Numerical analysis of differential operators on raw point clouds

Julie J. Digne, Jean-Michel Morel

Numerische Mathematik manuscript No.
(will be inserted by the editor)

Numerical analysis of differential operators on raw point clouds

Julie Digne · Jean-Michel Morel

Accepted for publication - preprint version

Abstract 3D acquisition devices acquire object surfaces with growing accuracy by obtaining 3D point samples of the surface. This sampling depends on the geometry of the device and of the scanned object and is therefore very irregular. Many numerical schemes have been proposed for applying PDEs to regularly meshed 3D data. Nevertheless, for high precision applications it remains necessary to compute differential operators on raw point clouds prior to any meshing. Indeed differential operators such as the mean curvature or the principal curvatures provide crucial information for the orientation and meshing process itself.

This paper reviews a half dozen local schemes which have been proposed to compute discrete curvature-like shape indicators on raw point clouds. All of them will be analyzed mathematically in a unified framework by computing their asymptotic form when the size of the neighborhood tends to zero. They are given in terms of the principal curvatures or of higher order intrinsic differential operators which, in return, characterize the discrete operators. All considered local schemes are of two kinds: either they perform a polynomial local regression, or they compute directly local moments. But the polynomial regression of order 1 is demonstrated to play a special role, because its iterations yield a *scale space*. This analysis is completed with numerical experiments comparing the accuracies of these schemes. We demonstrate that this accuracy is enhanced for all operators by applying previously the scale space.

Keywords Point clouds · scale space · Moving Least Squares surfaces · curvature computation

J. Digne
LIRIS-Geomod, Université de Lyon, CNRS UMR5205 3 boulevard du 11 novembre 1918
F-69622 Villeurbanne

J.-M. Morel
CMLA, Ecole Normale Supérieure de Cachan
61 Avenue du Président Wilson,
F-94230 Cachan

Introduction

The output of laser scanners or any surface acquisition system is a set of points sampled with variable density on the surface. The scanners often deliver directly a *mesh*, i.e., a set of triangles linking point samples. But the basic raw information is an unorganized point cloud which can be locally sparse or over-cluttered. In this paper we focus on the mathematical analysis and processing of such raw irregularly sampled surfaces. Indeed, they contain the most accurate information, before it is altered or smoothed by any re-sampling and meshing. We shall interpret in terms of intrinsic differential operators (the curvatures) the most interesting and simple local surface estimators. Iterative surface regularization processes will also be analyzed and the *scale space* method used to compute reliably local differential features. In one word, the challenge is: how to compute intrinsic operators which ideally only depend on the underlying surface, not on its sampling? Surprisingly enough, we shall see that this is possible and that the reliability of such operators can be enforced and evaluated.

The field of numerical surface analysis has been widely studied over the fifteen past years, due to the development of computer graphics. Yet, most studies take as starting surface representation a mesh. Meshes are much easier to handle than raw point clouds, being already oriented, having usually a uniform or regularly varying sampling, and having a definite surface topology. On the contrary, raw data point sets are completely unstructured heaps of points, known only by their Euclidean coordinates. Nevertheless the construction of a mesh and the constitution of its topology involve, implicitly or explicitly, the computation of differential operators on the raw data.

The most popular mesh reconstruction methods from a raw point cloud define a signed function over $\mathbb{R}^3$ representing the distance to the object, and then extract the 0 level set which approximates the object surface. See (e.g.) [15], [22], [9], [27], [4], and the well established Poisson method [28], which solves a Poisson equation to build the indicator function of the solid object. These methods vary in the approach to compute the distance function, but all extract its zero-level set by using the marching cubes algorithm [35], [29]. In such meshing processes, the initial raw points are irremediably lost. This incurs into a loss of resolution and explains the relevance of processing directly an unstructured raw point cloud.

