defect). A lot of methods fit or approximate locally the discrete data with a smooth enough surface, and deduce the local curvature information from the smooth surface. We can cite [GI04], where circles and parabola are fitted, and local polynomial approximation using points (and sometimes normals) like jet fitting [CP05]. These approaches have very limited accuracy results and are very sensitive to bad input. A closely related approach is to estimate locally the second fundamental form. [Rus04] estimates it per triangle with some linear regression and average it per vertex. Results are accurate where edges are well aligned with principal curvatures. The method is however too local to be useful on noisy data or to identify convex or concave features. It is also possible to reweigh how second fundamental forms are averaged with a statistical predictor [KSNS07], but this reweighting does not always improve results (see [VVP*16]).

Another group of methods rely on integral properties related to curvature information. Integral invariants [PWY*07] study the local Taylor expansion of several integral quantities when computing the intersection of a ball with the volume shape. Besides being only valid for closed surfaces, it is unclear how to choose the radius on a mesh: it cannot be reduced to zero and must be at least the size of a vertex umbrella while the Taylor expansion error hints towards an as small as possible radius. The Voronoi covariance measure [MOG11] exploits the distance transform to the surface to achieve a measure with stability guarantees. Eigendecomposition of the local covariance matrices provides unsigned curvature information, but the accuracy and convergence of such curvature estimates remain to be established. Pointwise approximations of the shape operator for polygonal meshes were proposed in [HP11] by convolution with r-local functions. They are proven convergent for a mesh with maximal edge length h when the normals of the polyhedral surface approximate the smooth normals with order $O(h)$ (hence vertex positions with $O(h^2)$), which is rather restrictive and difficult to achieve in practice.

There is a long history on the generalization of the curvatures in non-smooth geometry, from the seminal paper of H. Federer [Fed59]. Further works [Win82, Fu94] extended these notions of curvature measures to a wider class of objects including triangulations. Curvature measures of a smooth object can be approximated by the ones of a triangulation, provided that the points and the normals of the triangulation are close to the ones of the smooth object [Fu93, CSM03, CSM06]. Anisotropic measures were introduced to estimate principal curvatures and directions [CSM06, Mor08]. The normal cycle has also been extended to cloud of points using offsets in [CCSLT09], with some stability results. However these approaches do not provide a sound definition of curvatures when naive normals do not converge toward smooth normals [CCSLT09, HPW06]. This is the case for instance for the famous counterexample of the Schwarz lantern, or for digital surfaces that are boundary of voxel sets in 3D images (see discussion below).

Varifolds constitute another approach to define a geometry on a wide class of objects, e.g. it includes cloud of points with normals, and were first introduced to solve shape optimization problems [Alm66]. One can extract mean curvature information [BLM17] and recently a weak second fundamental form [BLM18]. This general framework does not yet provide easy and explicit stability or convergent rates.

Finally Váša *et al.* performed an extensive set of numerical experiments for many different curvature estimators [VVP*16], and their conclusion is that no curvature estimator outperforms all the others. On the contrary, analyzing some mesh property and choosing an estimator provides a much better estimator on average. According to us, such works show the need to develop a curvature theory with stability properties.

As mentioned earlier, the strength of our approach is to decouple the geometry from the normal vector field in the curvature computation, allowing us to prescribe a *good* normal vector field when the trivial one (*e.g.* orthogonal unit vectors to surface elements) is either noisy or irrelevant. This is exactly the case when dealing with digital surface (boundaries of subsets of $\mathbb{Z}^3$) where trivial normal vectors do not converge to the expected one when digitizing a smooth manifold on a finer and finer grid. On this kind of data, the Integral invariants approach [PWY*07] has led to multigrid convergent curvature estimators [CLL14], but it is shown in [LRT19] that our method is more accurate in practice.

Outline. In Section 2, we present how our corrected curvatures have simple expressions on polyhedral surfaces. Then in Section 3 we explain how our corrected curvature measures are built and we detail the computations that led to the closed-form formulas presented in the previous section. In Section 4 we prove the stability of our measures with respect to perturbation in geometry and normals. Section 5 provides experimental results on synthetic and real data, as well as some comparisons with classical methods. Finally, we conclude in Section 6.

2. Corrected curvature measures on meshes

We denote by $(\mathbf{x}_i)_{i=1\dots n}$ the positions of the n vertices of some polygonal mesh. Let also $(\mathbf{u}_i)_{i=1\dots n}$ be some normal vector field prescribed at these vertices. Before presenting the formal definition of our curvature measures, we first give their simple closed-form expressions on an arbitrary triangle of the input mesh.

Property 1 [Interpolated corrected curvature measures] The *interpolated corrected curvature measures* take the following values on a triangle τ_{ijk}, with vertices i, j, k:

$$\begin{aligned}
\mu^{(0)}(\tau_{ijk}) &= \frac{1}{2}\left\langle \bar{\mathbf{u}} \,\middle|\, (\mathbf{x}_j-\mathbf{x}_i)\times(\mathbf{x}_k-\mathbf{x}_i)\right\rangle,\\
\mu^{(1)}(\tau_{ijk}) &= \frac{1}{2}\left\langle \bar{\mathbf{u}} \,\middle|\, (\mathbf{u}_k-\mathbf{u}_j)\times\mathbf{x}_i+(\mathbf{u}_i-\mathbf{u}_k)\times\mathbf{x}_j+(\mathbf{u}_j-\mathbf{u}_i)\times\mathbf{x}_k\right\rangle,\\
\mu^{(2)}(\tau_{ijk}) &= \frac{1}{2}\left\langle \mathbf{u}_i \,\middle|\, \mathbf{u}_j\times\mathbf{u}_k\right\rangle,\\
\mu^{(\mathbf{X},\mathbf{Y})}(\tau_{ijk}) &= \frac{1}{2}\left\langle \bar{\mathbf{u}} \,\middle|\, \langle \mathbf{Y} \,|\, \mathbf{u}_k-\mathbf{u}_i\rangle \mathbf{X}\times(\mathbf{x}_j-\mathbf{x}_i)\right\rangle\\
&\quad -\frac{1}{2}\left\langle \bar{\mathbf{u}} \,\middle|\, \langle \mathbf{Y} \,|\, \mathbf{u}_j-\mathbf{u}_i\rangle \mathbf{X}\times(\mathbf{x}_k-\mathbf{x}_i)\right\rangle,
\end{aligned}$$

where $\langle\cdot\,|\,\cdot\rangle$ denotes the usual scalar product, $\bar{\mathbf{u}}=\frac{1}{3}(\mathbf{u}_i+\mathbf{u}_j+\mathbf{u}_k)$.

Measure $\mu^{(0)}$ is the *corrected area density* of the given triangle, $\mu^{(1)}$ its *corrected mean curvature density*, $\mu^{(2)}$ its *corrected Gaussian curvature density*. The (anisotropic) measure $\mu^{(\mathbf{X},\mathbf{Y})}$ is the trace of the *corrected second fundamental form* along directions $\mathbf{X}$ and $\mathbf{Y}$. While the smooth second fundamental form is naturally a symmetric 2-tensor, there is no easy way to define tangent directions at a vertex, so the anisotropic measure depends on two 3D vectors; when $\mathbf{X}$ and $\mathbf{Y}$ are tangent, $\mu^{(\mathbf{X},\mathbf{Y})}$ is close to the second fundamental form applied to these vectors, while its value along normal direction tends to zero asymptotically. See Section 4 for more details about the link with the second fundamental form.

Property 2 (Corrected curvature measures) Measures are extended to arbitrary subset of $\mathbb{R}^3$ as follows. Let symbol (k) be any of $\{(0),(1),(2),(\mathbf{X},\mathbf{Y})\}$. If B is any Borel set of $\mathbb{R}^3$ (any ball for instance), then the k-th corrected curvature measure on mesh M with triangles $(\tau_m)_{m=1\dots t}$ is:

$$\mu^{(k)}(B)=\sum_{m=1}^{t}\mu^{(k)}(\tau_m)\frac{\text{Area}(\tau_m\cap B)}{\text{Area}(\tau_m)}, \qquad (1)$$

where Area is the 2-dimensional Hausdorff measure.

In this paper, for simplicity, general polygonal meshes are implicitly triangulated before taking the measures, simply by splitting each non triangular face into triangles at its barycenter, and taking as normal the normalized average of the normals at the face vertices. We provide closed form formulas for generic quads in Appendix B, and digital surface quads in Appendix C. Note that our general framework allows more advanced implicit triangulation strategies such as the one described in [BHKB20], or other normal vectors interpolation schemes.

Definition 1 (Corrected mean and Gaussian curvatures) Let $B_\rho(\mathbf{x})$ be the ball of center $\mathbf{x}$ and radius ρ, then the *corrected mean curvature* $\hat{H}^{cnc}$ at $\mathbf{x}$, the *corrected Gaussian curvature* $\hat{G}^{cnc}$ at $\mathbf{x}$ are respectively defined as:

$$\begin{aligned}
\hat{H}^{cnc}(\mathbf{x},\rho) &:= \mu^{(1)}(B_\rho(\mathbf{x}))/\mu^{(0)}(B_\rho(\mathbf{x})), &(2)\\
\hat{G}^{cnc}(\mathbf{x},\rho) &:= \mu^{(2)}(B_\rho(\mathbf{x}))/\mu^{(0)}(B_\rho(\mathbf{x})). &(3)
\end{aligned}$$

For any point $\mathbf{x}$ within some face of M, the limits of the above quantities when ρ tends toward 0 are well defined (except in the unlikely event that every vertex normal is parallel to the face) and $\hat{H}^{cnc}(\mathbf{x},0)$ and $\hat{G}^{cnc}(\mathbf{x},0)$ are defined accordingly.

The anisotropic measures $\mu^{(\mathbf{X},\mathbf{Y})}$ are parametrized by two 3D vectors, but can be summed up into one 3×3 matrix $\mu^{(\mathbf{M})}$ using the three basis vector $(\mathbf{e}_i)_{i=1\dots 3}$ as follows:

$$\mu^{(\mathbf{M})} := \begin{bmatrix} \mu^{(\mathbf{e}_1,\mathbf{e}_1)} & \mu^{(\mathbf{e}_1,\mathbf{e}_2)} & \mu^{(\mathbf{e}_1,\mathbf{e}_3)}\\ \mu^{(\mathbf{e}_2,\mathbf{e}_1)} & \mu^{(\mathbf{e}_2,\mathbf{e}_2)} & \mu^{(\mathbf{e}_2,\mathbf{e}_3)}\\ \mu^{(\mathbf{e}_3,\mathbf{e}_1)} & \mu^{(\mathbf{e}_3,\mathbf{e}_2)} & \mu^{(\mathbf{e}_3,\mathbf{e}_3)} \end{bmatrix}.$$

The anisotropic measure $\mu^{(\mathbf{M})}$ is a 3×3 tensor which tends toward the second fundamental form, in the sense that two of its eigenvectors, resp. eigenvalues, tend to the principal directions, resp. principal curvatures. However, the matrix is not always symmetric (when $\mathbf{u}\neq\mathbf{n}$) and the last eigenvalue tends to zero while its eigenvector tends toward the normal vector $\mathbf{u}(\mathbf{x})$ at point $\mathbf{x}$. To obtain reliable directions even on flat parts, it is easier to extract the second fundamental form information by first symmetrizing the anisotropic measure and then adding a term $K\mathbf{u}(\mathbf{x})\otimes\mathbf{u}(\mathbf{x})$, for some big constant K, to force tangency of principal direction eigenvectors.

Definition 2 (Principal curvatures and directions) Letting

$$M_\rho(\mathbf{x}) := \frac{1}{2}\left(\mu^{(\mathbf{M})}(B_\rho(\mathbf{x})) + (\mu^{(\mathbf{M})}(B_\rho(\mathbf{x})))^{\mathsf{T}}\right) + K\mathbf{u}(\mathbf{x})\otimes\mathbf{u}(\mathbf{x}),$$

then the *corrected first and second principal curvatures,* $\hat{\kappa}_1^{cnc}$ *and* $\hat{\kappa}_2^{cnc}$ at $\mathbf{x}$, and their associated *corrected principal directions* $\hat{\mathbf{v}}_1^{cnc}$ *and* $\hat{\mathbf{v}}_2^{cnc}$ at $\mathbf{x}$ are respectively defined as:

$$\hat{\kappa}_1^{cnc}(\mathbf{x},\rho) := -\lambda_2(M_\rho(\mathbf{x}))/\mu^{(0)}(B_\rho(\mathbf{x})), \qquad (4)$$

$$\hat{\mathbf{v}}_1^{cnc}(\mathbf{x},\rho) := \mathbf{z}_2(M_\rho(\mathbf{x}))/\mu^{(0)}(B_\rho(\mathbf{x})), \qquad (5)$$

$$\hat{\kappa}_2^{cnc}(\mathbf{x},\rho) := -\lambda_1(M_\rho(\mathbf{x}))/\mu^{(0)}(B_\rho(\mathbf{x})), \qquad (6)$$

$$\hat{\mathbf{v}}_2^{cnc}(\mathbf{x},\rho) := \mathbf{z}_1(M_\rho(\mathbf{x}))/\mu^{(0)}(B_\rho(\mathbf{x})), \qquad (7)$$

where $\lambda_1(M) \leq \lambda_2(M) \leq \lambda_3(M)$ are the eigenvalues of M and $(\mathbf{z}_1(M), \mathbf{z}_2(M), \mathbf{z}_3(M))$ their associated eigenvectors.

The constant K should be chosen significantly greater than the maximal curvature and was set to 1000 in all our experiments.

3. Theory of corrected curvature measures

Let us recall briefly the ideas behind the corrected normal current framework presented in [LRT19], and explain why it is more flexible and more accurate than other approaches. Starting with any surface S, be it (piecewise-)continuous or discrete or even digital, we attach a normal cone at each point, and consider the corresponding set of points in the Grassmann bundle $\mathbb{R}^3\times\mathbb{S}^2$. Under mild regularity assuptions (see [LRT19, Definition 1]), this set is a piecewise continuous integral current N, which allows to define the curvature measures and to prove the stability. Various ways of defining this normal cone are possible, either from the geometric data (such as the normal bundle for a smooth surface, or, more generally, the normal cycle for a smooth or polyhedral surface [Win82]), or from algorithmically- or user-prescribed normals (including normal maps). (Note that the way to prescribe the normals will impact the quality of the final estimate.) Our framework allows to define normals that are constant or not on the faces, continuous or not across edges and vertices. The face-constant, edge-discontinuous case typical of digital surfaces was described in [LRT19], and we focus here on the edge-continuous, smoothly varying normal field defined by interpolating vertex-based data.

In the following, we formally define invariant forms and associated curvature measures. For a more complete overview in the smooth setting, please refer to J.M. Morvan's textbook *"Generalized cruvatures"* [Mor08].

Invariant and anisotropic forms. Curvature measures $\mu^{(k)}$ are obtained by integrating a normal current against canonical invariant 2-forms $\omega^{(k)}$ defined on $\mathbb{R}^3\times\mathbb{S}^2$ (for $k\in\{0,1,2\}$). For any point $(\mathbf{x},\mathbf{u})\in\mathbb{R}^3\times\mathbb{S}^2$ and tangent vectors $\boldsymbol{\xi},\boldsymbol{\nu}\in T_{(\mathbf{x},\mathbf{u})}(\mathbb{R}^3\times\mathbb{S}^2)$, we write $\boldsymbol{\xi}=(\boldsymbol{\xi}_p,\boldsymbol{\xi}_n)$ and $\boldsymbol{\nu}=(\boldsymbol{\nu}_p,\boldsymbol{\nu}_n)$ in $\mathbb{R}^3\times\mathbb{R}^3$ (separating position and velocity). Then the invariant forms are given by:

$$\omega^{(0)}_{(\mathbf{x},\mathbf{u})}(\boldsymbol{\xi},\boldsymbol{\nu}) = \det(\mathbf{u},\boldsymbol{\xi}_p,\boldsymbol{\nu}_p),\ \omega^{(2)}_{(\mathbf{x},\mathbf{u})}(\boldsymbol{\xi},\boldsymbol{\nu}) = \det(\mathbf{u},\boldsymbol{\xi}_n,\boldsymbol{\nu}_n)$$

$$\omega^{(1)}_{(\mathbf{x},\mathbf{u})}(\boldsymbol{\xi},\boldsymbol{\nu}) = \det(\mathbf{u},\boldsymbol{\xi}_p,\boldsymbol{\nu}_n) + \det(\mathbf{u},\boldsymbol{\xi}_n,\boldsymbol{\nu}_p)$$

Similarly, the anisotropic curvature form (for the directions $\mathbf{X},\mathbf{Y}$) is given by [CSM06]

$$\omega^{(\mathbf{X},\mathbf{Y})}_{(\mathbf{x},\mathbf{u})} = (\mathbf{u}\times\mathbf{X},0)^\flat \wedge (0,\mathbf{Y})^\flat,$$

where $\mathbf{w}^\flat$ denotes the linear map $\langle\mathbf{w}\mid\cdot\rangle$ dual to vector $\mathbf{w}$.

Corrected curvature measures. Fig. 2 illustrates the geometric idea of our approach in the 2D case. Curvature measures are defined by integration of invariant forms of the smooth object surface lifted in the Grassmannian (Fig. 2a). Polyhedral surfaces are lifted as the normal cycle (Fig. 2b), and if naive normals are not representative of the smooth surface, integration of invariant forms will not be precise or even close to expected. Our corrected curvature measures use an external vector field $\mathbf{u}$ for lifting the polyhedral mesh, which can be constant per face (Fig. 2c) as in [LRT19] or linearly interpolated (Fig. 2d) as we propose in the present paper.

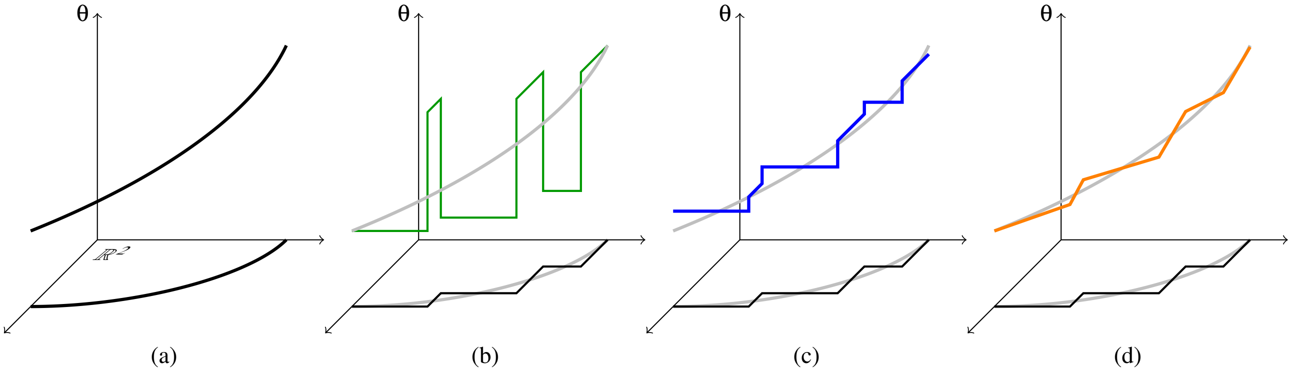
More formally, in our case, since there is a unique vector $\mathbf{u}(\mathbf{x})$ associated to every point $\mathbf{x}\in S$, the corrected normal cycle is a current that is characterized by its support $\operatorname{spt}(N) = \{(\mathbf{x},\mathbf{u}(\mathbf{x})),\mathbf{x}\in S\}$ and the corrected curvature measures are defined by $\mu^{(k)}(B) = \int_{\pi^{-1}(B)}\omega^{(k)}$, where $\pi:\operatorname{spt}(N)\to S$ is the projection, and B is any Borel set B on S. In the smooth case, when the current is exactly the normal bundle, we derive the Lipschitz–Killing curvature forms, which measure area, mean curvature density and Gaussian curvature density and also a form that measure the anisotropic curvature (see next section). In all other cases, we have closed form formulas, which are proven to be stable w.r.t. deformations, and hence converge to corresponding curvatures when S and the corrected normal $\mathbf{u}$ tends to a smooth limit.

In this section we prove the formulas given in Prop. 1, corresponding to the case when $\mathbf{u}$ is linearly interpolated over triangular faces from prescribed unitary vector data at vertices. Obviously, $\mathbf{u}$ is not in general unitary on the faces, so the formulas are not exact, and $\mathbf{u}$ ought to be replaced by $\mathring{\mathbf{u}} = \mathbf{u}/\|\mathbf{u}\|$. However, linear interpolated expressions allows to quickly derive simple closed formulas, which are asymptotical to the exact expression, as $\|\mathbf{u}\|$ tends to 1.

We parametrize the triangle τ by affine coordinates s,t. (but the interpolation could be quadratic (for quadrangular faces), or more general, e.g. splines, see appendices). So the position point is $\mathbf{x}(s,t) = \mathbf{x}_0 + s(\mathbf{x}_1-\mathbf{x}_0) + t(\mathbf{x}_2-\mathbf{x}_0)$ for (s,t) in the standard 2-simplex Δ, and the $\mathbf{x}_i$ are the triangle vertices (following the orientation). The corrected normal is interpolated on the triangle as $\mathbf{u}(s,t) = \mathbf{u}_0 + s(\mathbf{u}_1-\mathbf{u}_0) + t(\mathbf{u}_2-\mathbf{u}_0)$, where $\mathbf{u}_i$ is the prescribed normal at $\mathbf{x}_i$.

Because $\mathbf{u}$ is continuous, the measures are diffuse and localized on the faces, and are obtained as the pullbacks $\Gamma_{\mathbf{u}}^*\omega^{(k)}$ of the canonical invariant 2-forms, where $\Gamma_{\mathbf{u}}$ is the map $\mathbf{x}\mapsto(\mathbf{x},\mathbf{u}(\mathbf{x}))$. Therefore, on a triangle τ,

$$\mu^{(0)}(\tau) = \int_\tau \Gamma_{\mathbf{u}}^*\omega^{(0)} = \int_\Delta \det\left(\mathbf{u},\frac{\partial\mathbf{x}}{\partial s},\frac{\partial\mathbf{x}}{\partial t}\right)\mathrm{d}s\,\mathrm{d}t$$



(a) (b) (c) (d)

Figure 2: *Various approximations of the normal bundle of a planar curve. The Grassmann bundle $\mathbb{R}^2\times\mathbb{S}^1$ is represented as $\mathbb{R}^3$ where the third coordinate is the circular angle θ. The target is a smooth circular arc lifting (a) to $\mathbb{R}^2\times\mathbb{S}^1$ as a piece of a helix (in black). Its approximation by a digital curve can be lifted either as (b) the normal cycle, in which case the normal on each edge follows the axes (hence lies at height $k\pi/2$ for some integer k); therefore the extra circular arcs at the vertices are vertical edges of length $\pm\pi/2$; (c) corrected normals as in [LRT19] which are constant on the edges and interpolated at the vertices ; (d) corrected normals that are set at the vertices and interpolated along the edges. Obviously the green curve lies far away from the smooth helix, no matter the discretization step, which may induce errors in the curvature estimates, since it is an integral along the curve. The blue curve fares better, because the estimated normals are closer to the smooth curve's normals, but being constant along the edges creates a distance between the curve and the helix. Finally, the last interpolation along the edges fits the helix more closely, further improving the estimate.*

$$
\begin{aligned}
&= \int_\Delta \langle \mathbf{u} \mid (\mathbf{x}_1-\mathbf{x}_0)\times(\mathbf{x}_2-\mathbf{x}_0)\rangle \,\mathrm{d}s\,\mathrm{d}t\\
&= \Big\langle \int_\Delta (\mathbf{u}_0 + s(\mathbf{u}_1-\mathbf{u}_0) + t(\mathbf{u}_2-\mathbf{u}_0))\mathrm{d}s\,\mathrm{d}t\\
&\qquad \mid (\mathbf{x}_1-\mathbf{x}_0)\times(\mathbf{x}_2-\mathbf{x}_0)\Big\rangle\\
&= \frac{1}{2}\langle \bar{\mathbf{u}} \mid (\mathbf{x}_1-\mathbf{x}_0)\times(\mathbf{x}_2-\mathbf{x}_0)\rangle\,.
\end{aligned}
$$

Recall that $\bar{\mathbf{u}} = \frac{1}{3}(\mathbf{u}_0+\mathbf{u}_1+\mathbf{u}_2)$. Using pullback $\Gamma_{\mathbf{u}}^*\omega^{(1)}$, we get

$$
\begin{aligned}
\mu^{(1)}(\tau) &= \int_\Delta \left(\det\left(\mathbf{u}, \frac{\partial \mathbf{x}}{\partial s}, \frac{\partial \mathbf{u}}{\partial t}\right) + \det\left(\mathbf{u}, \frac{\partial \mathbf{u}}{\partial s}, \frac{\partial \mathbf{x}}{\partial t}\right)\right)\mathrm{d}s\,\mathrm{d}t\\
&= \int_\Delta \langle \mathbf{u} \mid (\mathbf{x}_1-\mathbf{x}_0)\times(\mathbf{u}_2-\mathbf{u}_0)\\
&\qquad + (\mathbf{u}_1-\mathbf{u}_0)\times(\mathbf{x}_2-\mathbf{x}_0)\rangle \mathrm{d}s\mathrm{d}t\\
&= \frac{1}{2}\langle \bar{\mathbf{u}} \mid (\mathbf{x}_1-\mathbf{x}_0)\times(\mathbf{u}_2-\mathbf{u}_0) + (\mathbf{u}_1-\mathbf{u}_0)\times(\mathbf{x}_2-\mathbf{x}_0)\rangle\\
&= \frac{1}{2}\langle \bar{\mathbf{u}} \mid (\mathbf{u}_2-\mathbf{u}_1)\times\mathbf{x}_0 + (\mathbf{u}_0-\mathbf{u}_2)\times\mathbf{x}_1 + (\mathbf{u}_1-\mathbf{u}_0)\times\mathbf{x}_2\rangle\,.
\end{aligned}
$$