The reminder of this paper is divided as follows: section 1 reviews surface operators and surface motions previously defined on meshes and on point clouds, section 2 gives the necessary definitions and tools for raw surfaces and their underlying smooth model. Section 3 analyzes the first kind of local "differential operator" computable on a raw cloud: these are simply local order 2 moments, which will be shown to asymptotically compute functions of the surface principal curvatures. Section 4 analyzes and compares the surface motions given by the projection on simple regression surfaces: a plane and a degree 2 polynomial surface. Finally, section 5 shows comparative numerical experiments comparing the accuracies of the mesh free methods to compute local pseudo-differential operators, and also show the improvement brought by applying a *scale space* strategy based on the iteration on the cloud of a local linear surface regression.

1 Computing curvatures on sampled surface: state of the art

1.1 Curvatures computed on meshes

Reviews for curvature estimation on meshed surfaces can be found in [44] or [36]. Curvature tensor estimation methods were pioneered by Taubin [46] who presented a simple approximation for computing the directional curvature in any tangent direction. The curvature is computed in all incident edge directions and a covariance matrix of the edge directions weighted by their directional curvatures and the area of the two incident triangles is built. Eigenvectors and eigenvalues of this covariance matrix yield a simple expression of the principal curvatures and curvature directions.

Other curvature tensor estimation methods include [44] where the tensor is estimated by building a linear system binding the tensor coefficients. This linear system expresses the constraints that multiplying the tensor by an edge direction should give the difference of the edge's endpoints normals. The same method is applied to find the curvature derivatives. Normals are also used in [48] to give a piecewise linear curvature estimation (see also [49]).

To avoid the computation of derivatives with irregular samples, a new kind of integral curvature estimation method has been proposed in [52], [43] and [42]. The intersection of the surface with either a sphere or a ball centered at a vertex is analyzed: the covariance matrix of this domain is computed and eigenvalues are expressed in terms of principal curvatures. By increasing the neighborhood radius, the curvature estimate can be made multiscale. A very interesting feature of these methods is that they do not rely on high order derivatives and are therefore more stable.

Surface motions have also been studied as part of a mesh fairing process. A key method was introduced by Taubin in [47] who considered a discrete Laplacian for a mesh V with vertices $\mathbf{v}_i$, $\frac{\partial V}{\partial t} = \lambda \mathcal{L}(V)$, $\mathcal{L}$ being a discretization of the Laplacian $\mathcal{L}(\mathbf{v}_i) = \frac{1}{\text{card } N(\mathbf{v}_i)} \sum j \in N(\mathbf{v}_i)(\mathbf{v}_j - \mathbf{v}_i)$ where $N(\mathbf{v}_i)$ is the set of vertices linked by an edge to $\mathbf{v}_i$ (1-ring neighborhood). This formulation is widely used. For example, [16] uses a similar "umbrella" operator. [20] also computes the discrete Laplacian for all mesh vertices its eigenvectors and eigenvalues. By removing the smallest eigenvalues, a *fair* mesh (i.e. a denoised mesh) is obtained.

A well known formulation of the Laplace Beltrami operator is the famous cotangent formula [38],

$$\Delta \mathbf{v}_i = \frac{1}{2} \sum_{j \in \mathcal{N}\mathbf{v}_i} (cotan\alpha_{ij} + cotan\beta_{ij})$$

where $\mathbf{v}_i$ is a vertex of the mesh, $\mathcal{N}_1(\mathbf{v}_i)$ its one ring neighborhood, α_{ij} and β_{ij} are the angle opposite to edge $\mathbf{v}_i\mathbf{v}_j$ in the two triangles adjacent to $\mathbf{v}_i\mathbf{v}_j$. This has been used to compute the surface intrinsic equation. Another definition of the curvatures for triangulated surfaces, based on the theory of normal cycles, can be found in [14].

1.2 Curvature estimation and surface motions defined on point clouds

We now examine the rare approaches dealing directly with point clouds. [50] introduced a *scale space* decomposition method for point clouds. The method builds an adjacency graph from the input points (in order to compute easily the geodesics on the surface). The geodesics are used to compute a density normalization kernel that regularizes the density. The scale space operator is the operator that moves each point to the barycenter of all points weighted by the regularization kernel and the distance to the center point. Then the scale space method is used to select "scale-space extrema". At each scale the point motion is considered. Introducing a scalar function on the displacement norm, the authors claim a recovery of the characteristic scales of the surface (the introduced function is extremal at the characteristic scales).

Estimating curvatures often necessitates the computation of surface derivatives. Yet derivating a potentially noisy surface can generate unstable estimates. Instead, it was noticed that local integral quantities contain all the information of the differential operators. This idea was used for example in [37], with a method to compute curvatures and normals based on defining Voronoi covariance matrices. More practically, a serious effort was made for defining integral invariants for surfaces.

Our study will consider the simplest local integral invariants. The most famous principal integral invariants were defined as follows: call D the interior of a surface $\mathcal{M}$, then the area A_r of the intersection of D with a sphere of radius r is an invariant. The second invariant is the volume V_r of the intersection of a ball of radius r with D. Both invariants were proved to be related to the mean curvature ([24], [11]) so that:

$$V_r = \frac{2\pi}{3} r^3 - \frac{\pi H}{4} r^4 + O(r^5)$$

$$A_r = 2\pi r^2 - \pi H r^3.$$

Such invariants were used in [19] for surface registration, or for feature detection ([13], [12]). Nevertheless a serious numerical drawback of such integral invariants expansions is that the dominant term never contains the actual surface information. The dominant first term is actually completely independent of the surface locus. This makes the method impractical because the term of interest (here the mean curvature H) is obtained as the difference of two lower order terms. Yet, since V_r and A_r are not exact but approximate volumes and areas, H cannot actually be obtained accurately from such formulas. The methods we will analyze in this paper actually solve the problem by designing the local operator in such a way that the differential operator of interest is the dominant term in the asymptotic expansion.

In terms of mathematical analysis, the analysis which goes closest to the present one is due to Pottman et al. in [43] and [52]. These authors analyzed the asymptotic behavior of several integral invariants, particularly the moments of inertia of various local intrinsic neighborhoods. Yet, once again, the quantity to estimate is not contained in their dominant terms, thus making the obtained asymptotic formulas numerically impractical. For example Theorem 2 of [43] shows that the principal moments of inertia of the neighborhood defined as the intersection of D

with a ball of radius r have the Taylor expansion

$$M_r^1 = \frac{2\pi}{15}r^5 - \frac{\pi}{48}(3k_1 + k_2)r^6 + O(r^7)$$

$$M_r^2 = \frac{2\pi}{15}r^5 - \frac{\pi}{48}(3k_2 + k_1)r^6 + O(r^7)$$

$$M_r^3 = \frac{19\pi}{480}r^5 - \frac{9\pi}{512}(k_2 + k_1)r^6 + O(r^7)$$

where k_1 and k_2 are the principal curvatures of the surface at the considered position. The authors then bypass the difficulty of not having the estimates in the dominant term by taking the difference $M_r^2 - M_r^1 = \frac{\pi}{24}(k_1 - k_2)r^6 + O(r^7)$. Yet this only yields the square of the principal curvature difference.

A more practical result was proved in [43] in theorem 6 : the barycenter of the surface patch (intersection of a ball of radius r with the surface $\mathcal{M}$) is proved to have coordinates $(0, 0, \frac{k_1+k_2}{8}r^2) + O(r^3)$. In this expression the signs are not lost and the curvature is indeed the dominant term of the expansion. For the sake of completeness the proof of this result will be recalled in Lemma 2.

In [45], the proposed framework for curvature estimation at a particular point is based on a set of curves representing the local neighborhood of the point under consideration.