And using pullback $\Gamma_{\mathbf{u}}^*\omega^{(2)}$, we get

$$
\begin{aligned}
\mu^{(2)}(\tau) &= \int_\Delta \det\left(\mathbf{u}, \frac{\partial \mathbf{u}}{\partial s}, \frac{\partial \mathbf{u}}{\partial t}\right)\mathrm{d}s\,\mathrm{d}t\\
&= \int_\Delta \langle \mathbf{u} \mid (\mathbf{u}_1-\mathbf{u}_0)\times(\mathbf{u}_2-\mathbf{u}_0)\rangle\,\mathrm{d}s\,\mathrm{d}t\\
&= \frac{1}{2}\langle \bar{\mathbf{u}} \mid (\mathbf{u}_1-\mathbf{u}_0)\times(\mathbf{u}_2-\mathbf{u}_0)\rangle\,.
\end{aligned}
$$

Similarly we show that the anisotropic measure is, for given vectors $\mathbf{X},\mathbf{Y}$:

$$
\begin{aligned}
\mu^{(\mathbf{X},\mathbf{Y})}(\tau) &= \int_\Delta (\mathbf{u}\times\mathbf{X},0)^\flat \wedge (0,\mathbf{Y})^\flat \left(\frac{\partial \Gamma_{\mathbf{u}}}{\partial s}, \frac{\partial \Gamma_{\mathbf{u}}}{\partial t}\right)\mathrm{d}s\,\mathrm{d}t\\
&= \frac{1}{2}\langle \bar{\mathbf{u}} \mid \langle \mathbf{Y} \mid \mathbf{u}_2-\mathbf{u}_0\rangle \mathbf{X}\times(\mathbf{x}_1-\mathbf{x}_0)\\
&\qquad - \langle \mathbf{Y} \mid \mathbf{u}_1-\mathbf{u}_0\rangle \mathbf{X}\times(\mathbf{x}_2-\mathbf{x}_0)\rangle\,.
\end{aligned}
$$

Computations are detailed in Appendix A. As expected, all the above expressions are invariant by circular permutation of the indices.

4. Stability of corrected curvature measures

We show in this section that the corrected curvatures of $(M,\mathbf{u})$ approximate well the curvature measures of a surface S of class C^2 provided that the mesh M is close to S in the Hausdorff sense, that the vector field $\mathbf{u}$ approximates well the normal $\mathbf{n}_S$ of S and that the length of the edges of M are small.

4.1. Stability result

We first need to recall the notion of reach introduced by Federer [Fed59] as well as the notion of curvature measures for a smooth surface.

Sets with positive reach. The distance function d_K of a compact set K of $\mathbb{R}^d$ associates to any point $\mathbf{x}$ of $\mathbb{R}^d$ its distance to K, namely $d_K(\mathbf{x}) := \min_{y\in K} d(\mathbf{x},\mathbf{y})$, where d is the Euclidean distance on $\mathbb{R}^d$. For a given real number $\varepsilon > 0$, we denote by $K^\varepsilon := \{\mathbf{x}\in\mathbb{R}^d, d_K(\mathbf{x}) \le \varepsilon\}$ the ε-offset of K. The Hausdorff distance $d_{\mathrm{H}}(K,K')$ between two compact sets K and K' is the minimum number ε such that $K \subset K'^\varepsilon$ and $K' \subset K^\varepsilon$.

The medial axis of K is the set of points $\mathbf{x} \in \mathbb{R}^d$ such that the distance $d(\mathbf{x}, K)$ is realized by at least two points $\mathbf{y}$ and $\mathbf{y}'$ in K. The reach of K, denoted by $\mathrm{reach}(K)$, is the infimum distance between K and its medial axis. As a consequence, whenever the reach is positive and $\varepsilon < \mathrm{reach}(K)$, the projection map

$$\pi_K : K^{\varepsilon} \to K$$

is well-defined. It is well know that smooth compact submanifolds have positive reach [Fed59].

Curvature measures of a smooth surface. When S is an oriented surface of class C^2, and $\mathbf{u}$ is the unit normal vector field, the corrected curvature measures coincide with the curvature measures and satisfy for any Borel set B of $\mathbb{R}^3$

$$\mu^{(0)}(B \cap S) = \int_{B\cap S} \mathrm{d}\mathbf{x},$$
$$\mu^{(1)}(B \cap S) = \int_{B\cap S} H(\mathbf{x})\mathrm{d}\mathbf{x},$$
$$\mu^{(2)}(B \cap S) = \int_{B\cap S} G(\mathbf{x})\mathrm{d}\mathbf{x},$$

where H and G denote the mean and Gaussian curvature. The second fundamental form at a point $\mathbf{x} \in S$ is a bilinear form defined on the tangent space $T_{\mathbf{x}}S \subset \mathbb{R}^3$. We denote by $\mathrm{II}_{\mathbf{x}} : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$ its bilinear extension to $\mathbb{R}^3$ by imposing that $\mathrm{II}_{\mathbf{x}}(\mathbf{X},\mathbf{Y}) = 0$ if either $\mathbf{X}$ or $\mathbf{Y}$ is orthogonal to $T_{\mathbf{x}}S$. The anisotropic curvature form in the direction $(\mathbf{X},\mathbf{Y})$ satisfies [CSM03]

$$\mu_S^{(\mathbf{X},\mathbf{Y})}(B \cap S) = \int_{B\cap S} \mathrm{II}_{\mathbf{x}}(\mathbf{X},\mathbf{Y})\mathrm{d}\mathbf{x} = \left(\int_{B\cap S} \mathrm{II}_{\mathbf{x}}\mathrm{d}\mathbf{x}\right)(\mathbf{X},\mathbf{Y}).$$

We can now state a stability result for corrected curvature measures. We denote by $\mu_S^{(k)}$ the curvature measure of a smooth surface S and by $\mu_{M,\mathbf{u}}^{(k)}$ the corrected curvature measures of a compact mesh M with a vector field $\mathbf{u}$ which is unitary on the vertices and defined by linear interpolation on each triangle.

Theorem 1 Let S be a compact surface of $\mathbb{R}^3$ of class C^2 with no boundary, M be a compact mesh with no boundary and $\mathbf{u} : M \to \mathbb{R}^3$ a normal vector field which is unitary on the vertices and defined by linear interpolation on each triangle. We assume that:

- there exists a non-empty open set U of S such that $\pi_S : \pi_S^{-1}(U) \cap M \to S$ is injective;
- $\varepsilon := d_{\mathrm{H}}(S,M) < \mathrm{reach}(S)/2$.

Then the corrected curvature measures of $(M,\mathbf{u})$ are close to the curvature measures of S. More precisely for any unit vector $\mathbf{X}$ and $\mathbf{Y}$, for any $(k) \in \{(0),(1),(2),(\mathbf{X},\mathbf{Y})\}$ and any connected union $B = \cup_{i\in I}\tau_i$ of triangles of M one has

$$|\mu_{M,\mathbf{u}}^{(k)}(B) - \mu_S^{(k)}(\pi_S(B))|$$
$$\leq 16\max(1,\kappa_{\max}^2)(1+\mathrm{L}_{\mathbf{u}}^4)\left[(\eta+\varepsilon)(\mathrm{Area}(B)+\mathrm{Length}(\partial B))+\delta^4 N_B\right]$$

where $\eta := \sup_{\mathbf{x}\in M}\|\mathbf{u}(\mathbf{x}) - \mathbf{n}(\pi_S(\mathbf{x}))\|$ is the normal error, δ is the length of the longest edge of M, $\mathbf{n}$ is the unit normal vector field of S, $\mathring{\mathbf{u}} = \mathbf{u}/\|\mathbf{u}\|$, N_B is the number of triangles in B, $\kappa_{\max}$ is the maximum of the absolute value of the principal curvatures of S and $\mathrm{L}_{\mathbf{u}}$ is the Lipschitz constant of $\mathring{\mathbf{u}}$.

In the bound of the theorem, the error term $N_B\delta^4$ comes from the fact that the curvature measures rely on a piecewise linear vector $\mathbf{u}$ while the theory of corrected normal cycle was developed for unitary vectors. Note that if B is large, the term N_B might be large but is asymptotically compensated by the factor δ^4.

4.2. Proof of Theorem 1

It is known that the corrected curvature measures are stable when the corrected normal is unitary [LRT19]. It is therefore sufficient to show that the numerical error induced by replacing the unit vector field $\mathring{\mathbf{u}}$ by the piecewise linear vector field $\mathbf{u}$ is controlled, which is done in Proposition 1.

Let us first recall that Theorem 1 of [LRT19] implies that for every $(k) \in \{(0),(1),(2)\}$

$$|\mu_{M,\mathring{\mathbf{u}}}^{(k)}(B) - \mu_S^{(k)}(\pi_S(B))|$$
$$\leq \quad 16\max(1,\kappa_{\max}^2)(1+\mathrm{L}_{\mathbf{u}}^2)\,(\eta+\varepsilon)\ (\mathrm{Area}(B)+\mathrm{Length}(\partial B)).$$

This theorem can be easily adapted to the anisotropic curvature measures by using the fact that the form $\omega^{(\mathbf{X},\mathbf{Y})}$ and its exterior derivative $\mathrm{d}\omega^{(\mathbf{X},\mathbf{Y})}$ are bounded.

The proof then directly follows from the following proposition.

Proposition 1 The error related to the fact that the interpolated $\mathbf{u}$ is generally not unitary is given by

$$|\mu_{M,\mathring{\mathbf{u}}}^{(k)}(B) - \mu_{M,\mathbf{u}}^{(k)}(B)| \leq \frac{\delta^4}{2}\ \max(1,\mathrm{L}_{\mathbf{u}}^4)\ N_B.$$

Proof Let τ be a triangle of B whose vertices are denoted by $\mathbf{x}_0$, $\mathbf{x}_1$ and $\mathbf{x}_2$. Let us denote by $\mathbf{h}_1 = \mathbf{x}_1 - \mathbf{x}_0$, $\mathbf{h}_2 = \mathbf{x}_2 - \mathbf{x}_0$, $\mathbf{v}_1 = \mathbf{u}_1 - \mathbf{u}_0$ and $\mathbf{v}_2 = \mathbf{u}_2 - \mathbf{u}_0$. Note that $\|\mathbf{h}_i\| \leq \delta$ and since $\mathring{\mathbf{u}}$ is $\mathrm{L}_{\mathbf{u}}$ -Lipschitz $\|\mathbf{v}_i\| \leq \mathrm{L}_{\mathbf{u}}\|\mathbf{h}_i\| \leq \mathrm{L}_{\mathbf{u}}\delta$. Furthermore for every $\mathbf{x} \in \tau$ a calculation shows that

$$\|\mathring{\mathbf{u}}(\mathbf{x}) - \mathbf{u}(\mathbf{x})\| \leq \frac{(\delta\,\mathrm{L}_{\mathbf{u}})^2}{2}.$$

From the formula of the previous section, one has

$$|\mu_{M,\mathbf{u}}^{(0)}(\tau) - \mu_{M,\mathring{\mathbf{u}}}^{(0)}(\tau)|$$
$$\leq \int_{\Delta} \left|\det(\mathbf{u},\mathbf{h}_1,\mathbf{h}_2) - \det\left(\mathring{\mathbf{u}},\mathbf{h}_1,\mathbf{h}_2\right)\right| \mathrm{d}s\,\mathrm{d}t$$
$$= \int_{\Delta} |\left\langle \mathbf{u} - \mathring{\mathbf{u}} \mid \mathbf{h}_1 \times \mathbf{h}_2 \right\rangle| \mathrm{d}s\,\mathrm{d}t$$
$$\leq \frac{\delta^4 \mathrm{L}_{\mathbf{u}}^2}{4}.$$

Similarly

$$|\mu_{M,\mathbf{u}}^{(2)}(\tau) - \mu_{M,\mathring{\mathbf{u}}}^{(2)}(\tau)| \leq \int_{\Delta} |\left\langle \mathbf{u} - \mathring{\mathbf{u}} \mid \mathbf{v}_1 \times \mathbf{v}_2 \right\rangle| \mathrm{d}s\,\mathrm{d}t \leq \frac{\delta^4\,\mathrm{L}_{\mathbf{u}}^4}{4}.$$