For each pair $(\mathbf{p}_i, \mathbf{p}_j)$ of neighbors of $\mathbf{p}$, the set of triplets $(\mathbf{p}_i, \mathbf{p}, \mathbf{p}_j)$ is built. Each of those triplets can be used to define a parametric space curve $\mathbf{p}(t)$ by quadratic polynomial interpolation with $\mathbf{p}(0) = \mathbf{p}_i$, $\mathbf{p}(1) = \mathbf{p}_j$ and $\mathbf{p}(t) = \mathbf{p}$ where $t = \frac{|\mathbf{p}-\mathbf{p}_i|}{|\mathbf{p}-\mathbf{p}_i|+|\mathbf{p}_j-\mathbf{p}|}$. This allows for the approximation of maximum and minimum curvature values as the minimum and maximum normal curvature values for all possible point triplets. This method can be used either on meshes or point clouds.

In [25], the authors proposed a statistical estimation of the curvature of point sampled surfaces based on M-estimators[1]. The position difference vector $\Delta\mathbf{p}$ and normal difference vector $\Delta\mathbf{n}$ are used to define a linear system yielding a first estimate of the curvature tensor. Then residuals are computed and used to weigh the samples and the objective function is minimized by iterative reweighing of point samples. This yields the final curvature tensor estimate.

Finally in [6], an algorithm to compute the Laplacian of a function defined on point clouds in $\mathbb{R}^d$ was proposed along with convergence proofs. Yet the model is not tested on real surfaces. Neighborhood covariances being used already for normal estimation, the idea to express fundamental forms as covariances matrices was introduced. The next section reviews the covariance techniques considered in the literature.

1.3 Curvature estimation using covariance techniques

There are few covariance approaches and they have seldom been analyzed mathematically yet, (with the notable exception of [43] and [52] which will be detailed in this section). Nevertheless, covariance methods can be an elegant alternative to

[1] *M-estimation: robust fitting of a model by minimization of an objective function of the residuals with an Iterative Reweighed Least Squares (IRLS) scheme*

surface regression. Three papers [7], [33] and [40] use covariance matrices for the curvature estimation.

The first one [7] considers the neighbors $(\mathbf{p}_i)$ of a point $\mathbf{p}$. The second fundamental form analog is then defined as the covariance matrix of the vectors $\mathbf{pp_i}$ projected onto the tangent plane of the surface at $\mathbf{p}$. An analog of the Gauss map is also introduced: it is the covariance matrix of the neighbors unit normals projected onto the surface tangent plane at point $\mathbf{p}$. The eigenvectors are said to give the principal directions. In fact these two covariance matrices are inspired from [33]. Indeed, [33] first proposed to compute the covariance matrix of the normals at the neighbors of $\mathbf{p}$, and to extract the principal eigenvalues which correspond to the principal curvatures of the surface at $\mathbf{p}$. The last covariance method, introduced in [40] is not claimed to be explicitly linked to surface curvatures or fundamental forms, yet it is used to account for the surface geometric variations. Consider the covariance matrix of vectors $\mathbf{p}_i$ where is the barycenter of the neighborhood of $\mathbf{p}$. The *surface variation* is defined as the ratio of the least eigenvector over the sum of all eigenvectors of this covariance matrix. This quantity has the nice property that it is bounded between 0 (flat case) and 1/3 (isotropic case). All of these methods will be detailed and analyzed in section 3.

1.4 Moving Least Squares Surfaces

MLS (*Moving least square*) surfaces were introduced in [30] as follows. Given a data set of points $\{\mathbf{p}_i\}_i$ (possibly acquired by a 3D scanning device) and belonging to a smooth surface $\mathcal{M}$, the goal is to replace the points $\mathbf{p}$ defining $\mathcal{M}$ by a reduced set $R = \{r_i\}$ defining a so called MLS $\mathcal{M}'$ surface which approximates $\mathcal{M}$. The surface $\mathcal{M}$ is assumed to be a C^∞ 2-manifold. The authors fix a bounding error ε such that $d(\mathcal{M}, \mathcal{M}') < \varepsilon$, where d is the Hausdorff distance.

The projection of a point on the MLS surface is defined as follows: given a point $\mathbf{p}$, find a local reference domain (plane) for $\mathbf{p}$. The local regression plane H is obtained by minimizing a local weighted sum of square distances of the points $\mathbf{p}_i$ to the plane. The weights attached to $\mathbf{p}_i$ are defined as functions of the distance of $\mathbf{p}_i$ to the projection of $\mathbf{p}$ on plane H, rather than their distance to $\mathbf{p}$.

Assume Q is the projection of $\mathbf{p}$ onto H, then H is found by locally minimizing with respect to $\mathbf{n}$ and D the quadratic cost

$$\sum_{i=1}^{N} (< \mathbf{n}, \mathbf{p}_i > -D)^2 \theta(\|\mathbf{p}_i - Q\|)$$

where θ is a smooth, monotone decreasing positive function. We can set $Q = \mathbf{p} + t\mathbf{n}$ for some $t \in \mathbb{R}$, which leads to the minimization of

$$\sum_{i=1}^{N} (< \mathbf{n}, \mathbf{p}_i - \mathbf{p} - t\mathbf{n} >)^2 \theta(\|\mathbf{p}_i - \mathbf{p} - t\mathbf{n}\|).$$

The local reference domain is then given by an orthonormal coordinate system on H with origin Q. The reference domain for $\mathbf{p}$ is used to compute a local bivariate polynomial approximation to the surface in a neighborhood of $\mathbf{p}$. Let Q_i be the projection of $\mathbf{p}_i$ onto H, and $f_i =< \mathbf{n}, \mathbf{p}_i - Q_i >$. In this local coordinate system,

let (x_i, y_i) be the coordinates of Q_i on H. The coefficients of the polynomial are computed by minimizing the least square error $\sum_{i=1}^{N}(g(x_i, y_i) - f_i)^2\theta(\|\mathbf{p}_i - Q\|)$ The projection of $\mathbf{p}$ onto $\mathcal{M}$ is defined by the polynomial value at the origin, i.e. $Q + g(0,0)\mathbf{n} = \mathbf{p} + (t + g(0,0))\mathbf{n}$. Thus, given a point $\mathbf{p}$ and its neighborhood, its projection onto the MLS surface can be computed. The approximation power of MLS surfaces was evaluated in [31] and the first applications were introduced in [2], [5] and [32].

MLS surfaces are not only theoretically powerful; they also provide fine implementations for rendering, up-sampling or down-sampling point sets [3],[40]. Finer variants of MLS were subsequently proposed for a better preservation of sharp edges in surfaces defined by point clouds [39], [34], [18], [21] and [1].

The same framework was used to build a scale space for point clouds in [41]. The surface is evolved through a diffusion process $\frac{\partial \mathbf{p}}{\partial t} - \lambda \cdot \Delta\mathbf{p} = 0$, where $\mathbf{p}$ is a point of the surface, λ a diffusion parameter and $\Delta\mathbf{p} = H\mathbf{n}$ is the Laplace Beltrami Operator (H is the curvature and $\mathbf{n}$ the normal at point $\mathbf{p}$, this is the decomposition process). By remembering the set of displacements $D_i(\mathbf{p})$ of each point $\mathbf{p}$ we have a reconstruction operator. The choice of the Laplacian discretization is very important: a first possibility is to use the standard mesh Laplacian techniques [47] adapted for point clouds using the k-nearest neighbors instead of the one ring neighborhood. Another possibility is to use the weighted least squares projection [23], [26]: the surface is iteratively projected onto the plane defined by the weighted barycenter $\mathbf{o}$ and the normal estimated using the weighted neighborhood covariance matrix. The weights are a Gaussian function of the distance to $\mathbf{p}$, and the size of the Gaussian kernel is a parameter that controls the amount of smoothing. This projection process is in fact an order 1 projection motion (MLS1) that will be analyzed in the following sections.