For the mean curvature, one has

$$|\mu_{M,\mathbf{u}}^{(1)}(\tau) - \mu_{M,\mathring{\mathbf{u}}}^{(1)}(\tau)|$$
$$\leq \int_{\Delta} |\left\langle \mathbf{u} - \mathring{\mathbf{u}} \mid \mathbf{h}_1 \times \mathbf{v}_2 + \mathbf{h}_2 \times \mathbf{v}_1 \right\rangle| \mathrm{d}s\,\mathrm{d}t$$
$$\leq \frac{\delta^4\,\mathrm{L}_{\mathbf{u}}^3}{2}.$$

For the anisotropic curvature, on has

$$\begin{aligned}
&\mu^{(\mathbf{X},\mathbf{Y})}_{M,\mathbf{u}}(\tau) - \mu^{(\mathbf{X},\mathbf{Y})}_{M,\mathring{\mathbf{u}}}(\tau) \\
&= \int_\Delta \begin{vmatrix} \langle \mathbf{u} \times \mathbf{X} \mid \mathbf{h}_1 \rangle & \langle \mathbf{u} \times \mathbf{X} \mid \mathbf{h}_2 \rangle \\ \langle \mathbf{Y} \mid \mathbf{v}_1 \rangle & \langle \mathbf{Y} \mid \mathbf{v}_2 \rangle \end{vmatrix} \mathrm{d}s\,\mathrm{d}t \\
&\quad - \int_\Delta \begin{vmatrix} \langle \mathring{\mathbf{u}} \times \mathbf{X} \mid \mathbf{h}_1 \rangle & \langle \mathring{\mathbf{u}} \times \mathbf{X} \mid \mathbf{h}_2 \rangle \\ \langle \mathbf{Y} \mid \mathbf{v}_1 \rangle & \langle \mathbf{Y} \mid \mathbf{v}_2 \rangle \end{vmatrix} \mathrm{d}s\,\mathrm{d}t \\
&= \int_\Delta \begin{vmatrix} \langle \mathbf{u} - \mathring{\mathbf{u}} \mid \mathbf{X} \times \mathbf{h}_1 \rangle & \langle \mathbf{u} - \mathring{\mathbf{u}} \mid \mathbf{X} \times \mathbf{h}_2 \rangle \\ \langle \mathbf{Y} \mid \mathbf{v}_1 \rangle & \langle \mathbf{Y} \mid \mathbf{v}_2 \rangle \end{vmatrix} \mathrm{d}s\,\mathrm{d}t
\end{aligned}$$

Since $\mathbf{X}$ and $\mathbf{Y}$ are unit vectors, we deduce that

$$|\mu^{(\mathbf{X},\mathbf{Y})}_{M,\mathbf{u}}(\tau) - \mu^{(\mathbf{X},\mathbf{Y})}_{M,\mathring{\mathbf{u}}}(\tau)| \leq \frac{1}{2} \frac{\delta^2 \mathrm{L}_{\mathbf{u}}^2}{2} 2\mathrm{L}_{\mathbf{u}} \delta^2 = \frac{\delta^4 \, \mathrm{L}_{\mathbf{u}}^3}{2}.$$

We get the result by summing over the triangles of B. □

5. Experimental evaluation

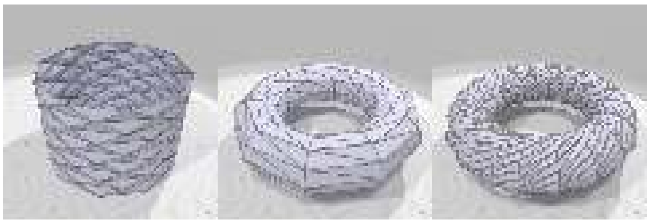
As noted by Váša *et al.* [VVP*16], it is difficult — and often biased — to perform a full set of experiments to show that some curvature estimator is better than others in most of the cases. However, as said above, our approach does not *estimate* curvatures of an undefined underlying smooth object. It *defines* a concrete $C^{1,1}$-smooth surface X in the Grassmannian $\mathbb{R}^3 \times \mathbb{S}^2$, and then computes a *close approximation* of its curvature measures. It follows that surfaces close to X in the Grassmannian have indeed close measures.

For instance, on a triangle, we are not estimating the curvatures as some variation of normals between its vertices, we are computing the curvatures of a curved bend triangle patch whose normals are aligned with any user-prescribed interpolated normal vector field.

Corrected normal vector field u. In our experiments we will use different vector fields **u** prescribed at vertices (assuming **n** is the canonical geometric normal vector field — orthogonal to faces —): **u**=GEOM means that vertex normals are averaged from surrounding **n** face normals. **u**=GEOM-MAX means that vertex normals are averaged from surrounding **n** face normals with Max's weights [Max99]. **u**=TRUE means that vertex normals are the ground truth normals. When processing a digital surface digitized with sampling h (see discussion below), **u**=II means that vertex normals are computed with digital Integral Invariant method with $r = 3h^{\frac{1}{2}}$ [CLL14]. In supplementary material, we consider additional normal vector fields given by some regularization of above-mentioned ones.

Pointwise convergence of curvatures. First, we numerically verify the convergence of our curvature estimators. We compare our approach with two representative methods of the literature. The first one is Rusinkiewicz's method [Rus04], which is local and precise for rather clean meshes. The second one is Cohen-Steiner and Morvan's normal cycle approach [CSM03], which requires a measuring ball and is more stable for noisy data. We slightly adapt it to use a prescribed normal vector field **u** and not only the canonical normal vector field **n**. Since we observed similar behaviors between estimators whether we look for mean, Gaussian and principal curvatures, we will plot only results for one estimator at a time.

We first run experiments on very specific generated meshes presented on Figure 3, where convergence is achievable. As seen on Figure 5, Rusinkiewicz estimator $\hat{\kappa}_1^{rz}$ [Rus04] performs best on "torus" shape, but this is not true anymore on twisted torus or Schwartz lantern. In fact, this estimator is very good when principal directions are aligned with edges, but becomes much less accurate otherwise. Normal cycle estimator $\hat{\kappa}_1^{nc}$ [CSM03] requires a minimal radius ρ and, even with this averaging, it is systematically less accurate than the others. Our estimator $\hat{\kappa}_1^{cnc}$ is competitive with $\hat{\kappa}_1^{rz}$ and can achieve similar accuracy on "torus" with an appropriate radius, while it can outperform it on more generic shapes. On the specific "lantern" shape, with true normals, our estimator outperforms by several orders of magnitude the other ones. Note that our mean and Gaussian curvature estimations are exact on the "lantern", up to numerical precision, and are hence not plotted.



"lantern" "torus" "twisted torus"

Figure 3: *Generated meshes whose sampling is parameterized by two parameters m and n, the number of vertices along transversal and longitudinal slices.*

We look now at more general polyhedral shapes whose edges are not so regular and not nicely aligned with principal directions. We recall that polygonal faces are simply triangulated at their barycenter with a normal equal to the average of surrounding vertex normals. Figure 4 shows how we built polyhedral meshes approximating a polynomial surface S. If h is a gridstep, then its h-digitized shape Z_h is defined as $\{\mathbf{x} \in h\mathbb{Z}^3, S(\mathbf{x}) \leq 0\}$. The "PRIMAL" surface is the topological boundary of the voxels of Z_h, seen as a union of cubes with side length h, generally called a digital surface. It does not sample S and its canonical normals are bad whatever h. The "DUAL" surface is the dual mesh to "PRIMAL". The "PDUAL" surface is obtained by projecting "DUAL" vertices and cell combinatorics onto S and corresponds to the "Marching-cubes" surface of the polynomial implicit function.

Such surfaces are particularly challenging because for "PRIMAL" and "DUAL" surfaces, their canonic geometric normal vectors do not converge to the expected ones when digitizing a smooth manifold as h tends to zero. Even if we consider multigrid convergent normal vector estimators (*e.g.* **u** =II [CLL14]), there is no bijection between such surfaces and their underlying smooth manifolds [LT16]. In that setting, no *pointwise convergence* can be expected from local curvature estimation method where normals are deduced from vertex positions [HPW06]. Even for "PDUAL" surface, convergence of normals is too slow to expect pointwise convergence of a local estimator that exploits information on a bounded number of triangles [HP11]. Hence, Ruzinkiewicz's method [Rus04] is doomed to fail at high resolution/sampling, even when using Max's method [Max99] for estimating normals at vertices. This is illustrated on Figure 6 (left), where we check point-

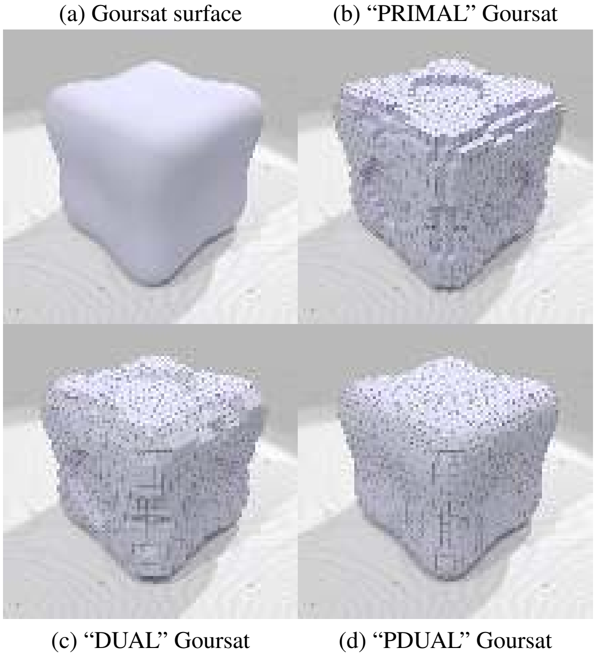


Figure 4: *Possible input shapes for convergence tests, given input polynomial Goursat surface* $3(x^4+y^4+z^4)-200(x^2+y^2+z^2)=800$ *digitized at gridstep* $h=1$*: (a) continuous polynomial surface, (b) digital surface bordering interior voxels, (c) dual surface to "PRIMAL" surface, (d) projection of "DUAL" vertices onto continuous surface.*

wise convergence on "PDUAL" goursat with finer and finer grid step h (graphs to be read from right to left), which is a rather nice sampling of the true Goursat surface. It shows that one need true normals to get pointwise convergence of curvatures, while geometric normals are not enough. Max's weighting scheme only improves the constant (in this case), but does not ensure convergence. Estimations of $\hat{H}^{rz}$ and $\hat{H}^{cnc}$ are indistinguishable. However, as shown by Figure 6 (middle), if one uses a radius ρ for the mesure such that ρ/h increases as h tends toward zero, the normal cycle estimator $\hat{H}^{nc}$ can achieve better results than estimator $\hat{H}^{rz}$. Even better our corrected mean curvature $\hat{H}^{cnc}$ converges toward the true mean curvature for $\rho=\Theta(h^{\frac{1}{2}})$ and is much more accurate than the others.

Finally, on Figure 6 (right), we test mean curvature estimators on the more difficult "PRIMAL" digital surface, but we provide as input normal field $\mathbf{u}$ the result of the digital Integral Invariant normal estimator [CLL14], known to be convergent. Our stability result guarantees convergence of measures, but convergence of curvatures is not obvious. We indeed observe convergence with speed $\Theta(h^{\frac{2}{3}})$, although only $\mathcal{O}(h^{\frac{1}{3}})$ has been proven in [LRT19]. Both estimators $\hat{H}^{rz}$ and $\hat{H}^{nc}$ fail for that kind of data. Other graphs showing ℓ_∞-convergence or on "DUAL" surface, or measuring Gaussian or principal curvatures can be found in supplementary. All of them show a similar behavior for all three estimators.

Experiments on classic datasets. We test our estimators on the Stanford "Armadillo" shape, which is a mesh of good quality with regular sampling but rich geometrical details. As shown on Figure 7 and Figure 8, our estimator resembles Rusinkiewicz's method for $\rho=0$ while for greater ρ it provides results visually similar to normal cycle approach. Our method thus encompasses the two previous ones, with a clear scale-space starting from $\rho=0$ till ∞.

Furthermore, as shown on Figure 9, it is often difficult to find a correct radius for normal cycle method, especially when the mesh is coarse at certain places and precise at others. The normal cycle cannot set its radius close to 0 and requires to average computations made on edges and faces (for H and principal and curvature directions) or vertices (for G). This is because the shape of the normal cycle in the Grassmannian is piecewise planar, cylindrical or spherical, instead of the C^2-flexibility like in our method.

On the contrary, our approach works well on that kind of data mixing coarse and fine sampling, as seen on Figure 10. It can also be seen on Figure 11 and Figure 12 that our approach gives very similar results for principal curvatures and principal directions on a nice polygonal mesh ("Octaflower"), using its geometric normals, and its voxel digitization, using digital Integral Invariant normal estimations. Note that Rusinkiewicz estimators are too unstable on "Octaflower" mesh and meaningless on its voxel digitization, while the normal cycle is incorrect on the voxel digitization.