To make the projection more efficient, [41] proposed to sub-sample the point cloud. This yields a scale space decomposition where at each level the surface is smoothed and sub-sampled. The scale space decomposition is then applied to the multi-scale freeform deformation and to the morphing problem, with satisfactory results.

The moving least squares (MLS) were used to estimate curvatures. For example, in [51], the authors use the MLS framework to build a closed form solution for curvature estimation. Indeed, surfaces implied by point clouds can be seen as the zero level set of an implicit function f whose gradient and Hessian Matrix are built. Finally, using formulas for the Gaussian and the mean curvature depending on the Hessian and gradient of f, those curvatures can be computed.

In [10], the problem of estimating differential quantities on point clouds is recast to that of fitting the local representation of the manifold by a jet. A jet is simply a truncated Taylor expansion. A n jet is a Taylor expansion truncated at order n. A jet of order n contains differential information up to the n-th order. In particular it is stated that a polynomial fitting of degree n estimates any $k^{th} order$ differential quantity to accuracy $O(h^{n-k+1})$. This implies that the coefficient of the first fundamental form and unit vector normal are estimated with $O(h^n)$ precision and the coefficients of the second fundamental form and shape operator are approximated with accuracy $O(h^{n-1})$, and so are the principal directions. In order to characterize curvature properties, the method resorts to the Weingarten map A of the surface, also called the shape operator, that is the tangent map of

the Gauss map. Recall that the first and second fundamental forms I, II and A satisfy $II(\mathbf{t},\mathbf{t}) = I(A(\mathbf{t}),\mathbf{t})$ for any vector $\mathbf{t}$ of the tangent space. Second order derivatives are computed by building the Weingarten map of the osculating jet whose eigenvalues are the principal curvatures. Note that the described methods can be used either with a mesh or with a point cloud. Jets are in fact very related to MLS surfaces. Indeed, to estimate differential quantities a polynomial fitting of degree n is done, which is exactly what MLS does. Therefore the analysis in section 4 giving the equation governing MLS1 and MLS2 motions are valid for the jets too.

Nevertheless, we shall prove that none of the above mentioned moment based methods for computing the curvatures without a surface regression gives back the signed curvatures. We shall also prove experimentally that in order to be stable, those integral estimates as well as the surface regression estimates require a large neighborhood, which leads to larger computation time. In terms of Signal-to-Noise Ratio, it will turn out to be better to consider a scale-space approach: applying the scale space iterations with a small neighborhood and then extracting the differential operator analogue.

This section has reviewed the main methods aiming at estimating locally the surface shape, thus implicitly computing local equivalents of the infinitesimal curvature tensor. We have seen that two sorts of methods, logically, dominate: the polynomial regressions on one side, and the local moments on the other. (It is actually difficult to imagine other kinds of local methods on a raw point set). These kinds have very different techniques, but we shall be able to compare them in two unifying frameworks. We shall first give their asymptotic equivalents, which are functions of the surface principal curvatures. Then we shall compare their reliability by a numerical set up in the experimental section.

In particular section 3 finds the form of the differential operators underlying the four mentioned discrete schemes based on local cloud point statistics, and proposing discrete analogues of the "second fundamental forms" or of the "principal curvatures". These discrete schemes have very simple and robust form, being based on the computation of local moments and eigenvalues of the point cloud. The next section 2 provides the tools to analyze numerically point cloud motions. The analysis is in spirit close to the image filter analysis performed in [8].