We can also decompose a shape into its concave, convex, hyperbolic/saddle and flat parts using principal curvatures, as illustrated on Figure 13. A zero-radius measure is generally meaningless on a polyhedral mesh, even on a nice mesh like "Octaflower". This makes $\hat{H}^{rz}$ not useful for this task since it is purely local. Increasing measure radius gives more meaningful results, but clearly estimator $\hat{H}^{cnc}$ better estimates principal curvatures than $\hat{H}^{nc}$. This comes from the better estimation of the adequation between normals and differential of positions in our anisotropic measure.

Robustness to noise and tests on LIDAR data. We tested also the robustness of principal curvatures estimators $\hat{\kappa}_1^{cnc}$ and $\hat{\kappa}_2^{cnc}$ with respect to a uniform random perturbation of positions of radius $n\bar{e}(p)$, where $\bar{e}(p)$ is the average length of edges incident to p. Results are displayed on Figure 14 and Figure 15, with noise increasing from $n=0\%$ to $n=90\%$. The corrected normal vector field $\mathbf{u}$ is computed from the geometric normals $\mathbf{n}$ by averaging it four times. As expected, concave, convex, hyperbolic/saddle and flat parts are stable for our method, if the measure radius is adapted. Note the consistency throughout the scales. On the contrary, Rusinkiewicz approach is not useful for noisy data, while normal cycle anisotopric measure has a tendency to overestimates curvatures, making hyperbolic zones disappear.

We show how our estimator behaves on real LIDAR data on Figure 16. It is clearly an advantage that our method works on polyhedral meshes without manifold topology, since this input data presents many manifold inconsistencies. More experiments on robustness or LIDAR data can be found in supplementary.

Curvature extraction on normal map textures. Since normals and positions are decoupled in our framework, it is straightforward to compute all curvature information at the precision of a given normal map. Positions are given by the barycentric coordinates of vertices, while normals $\mathbf{u}$ are given by the normal map given a 2D parametrization of the mesh. An example of mean and Gaus-

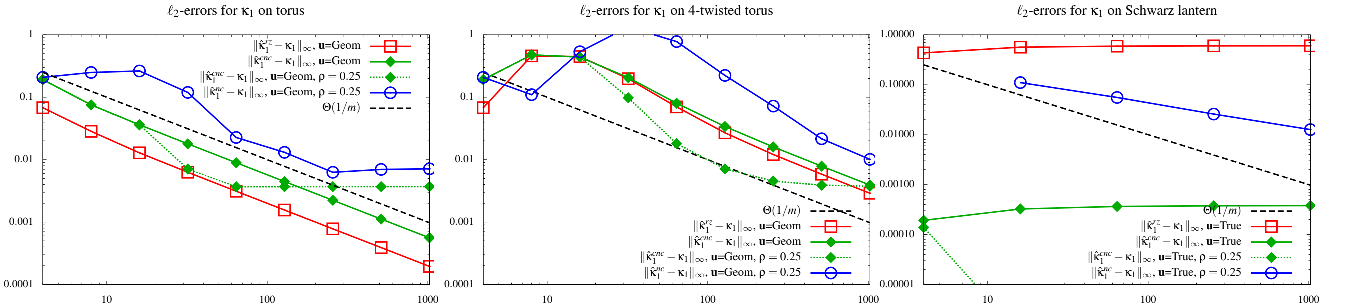


Figure 5: *Pointwise convergence for first principal curvature: in abscissa, parameter m giving the number of transversal slices, in ordinate ℓ_2-error estimation of first principal curvature.* ***Left***: *torus with $n = m$.* ***Middle***: *4-twisted torus with $n = m$.* ***Right***: *Schwarz lantern with $n = \sqrt{m}$.*

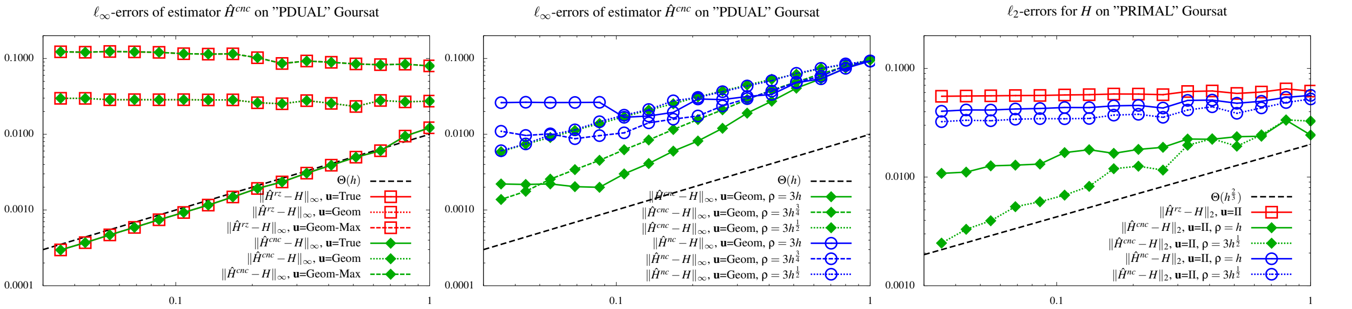


Figure 6: *Pointwise convergence for mean curvature H on (left and middle) "PDUAL" Goursat and (right) "PRIMAL" Goursat : in abscissa, parameter h giving the sampling grid step (left: finest scale, right: coarsest scale).* Left*: ℓ_∞-error for estimator $\hat{H}^{rz}$ and estimator $\hat{H}^{cnc}$ with $\rho = 0$ on "PDUAL" Goursat.* Middle*: ℓ_∞-error for estimator $\hat{H}^{cnc}$ and estimator $\hat{H}^{nc}$ with varying radii ρ on "PDUAL" Goursat.* Right*: ℓ_2-error for estimator $\hat{H}^{cnc}$ and estimator $\hat{H}^{nc}$ with varying radii ρ on "PRIMAL" Goursat.*

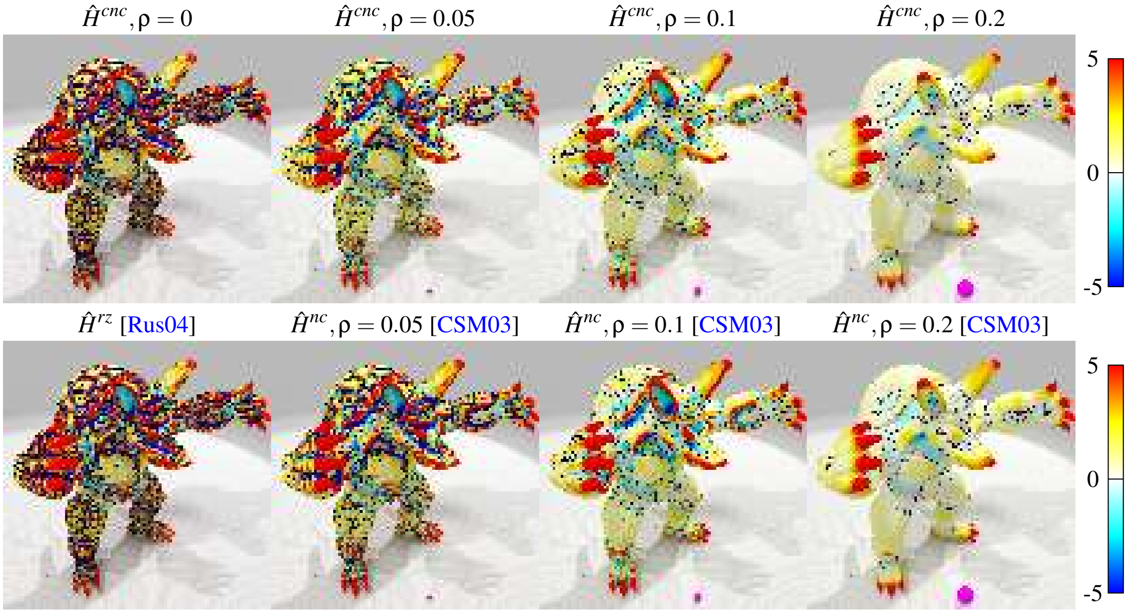

$\hat{H}^{cnc}, \rho = 0$
$\hat{H}^{cnc}, \rho = 0.05$
$\hat{H}^{cnc}, \rho = 0.1$
$\hat{H}^{cnc}, \rho = 0.2$
5
0
-5
$\hat{H}^{rz}$ [Rus04]
$\hat{H}^{nc}, \rho = 0.05$ [CSM03]
$\hat{H}^{nc}, \rho = 0.1$ [CSM03]
$\hat{H}^{nc}, \rho = 0.2$ [CSM03]
5
0
-5

Figure 7: *Mean curvature estimation on Armadillo dataset. In magenta, a ball of the same radius as the measuring ball.*

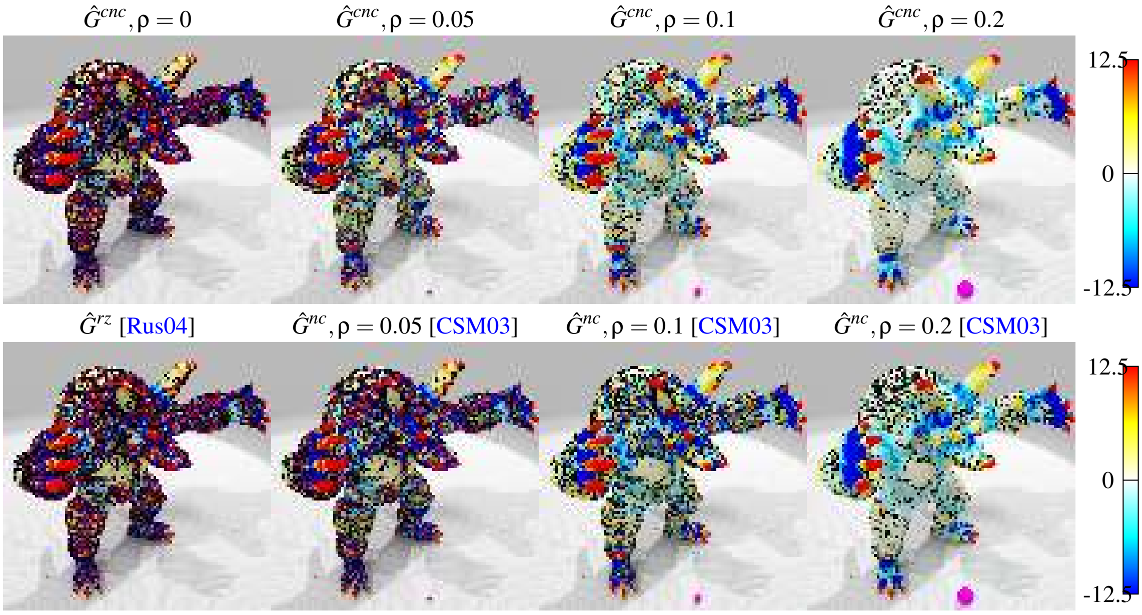

$\hat{G}^{cnc}, \rho = 0$
$\hat{G}^{cnc}, \rho = 0.05$
$\hat{G}^{cnc}, \rho = 0.1$
$\hat{G}^{cnc}, \rho = 0.2$
12.5
0
-12.5
$\hat{G}^{rz}$ [Rus04]
$\hat{G}^{nc}, \rho = 0.05$ [CSM03]
$\hat{G}^{nc}, \rho = 0.1$ [CSM03]
$\hat{G}^{nc}, \rho = 0.2$ [CSM03]
12.5
0
-12.5

Figure 8: *Gaussian curvature estimation on Armadillo dataset. In magenta, a ball of the same radius as the measuring ball.*

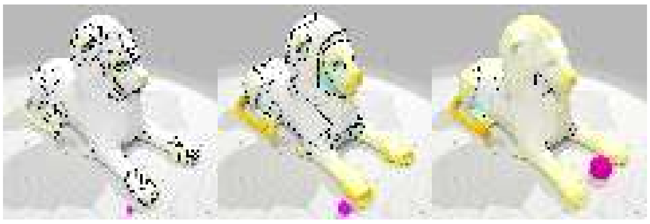


Figure 9: *It is difficult to find a correct measuring ball radius ρ for normal cycle estimator $\hat{H}^{nc}$ [CSM03], to be compared with $\hat{H}^{cnc}$ result on Figure 10.*

sian curvature computation is shown on Figure 17 for the “Lion” dataset, which has a very coarse geometry. Tiny details corresponding to the normal map are naturally extracted with our framework.

Timings and implementation. Computing local measures on triangles is linear with the number of triangles and takes less than 200ms for 1 million faces, while computing measures within a ball radius ρ is linear with the number of faces within the ball. We provide an interactive application to test our method, as well as its source code at https://github.com/dcoeurjo/CorrectedNormalCurrent.

6. Conclusion

We have presented a new definition of curvatures for polygonal meshes, based on a generalization of the normal cycle and with easy to compute closed form formulas. The user can prescribe any normal vectors at vertices to get a very close approximation of the curvatures of the smooth surface in the Grassmaniann that has these interpolated normal vectors. We have shown a stability property for these curvature measures with respect to Hausdorff and normal perturbations. Numerically our method is at least as good as the state-of-the-art on clean data and outperforms other approaches on noisy mesh or mesh with bad geometric normals. Our estimators include an intrinsic scale parameter (measure radius ρ), which builds a smooth curvature scale space: measures are continuous with respect to $\rho \in [0, +\infty[$.

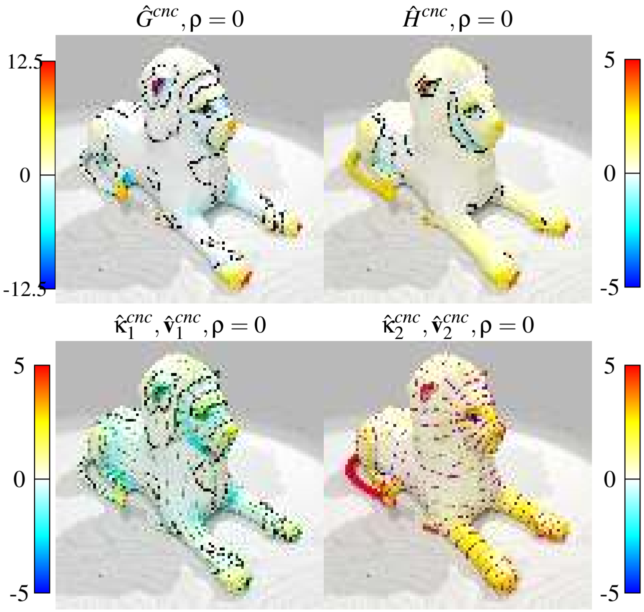

$\hat{G}^{cnc}, \rho = 0$
$\hat{H}^{cnc}, \rho = 0$
12.5
0
-12.5
5
0
-5
$\hat{\kappa}_1^{cnc}, \hat{\mathbf{v}}_1^{cnc}, \rho = 0$
$\hat{\kappa}_2^{cnc}, \hat{\mathbf{v}}_2^{cnc}, \rho = 0$
5
0
-5
5
0
-5

Figure 10: *Gaussian, mean and principal curvatures and directions with our method on Lion dataset.*

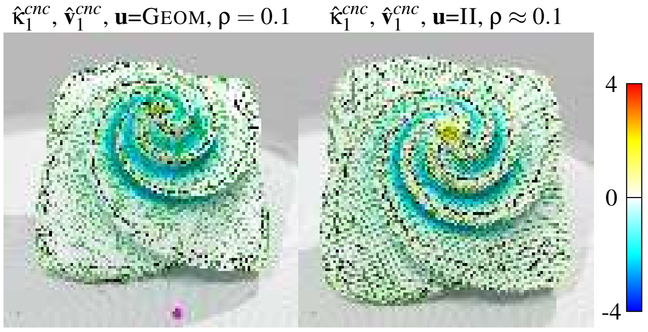

$\hat{\kappa}_1^{cnc}$, $\hat{\mathbf{v}}_1^{cnc}$, **u**=GEOM, $\rho = 0.1$ $\hat{\kappa}_1^{cnc}$, $\hat{\mathbf{v}}_1^{cnc}$, **u**=II, $\rho \approx 0.1$

Figure 11: *First principal curvatures and directions on polyhedral mesh "Octaflower" and its digitization.*

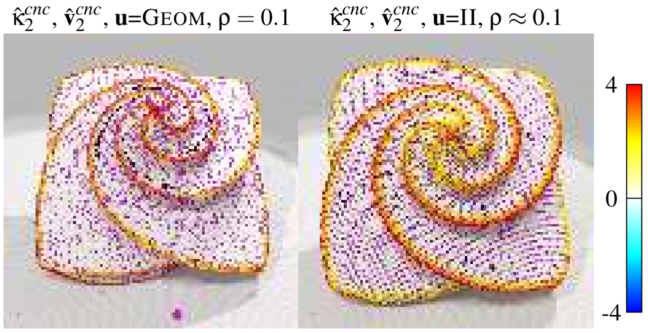

$\hat{\kappa}_2^{cnc}$, $\hat{\mathbf{v}}_2^{cnc}$, **u**=GEOM, $\rho = 0.1$ $\hat{\kappa}_2^{cnc}$, $\hat{\mathbf{v}}_2^{cnc}$, **u**=II, $\rho \approx 0.1$

Figure 12: *Second principal curvatures and directions on polyhedral mesh "Octaflower" and its digitization.*

In this paper, we have demonstrated the strength and the numerical efficiency of the new curvature measures when normal vectors are linearly interpolated within triangles. The general framework of corrected normal current is more versatile and would support other interpolating schemes (e.g. bilinear interpolation for quad mesh, splines) and could also handle piecewise smooth schemes (with discontinuity at edges). These specific challenging cases would lead to more complex formulas for the measure computation, but may be interesting for geometry processing applications.

For further studies, we wish to focus on the pointwise convergence of our curvature estimates. Such results can be obtained for digital surface with an appropriate corrected normal vector field [LRT19], but for polygonal surfaces the optimal radius remains to be found, and may depend on other parameters than the maximum edge length.

References

[Alm66] ALMGREN F. J.: *Plateau's problem: an invitation to varifold geometry*, vol. 13. American Mathematical Soc., 1966. 2, 3

[BHKB20] BUNGE A., HERHOLZ P., KAZHDAN M., BOTSCH M.: Polygon Laplacian Made Simple. *Computer Graphics Forum* (2020). doi:10.1111/cgf.13931. 3

[BLM17] BUET B., LEONARDI G. P., MASNOU S.: A varifold approach to surface approximation. *Archive for Rational Mechanics and Analysis 226*, 2 (2017), 639–694. 3

[BLM18] BUET B., LEONARDI G. P., MASNOU S.: Discretization and approximation of surfaces using varifolds. *Geometric Flows 3*, 1 (2018), 28–56. 3

[CCSLT09] CHAZAL F., COHEN-STEINER D., LIEUTIER A., THIBERT B.: Stability of curvature measures. *Computer Graphics Forum 28*, 5 (2009), 1485–1496. 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. 3, 7, 8

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

[CSM03] COHEN-STEINER D., MORVAN J.-M.: Restricted delaunay triangulations and normal cycle. In *Proceedings of the nineteenth annual symposium on Computational geometry* (2003), pp. 312–321. 2, 6, 7, 9, 10

[CSM06] COHEN-STEINER D., MORVAN J.-M.: Second fundamental measure of geometric sets and local approximation of curvatures. *Journal of Differential Geometry 74*, 3 (2006), 363–394. 2, 4, 13

[Fed59] FEDERER H.: Curvature measures. *Transactions of the American Mathematical Society 93*, 3 (1959), 418–491. 2, 5, 6

[Fu93] FU J. H. G.: Convergence of curvatures in secant approximations. *Journal of Differential Geometry 37*, 1 (1993), 177–190. 2

[Fu94] FU J. H.: Curvature measures of subanalytic sets. *American Journal of Mathematics* (1994), 819–880. 2

[GI04] GOLDFEATHER J., INTERRANTE V.: A novel cubic-order algorithm for approximating principal direction vectors. *ACM Trans. Graph. 23*, 1 (Jan. 2004), 45–63. 2

[HP11] HILDEBRANDT K., POLTHIER K.: Generalized shape operators on polyhedral surfaces. *Comput. Aided Geom. Des. 28*, 5 (June 2011), 321–343. 2, 7

[HPW06] HILDEBRANDT K., POLTHIER K., WARDETZKY M.: On the convergence of metric and geometric properties of polyhedral surfaces. *Geometriae Dedicata 123*, 1 (2006), 89–112. 2, 7

[KSNS07] KALOGERAKIS E., SIMARI P., NOWROUZEZAHRAI D., SINGH K.: Robust statistical estimation of curvature on discretized surfaces. In *Symposium on Geometry Processing* (2007), vol. 13, pp. 110–114. 2

[LRT19] LACHAUD J.-O., ROMON P., THIBERT B.: Corrected curvature measures. Working paper or preprint. https://hal.archives-ouvertes.fr/hal-02193774, July 2019. 2, 3, 4, 5, 6, 8, 11

[LT16] LACHAUD J.-O., THIBERT B.: Properties of Gauss digitized sets and digital surface integration. *Journal of Mathematical Imaging and Vision 54*, 2 (Feb. 2016), 162–180. 7

[Max99] MAX N.: Weights for computing vertex normals from facet normals. *Journal of graphics tools 4*, 2 (1999), 1–6. 7

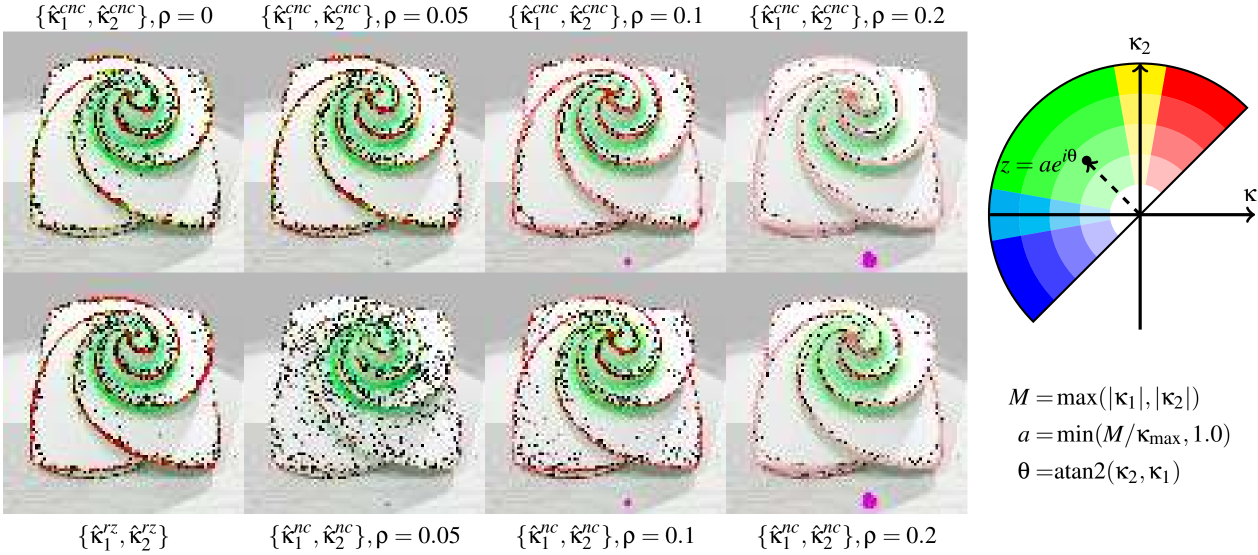
[MOG11] MÉRIGOT Q., OVSJANIKOV M., GUIBAS L.: Voronoi-based curvature and feature estimation from point clouds. *Visualization and Computer Graphics, IEEE Transactions on 17*, 6 (2011), 743–756. 2

[Mor08] MORVAN J.-M.: *Generalized curvatures*. Springer, 2008. 2, 4

[PWY*07] POTTMANN H., WALLNER J., YANG Y.-L., LAI Y.-K., HU S.-M.: Principal curvatures from the integral invariant viewpoint. *Computer Aided Geometric Design 24*, 8-9 (2007), 428–442. 2, 3

[Rus04] RUSINKIEWICZ S.: Estimating curvatures and their derivatives on triangle meshes. In *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004.* (2004), IEEE, pp. 486–493. 2, 7, 9, 10

[VVP*16] VÁŠA L., VANĚČEK P., PRANTL M., SKORKOVSKÁ V., MARTÍNEK P., KOLINGEROVÁ I.: Mesh Statistics for Robust Curvature Estimation. *Computer Graphics Forum* (2016). 2, 3, 7


$\{\hat{\kappa}_1^{cnc}, \hat{\kappa}_2^{cnc}\}, \rho = 0$
$\{\hat{\kappa}_1^{cnc}, \hat{\kappa}_2^{cnc}\}, \rho = 0.05$
$\{\hat{\kappa}_1^{cnc}, \hat{\kappa}_2^{cnc}\}, \rho = 0.1$
$\{\hat{\kappa}_1^{cnc}, \hat{\kappa}_2^{cnc}\}, \rho = 0.2$
κ_2
$z = ae^{i\theta}$
κ_1
$\{\hat{\kappa}_1^{rz}, \hat{\kappa}_2^{rz}\}$
$\{\hat{\kappa}_1^{nc}, \hat{\kappa}_2^{nc}\}, \rho = 0.05$
$\{\hat{\kappa}_1^{nc}, \hat{\kappa}_2^{nc}\}, \rho = 0.1$
$\{\hat{\kappa}_1^{nc}, \hat{\kappa}_2^{nc}\}, \rho = 0.2$

$$M = \max(|\kappa_1|, |\kappa_2|)$$
$$a = \min(M/\kappa_{\max}, 1.0)$$
$$\theta = \mathrm{atan2}(\kappa_2, \kappa_1)$$