2 Tools for numerical analysis of point cloud surface motions

We always assume the existence of a smooth surface $\mathcal{M}$ supporting the point set. These surfaces are the boundaries of solid objects and can therefore be assumed to be locally Lipschitz graphs. However, for a mathematical analysis of smoothing algorithms and curvature estimations on the surface, we shall always assume that the surface is a C^∞ embedded manifold, known from its samples denoted by $\mathcal{M}_S$. This is not a limitation, in the sense that any finite sample set can be anyway interpolated by an arbitrarily smooth surface. Let $\mathbf{p} = \mathbf{p}(x_\mathbf{p}, y_\mathbf{p}, z_\mathbf{p})$ be a point of the surface $\mathcal{M}$. At each non umbilical point $\mathbf{p}$, consider the principal curvatures k_1 and k_2 linked to the principal directions $\mathbf{t}_1$ and $\mathbf{t}_2$, with $k_1 > k_2$ where $\mathbf{t}_1$ and $\mathbf{t}_2$ are orthogonal vectors. (At umbilical points, any orthogonal pair $(\mathbf{t}_1, \mathbf{t}_2)$ can be taken.) Set $\mathbf{n} = \mathbf{t}_1 \times \mathbf{t}_2$ so that $(\mathbf{t}_1, \mathbf{t}_2, \mathbf{n})$ is an orthonormal basis. The quadruplet $(\mathbf{p}, \mathbf{t}_1, \mathbf{t}_2, \mathbf{n})$ is called the local intrinsic coordinate system. In this system we can

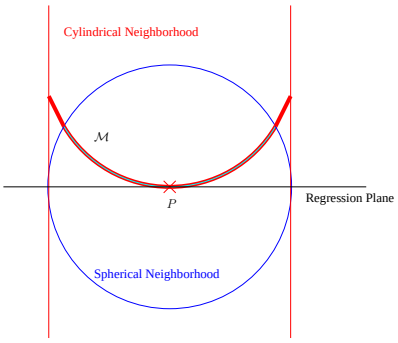


Fig. 1 Comparison between cylindrical and spherical neighborhoods

express locally the surface as a C^2 graph $z = f(x, y)$. By Taylor expansion[2],

$$z = f(x, y) = \frac{1}{2}(k_1 x^2 + k_2 y^2) + o(x^2 + y^2). \tag{1}$$

Notice that the sign of the pair (k_1, k_2) depends on the arbitrary surface orientation. Points where k_1 and k_2 have the same sign are called parabolic, and points where they have opposite signs are hyperbolic.

Consider two kinds of neighborhoods in $\mathcal{M}$ for $\mathbf{p}$ defined in the local intrinsic coordinate system $(\mathbf{p}, \mathbf{t}_1, \mathbf{t}_2, \mathbf{n})$:

- the *spherical* neighborhood $\mathcal{B}_r = B_r(\mathbf{p}) \cap \mathcal{M}$ is the set of all points $\mathbf{m}$ of $\mathcal{M}$ with coordinates (x, y, z) satisfying $(x - x_\mathbf{p})^2 + (y - y_\mathbf{p})^2 + (z - z_\mathbf{p})^2 < r^2$
- the *cylindrical neighborhood* $\mathcal{C}_r = C_r(\mathbf{p}) \cap \mathcal{M}$ is the set of all points $\mathbf{m}(x, y, z)$ on $\mathcal{M}$ such that $(x - x_\mathbf{p})^2 + (y - y_\mathbf{p})^2 < r^2$.

The spherical neighborhood in the sampled surface $\mathcal{M}_2$ is the only neighborhood to which there is a direct numerical access. It serves for defining all numerical schemes considered here. Nevertheless, for the forthcoming asymptotic numerical analysis, the cylindrical neighborhood will prove much handier than the spherical one. The next technical lemma justifies its use in theoretical calculations.