Figure 13: *Extraction of convex (red), concave (blue), hyperbolic (green), concave/convex cylindric (cyan/yellow) and flat parts (white) on "Octaflower" dataset. Colors are computed as explained by the right diagram, with $\kappa_{\max} = 4$, and $\kappa_1 \leq \kappa_2$.*

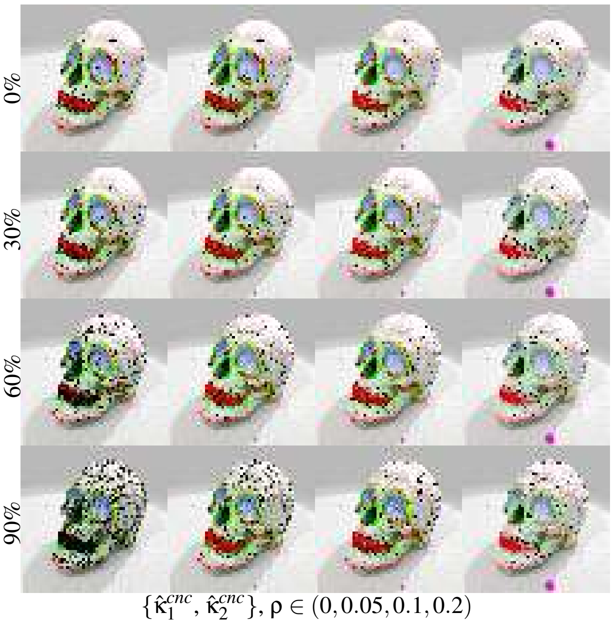

0%
30%
60%
90%
$\{\hat{\kappa}_1^{cnc}, \hat{\kappa}_2^{cnc}\}, \rho \in (0, 0.05, 0.1, 0.2)$

Figure 14: *Stability of convex (red), concave (blue), hyperbolic (green), concave/convex cylindric (cyan/yellow) and flat parts (white) on "Skull" dataset with increasing noise (0%, 30%, 60%, 90%) using $\hat{\kappa}_1^{cnc}$ and $\hat{\kappa}_2^{cnc}$. (See explanation of colors on Figure 13.) Corrected field **u** is obtained from geometric normals **n** by four averaging passes.*

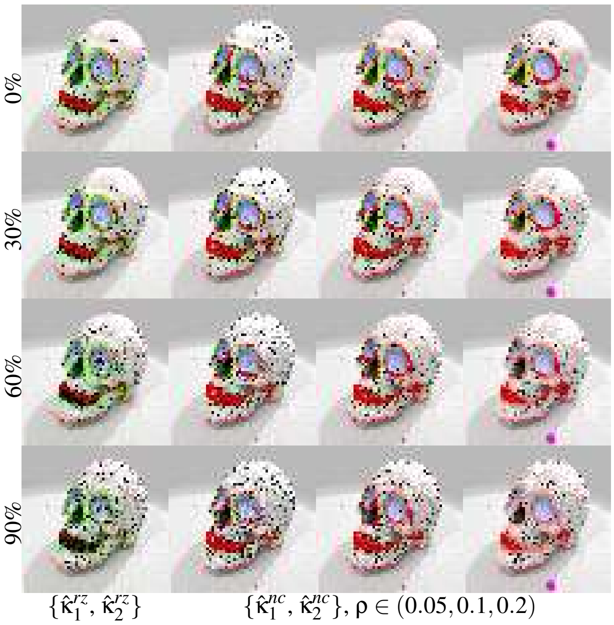

0%
30%
60%
90%
$\{\hat{\kappa}_1^{rz}, \hat{\kappa}_2^{rz}\}$
$\{\hat{\kappa}_1^{nc}, \hat{\kappa}_2^{nc}\}, \rho \in (0.05, 0.1, 0.2)$

Figure 15: *Stability of convex (red), concave (blue), hyperbolic (green), concave/convex cylindric (cyan/yellow) and flat parts (white) on "Skull" dataset with increasing noise (0%, 30%, 60%, 90%) using $\hat{\kappa}_1^{rz}$ and $\hat{\kappa}_2^{rz}$ or $\hat{\kappa}_1^{nc}$ and $\hat{\kappa}_2^{nc}$. (See explanation of colors on Figure 13.) Corrected field **u** is obtained from geometric normals **n** by four averaging passes.*

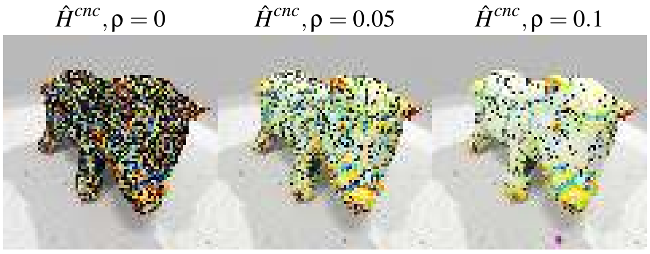

$\hat{H}^{cnc}, \rho = 0$
$\hat{H}^{cnc}, \rho = 0.05$
$\hat{H}^{cnc}, \rho = 0.1$

Figure 16: *Mean curvature estimation on LIDAR data according to measure radius* ρ.

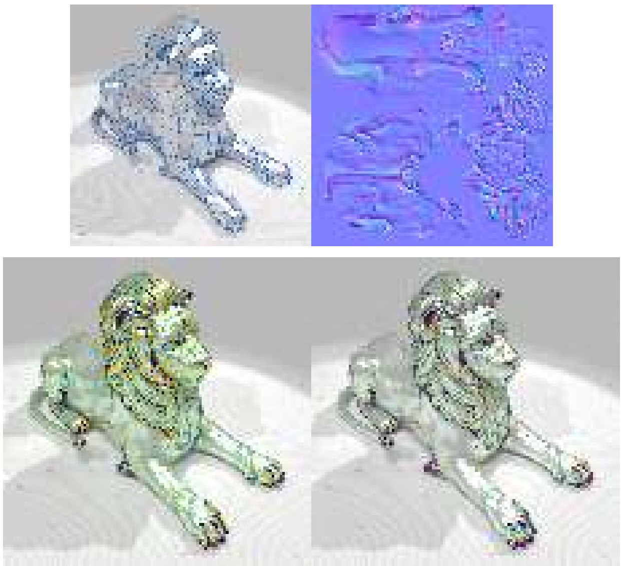


Figure 17: *Mean (left) and Gaussian (right) curvatures mapped onto "Lion" dataset, using* $\hat{H}^{cnc}$ *and* $\hat{G}^{cnc}$ *with corrected vector field* $\mathbf{u}$ *given by its normal map (first row).*

[Win82] WINTGEN P.: Normal cycle and integral curvature for polyhedra in riemannian manifolds. *Journal of Differential geometry 21* (1982). 2, 4

Appendix A: Computation of anisotropic measure on a triangle

The computation of the anisotropic measure (a.k.a. second fundamental measure of [CSM06]) on a triangle τ (last relation of Property 1) can be computed in a similar manner as the other measures. For given vectors $\mathbf{X}, \mathbf{Y}$, we use the pullback $\Gamma_{\mathbf{u}}^{*}\omega^{(\mathbf{X},\mathbf{Y})}$ as follows:

$$
\begin{aligned}
\mu^{\mathbf{X},\mathbf{Y}}(\tau) &= \int_{\tau} \Gamma_{\mathbf{u}}^{*}\omega^{(\mathbf{X},\mathbf{Y})} \\
&= \int_{\Delta} (\mathbf{u} \times \mathbf{X}, 0)^{\flat} \wedge (0, \mathbf{Y})^{\flat} \left(\frac{\partial \Gamma_{\mathbf{u}}}{\partial s}, \frac{\partial \Gamma_{\mathbf{u}}}{\partial t} \right) \mathrm{d}s\,\mathrm{d}t \\
&= \int_{\Delta} \begin{vmatrix} \left\langle \mathbf{u} \times \mathbf{X} \mid \frac{\partial \mathbf{x}}{\partial s} \right\rangle & \left\langle \mathbf{u} \times \mathbf{X} \mid \frac{\partial \mathbf{x}}{\partial t} \right\rangle \\ \left\langle \mathbf{Y} \mid \frac{\partial \mathbf{u}}{\partial s} \right\rangle & \left\langle \mathbf{Y} \mid \frac{\partial \mathbf{u}}{\partial t} \right\rangle \end{vmatrix} \mathrm{d}s\,\mathrm{d}t \\
&= \int_{\Delta} \big(\langle \mathbf{X} \mid (\mathbf{x}_1 - \mathbf{x}_0) \times (\mathbf{u}_0 + s(\mathbf{u}_1 - \mathbf{u}_0) \\
&\qquad + t(\mathbf{u}_2 - \mathbf{u}_0)) \rangle \langle \mathbf{Y} \mid \mathbf{u}_2 - \mathbf{u}_0 \rangle \\
&\qquad - \langle \mathbf{X} \mid (\mathbf{x}_2 - \mathbf{x}_0) \times (\mathbf{u}_0 + s(\mathbf{u}_1 - \mathbf{u}_0) \\
&\qquad + t(\mathbf{u}_2 - \mathbf{u}_0)) \rangle \langle \mathbf{Y} \mid \mathbf{u}_1 - \mathbf{u}_0 \rangle \big) \mathrm{d}s\mathrm{d}t \\
&= \int_0^1 \int_0^{1-s} \Big(\langle \mathbf{X} \mid (\mathbf{x}_1 - \mathbf{x}_0) \times \mathbf{u}_0 \rangle \langle \mathbf{Y} \mid \mathbf{u}_2 - \mathbf{u}_0 \rangle \\
&\quad - \langle \mathbf{X} \mid (\mathbf{x}_2 - \mathbf{x}_0) \times \mathbf{u}_0 \rangle \langle \mathbf{Y} \mid \mathbf{u}_1 - \mathbf{u}_0 \rangle \\
&\quad + s\big(\langle \mathbf{X} \mid (\mathbf{x}_1 - \mathbf{x}_0) \times (\mathbf{u}_1 - \mathbf{u}_0) \rangle \langle \mathbf{Y} \mid \mathbf{u}_2 - \mathbf{u}_0 \rangle \\
&\qquad - \langle \mathbf{X} \mid (\mathbf{x}_2 - \mathbf{x}_0) \times (\mathbf{u}_1 - \mathbf{u}_0) \rangle \langle \mathbf{Y} \mid \mathbf{u}_1 - \mathbf{u}_0 \rangle \big) \\
&\quad + t\big(\langle \mathbf{X} \mid (\mathbf{x}_1 - \mathbf{x}_0) \times (\mathbf{u}_2 - \mathbf{u}_0) \rangle \langle \mathbf{Y} \mid \mathbf{u}_2 - \mathbf{u}_0 \rangle \\
&\qquad - \langle \mathbf{X} \mid (\mathbf{x}_2 - \mathbf{x}_0) \times (\mathbf{u}_2 - \mathbf{u}_0) \rangle \langle \mathbf{Y} \mid \mathbf{u}_1 - \mathbf{u}_0 \rangle \big) \Big) \mathrm{d}t\mathrm{d}s \\
&= \frac{1}{6} \Big(\langle \mathbf{X} \mid (\mathbf{x}_1 - \mathbf{x}_0) \times (\mathbf{u}_0 + \mathbf{u}_1 + \mathbf{u}_2) \rangle \langle \mathbf{Y} \mid \mathbf{u}_2 - \mathbf{u}_0 \rangle \\
&\quad - \langle \mathbf{X} \mid (\mathbf{x}_2 - \mathbf{x}_0) \times (\mathbf{u}_0 + \mathbf{u}_1 + \mathbf{u}_2 \rangle \langle \mathbf{Y} \mid \mathbf{u}_1 - \mathbf{u}_0 \rangle \Big) \\
&= \frac{1}{2} \langle \bar{\mathbf{u}} \mid \langle \mathbf{Y} \mid \mathbf{u}_2 - \mathbf{u}_0 \rangle \mathbf{X} \times (\mathbf{x}_1 - \mathbf{x}_0) \\
&\qquad\quad - \langle \mathbf{Y} \mid \mathbf{u}_1 - \mathbf{u}_0 \rangle \mathbf{X} \times (\mathbf{x}_2 - \mathbf{x}_0) \rangle .
\end{aligned}
$$

Appendix B: Closed form for quads with bilinear interpolation

Let $\mathbf{x}_{00}$, $\mathbf{x}_{10}$, $\mathbf{x}_{01}$, $\mathbf{x}_{11}$ be the coordinates of the vertices of a quad q and $\mathbf{u}_{00}$, $\mathbf{u}_{10}$, $\mathbf{u}_{01}$, $\mathbf{u}_{11}$ the corrected normals at these vertices. Positions and normals are linearly interpolated as

$$
\begin{aligned}
\mathbf{x}(s,t) &:= (1-s)(1-t)\mathbf{x}_{00} + s(1-t)\mathbf{x}_{10} + (1-s)t\mathbf{x}_{01} + st\mathbf{x}_{11}, \\
\mathbf{u}(s,t) &:= (1-s)(1-t)\mathbf{u}_{00} + s(1-t)\mathbf{u}_{10} + (1-s)t\mathbf{u}_{01} + st\mathbf{u}_{11}.
\end{aligned}
$$

for all (s,t) is the standard unit square. The quad may thus not be planar. Similar computations to the ones made for triangles give the corrected area measure $\mu^{(0)}$:

$$
\begin{aligned}
&\mu^{(0)}(q) := \\
&\quad \frac{1}{36} \big[\langle 4\mathbf{u}_{00} + 2\mathbf{u}_{10} + 2\mathbf{u}_{01} + \mathbf{u}_{11} \mid (\mathbf{x}_{10} - \mathbf{x}_{00}) \times (\mathbf{x}_{01} - \mathbf{x}_{00}) \rangle \\
&\qquad + \langle 2\mathbf{u}_{00} + 4\mathbf{u}_{10} + \mathbf{u}_{01} + 2\mathbf{u}_{11} \mid (\mathbf{x}_{10} - \mathbf{x}_{00}) \times (\mathbf{x}_{11} - \mathbf{x}_{10}) \rangle \\
&\qquad + \langle 2\mathbf{u}_{00} + \mathbf{u}_{10} + 4\mathbf{u}_{01} + 2\mathbf{u}_{11} \mid (\mathbf{x}_{11} - \mathbf{x}_{01}) \times (\mathbf{x}_{01} - \mathbf{x}_{00}) \rangle \\
&\qquad + \langle \mathbf{u}_{00} + 2\mathbf{u}_{10} + 2\mathbf{u}_{01} + 4\mathbf{u}_{11} \mid (\mathbf{x}_{11} - \mathbf{x}_{01}) \times (\mathbf{x}_{11} - \mathbf{x}_{10}) \rangle \big].
\end{aligned}
$$

The computation of the corrected mean curvature measure $\mu^{(1)}$ is a little bit more involved and leads to the following formula:

$$
\begin{aligned}
&\mu^{(1)}(q) := \\
&\quad \frac{1}{12} \big[\langle \mathbf{u}_{00} \mid 2(\mathbf{u}_{01} - \mathbf{u}_{10}) \times \mathbf{x}_{00} - (\mathbf{u}_{01} + \mathbf{u}_{11}) \times \mathbf{x}_{10} \\
&\qquad\qquad + (\mathbf{u}_{10} + \mathbf{u}_{11}) \times \mathbf{x}_{01} + (\mathbf{u}_{10} - \mathbf{u}_{01}) \times \mathbf{x}_{11} \rangle \\
&\qquad + \langle \mathbf{u}_{10} \mid 2(\mathbf{u}_{00} - \mathbf{u}_{11}) \times \mathbf{x}_{10} - (\mathbf{u}_{00} + \mathbf{u}_{01}) \times \mathbf{x}_{11} \\
&\qquad\qquad + (\mathbf{u}_{01} + \mathbf{u}_{11}) \times \mathbf{x}_{00} + (\mathbf{u}_{11} - \mathbf{u}_{00}) \times \mathbf{x}_{01} \rangle \\
&\qquad + \langle \mathbf{u}_{11} \mid 2(\mathbf{u}_{10} - \mathbf{u}_{01}) \times \mathbf{x}_{11} - (\mathbf{u}_{00} + \mathbf{u}_{10}) \times \mathbf{x}_{01} \\
&\qquad\qquad + (\mathbf{u}_{00} + \mathbf{u}_{01}) \times \mathbf{x}_{10} + (\mathbf{u}_{01} - \mathbf{u}_{10}) \times \mathbf{x}_{00} \rangle \\
&\qquad + \langle \mathbf{u}_{01} \mid 2(\mathbf{u}_{11} - \mathbf{u}_{00}) \times \mathbf{x}_{01} - (\mathbf{u}_{10} + \mathbf{u}_{11}) \times \mathbf{x}_{00} \\
&\qquad\qquad + (\mathbf{u}_{00} + \mathbf{u}_{10}) \times \mathbf{x}_{11} + (\mathbf{u}_{00} - \mathbf{u}_{11}) \times \mathbf{x}_{10} \rangle \big].
\end{aligned}
$$

The corrected Gaussian curvature measure $\mu^{(2)}$ on q is:

$$\mu^{(2)}(q) :=$$

$$\begin{aligned}\frac{1}{36}\Big[&\langle 4\mathbf{u}_{00}+2\mathbf{u}_{10}+2\mathbf{u}_{01}+\mathbf{u}_{11} \mid (\mathbf{u}_{10}-\mathbf{u}_{00})\times(\mathbf{u}_{01}-\mathbf{u}_{00})\rangle \\
&+\langle 2\mathbf{u}_{00}+4\mathbf{u}_{10}+\mathbf{u}_{01}+2\mathbf{u}_{11} \mid (\mathbf{u}_{10}-\mathbf{u}_{00})\times(\mathbf{u}_{11}-\mathbf{u}_{10})\rangle \\
&+\langle 2\mathbf{u}_{00}+\mathbf{u}_{10}+4\mathbf{u}_{01}+2\mathbf{u}_{11} \mid (\mathbf{u}_{11}-\mathbf{u}_{01})\times(\mathbf{u}_{01}-\mathbf{u}_{00})\rangle \\
&+\langle \mathbf{u}_{00}+2\mathbf{u}_{10}+2\mathbf{u}_{01}+4\mathbf{u}_{11} \mid (\mathbf{u}_{11}-\mathbf{u}_{01})\times(\mathbf{u}_{11}-\mathbf{u}_{10})\rangle\Big].\end{aligned}$$

Last the anisotropic measure $\mu^{(\mathbf{X},\mathbf{Y})}$ can be written as:

$$\begin{aligned}&\mu^{(\mathbf{X},\mathbf{Y})}(q) := \\
&\frac{1}{72}\Big[\langle\mathbf{Y}\mid\mathbf{u}_{00}\rangle(\langle\mathbf{u}_{00}\times\mathbf{X}\mid(-\mathbf{x}_{00}-11\mathbf{x}_{10}+13\mathbf{x}_{01}-\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{10}\times\mathbf{X}\mid(-5\mathbf{x}_{00}-7\mathbf{x}_{10}+11\mathbf{x}_{01}+\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{01}\times\mathbf{X}\mid(\mathbf{x}_{00}-7\mathbf{x}_{10}+11\mathbf{x}_{01}-5\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{11}\times\mathbf{X}\mid(-\mathbf{x}_{00}-5\mathbf{x}_{10}+7\mathbf{x}_{01}-\mathbf{x}_{11})\rangle) \\
&\quad+\langle\mathbf{Y}\mid\mathbf{u}_{10}\rangle(\langle\mathbf{u}_{00}\times\mathbf{X}\mid(13\mathbf{x}_{00}-\mathbf{x}_{10}-7\mathbf{x}_{01}-5\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{10}\times\mathbf{X}\mid(17\mathbf{x}_{00}-5\mathbf{x}_{10}-5\mathbf{x}_{01}-7\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{01}\times\mathbf{X}\mid(5\mathbf{x}_{00}+\mathbf{x}_{10}+\mathbf{x}_{01}-7\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{11}\times\mathbf{X}\mid(7\mathbf{x}_{00}-\mathbf{x}_{10}+5\mathbf{x}_{01}-11\mathbf{x}_{11})\rangle) \\
&\quad+\langle\mathbf{Y}\mid\mathbf{u}_{01}\rangle(\langle\mathbf{u}_{00}\times\mathbf{X}\mid(-11\mathbf{x}_{00}+5\mathbf{x}_{10}-\mathbf{x}_{01}+7\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{10}\times\mathbf{X}\mid(-7\mathbf{x}_{00}+\mathbf{x}_{10}+\mathbf{x}_{01}+5\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{01}\times\mathbf{X}\mid(-7\mathbf{x}_{00}-5\mathbf{x}_{10}-5\mathbf{x}_{01}+17\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{11}\times\mathbf{X}\mid(-5\mathbf{x}_{00}-7\mathbf{x}_{10}-\mathbf{x}_{01}+13\mathbf{x}_{11})\rangle) \\
&\quad+\langle\mathbf{Y}\mid\mathbf{u}_{11}\rangle(\langle\mathbf{u}_{00}\times\mathbf{X}\mid(-\mathbf{x}_{00}+7\mathbf{x}_{10}-5\mathbf{x}_{01}-\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{10}\times\mathbf{X}\mid(-5\mathbf{x}_{00}+11\mathbf{x}_{10}-7\mathbf{x}_{01}+\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{01}\times\mathbf{X}\mid(\mathbf{x}_{00}+11\mathbf{x}_{10}-7\mathbf{x}_{01}-5\mathbf{x}_{11})\rangle \\
&\qquad\qquad+\langle\mathbf{u}_{11}\times\mathbf{X}\mid(-\mathbf{x}_{00}+13\mathbf{x}_{10}-11\mathbf{x}_{01}-\mathbf{x}_{11})\rangle)\Big].\end{aligned}$$

Appendix C: Closed form for digital surfaces

We may also consider a digital surface where the corrected normal vector field $\mathbf{u}$ is known at vertices and linearly interpolated over each quad surfel (axis aligned unit square). The formulas above can be simplified since the geometry of quad surfels is simpler. Indeed let $(\mathbf{1}_1,\mathbf{1}_2,\mathbf{1}_3)$ be the canonical basis of $\mathbb{R}^3$ and (x,y,z) an even permutation of $(1,2,3)$. The corrected measures over a surfel f of naive geometric normal $\mathbf{1}_z$ with boundary vertices $\mathbf{x}_{00},\mathbf{x}_{10},\mathbf{x}_{01},\mathbf{x}_{11}$ such that $\mathbf{x}_{10}=\mathbf{x}_{00}+\mathbf{1}_x$, $\mathbf{x}_{01}=\mathbf{x}_{00}+\mathbf{1}_y$, $\mathbf{x}_{11}=\mathbf{x}_{00}+\mathbf{1}_x+\mathbf{1}_y$ have the following form

$$\begin{aligned}\mu^{(0)}(f) &= \frac{1}{4}\left[\mathbf{a}_z+\mathbf{b}_z+\mathbf{c}_z+\mathbf{d}_z\right], \\
\mu^{(1)}(f) &= \frac{1}{6}\left[(2\mathbf{b}_x+\mathbf{d}_x+2\mathbf{c}_y+\mathbf{d}_y)\mathbf{a}_z-(\mathbf{a}_x+2\mathbf{c}_x+\mathbf{a}_y+2\mathbf{b}_y)\mathbf{d}_z\right] \\
&\quad+\frac{1}{6}\left[(\mathbf{b}_x+2\mathbf{d}_x-2\mathbf{a}_y-\mathbf{b}_y)\mathbf{c}_z-(2\mathbf{a}_x+\mathbf{c}_x-\mathbf{c}_y-2\mathbf{d}_y)\mathbf{b}_z\right], \\
\mu^{(2)}(f) &= \frac{1}{4}\left[(\mathbf{c}_x-\mathbf{b}_x)\mathbf{a}_y+(\mathbf{a}_x+\mathbf{c}_x)\mathbf{b}_y-(\mathbf{a}_x+\mathbf{b}_x)\mathbf{c}_y\right]\mathbf{d}_z \\
&\quad-\frac{1}{4}\left[(\mathbf{b}_x+\mathbf{d}_x)\mathbf{a}_y-(\mathbf{a}_x-\mathbf{d}_x)\mathbf{b}_y-(\mathbf{a}_x+\mathbf{b}_x)\mathbf{d}_y)\right]\mathbf{c}_z \\
&\quad+\frac{1}{4}\left[(\mathbf{c}_x+\mathbf{d}_x)\mathbf{a}_y-(\mathbf{a}_x-\mathbf{d}_x)\mathbf{c}_y-(\mathbf{a}_x+\mathbf{c}_x)\mathbf{d}_y)\right]\mathbf{b}_z \\
&\quad-\frac{1}{4}\left[(\mathbf{c}_x+\mathbf{d}_x)\mathbf{b}_y-(\mathbf{b}_x+\mathbf{d}_x)\mathbf{c}_y+(\mathbf{c}_x-\mathbf{b}_x)\mathbf{d}_y)\right]\mathbf{a}_z,\end{aligned}$$

with the notations $\mathbf{a}:=\mathbf{u}_{00},\mathbf{b}:=\mathbf{u}_{10},\mathbf{c}:=\mathbf{u}_{01},\mathbf{d}:=\mathbf{u}_{11}$.