Lemma 1 *Integrating on $\mathcal{M}$ any function $f(x, y)$ such that $f(x, y) = O(r^n)$ on a cylindrical neighborhood $\mathcal{C}_r$ instead of a spherical neighborhood $\mathcal{B}_r$ introduces an $o(r^{n+4})$ error. More precisely:*

$$\int_{\mathcal{B}_r} f(x, y) d\mathbf{m} = \int_{x^2+y^2<r^2} f(x, y) dx dy + O(r^{4+n}). \tag{2}$$

Proof The surface area element of a point $\mathbf{m}(x, y, z(x, y))$ on the surface $\mathcal{M}$, expressed as a function of x, y, dx and dy is $d\mathbf{m}(x, y) = \sqrt{1 + z_x^2 + z_y^2} dx dy$. One has $z_x = k_1 x + O(r^2)$ and $z_y = k_2 y + O(r^2)$. Thus

$$d\mathbf{m}(x, y) = \sqrt{(1 + k_1^2 x^2 + k_2^2 y^2 + O(r^3))} dx dy$$

[2] We could use $z = f(x, y) = -\frac{1}{2}(k_1 x^2 + k_2 y^2) + o(x^2 + y^2)$ at the cost of changing the orientation and sign of k_1,k_2.

which yields

$$d\mathbf{m}(x,y) = (1+O(r^2))dxdy. \tag{3}$$

Using (3), the integrals we are interested in become

$$\int_{\mathcal{B}_r} f(x,y)d\mathbf{m} = (1+O(r^2)) \int_{\S^2+y^2+z^2<r^2,(x,y,z)\in\mathcal{M}} f(x,y)dxdy; \tag{4}$$

$$\int_{\mathcal{C}_r} f(x,y)d\mathbf{m} = (1+O(r^2)) \int_{x^2+y^2<r^2,\ (x,y,z)\in\mathcal{M}} f(x,y)dxdy. \tag{5}$$

The right hand forms are amenable to analytic computations. Consider polar coordinates (ρ,θ) such that $x=\rho\cos\theta$ and $y=\rho\sin\theta$ with $-r\leq\rho\leq r$ and $0\leq\theta\leq\pi$. Then for $\mathbf{m}(x,y,z)$ belonging to the surface $\mathcal{M}$, we have $z=\frac{1}{2}\rho^2(k_1\cos^2\theta+k_2\sin^2\theta)+O(r^3)$. Thus $z=\frac{1}{2}\rho^2k(\theta)+O(r^3)$, where $k(\theta)=k_1\cos^2\theta+k_2\sin^2\theta$. The condition that (x,y,z) belongs to the neighborhood $\mathcal{B}_r$ can therefore be rewritten as $\rho^2+z^2<r^2$, that is

$$\rho^2+\frac{1}{4}k(\theta)^2\rho^4 < r^2+O(r^5).$$

Computing the boundaries $\pm\rho(\theta)$ of this neighborhood yields $\rho(\theta)^2+\frac{1}{4}k(\theta)^2\rho(\theta)^4-r^2+O(r^5)=0$. Thus

$$\rho(\theta)^2 = \frac{-1+\sqrt{1+k(\theta)^2(r^2+O(r^5))}}{\frac{1}{2}k(\theta)^2}.$$

This yields $\rho(\theta)=r-\frac{1}{8}k(\theta)^2r^3+O(r^4)$. We shall use this estimate for the error term E appearing in

$$\begin{aligned}\int_{\mathcal{B}_r} f(x,y)dxdy &= \int_{[0,2\pi]}\int_{[0,\rho(\theta)]} f(x,y)\rho d\rho d\theta \\ &= \int_{[0,2\pi]}\int_{[0,r]} f(x,y)\rho d\rho d\theta - E \\ &= \int_{\mathcal{C}_r} f(x,y)dxdy - E,\end{aligned}$$

with $E =: \int_{[0,2\pi]}\int_{[\rho(\theta),r]} f(x,y)\rho d\rho d\theta$. Thus

$$|E| \leq 2\pi \sup_{x^2+y^2\leq r^2} |f(x,y)| \max(k_1^2,k_2^2)r^4 + O(r^5).$$

In particular if $f(x,y)=O(r^n)$, then $|E|\leq O(r^{4+n})$. Finally we have

$$\int_{\mathcal{B}_r} f(x,y)dxdy = \int_{\mathcal{C}_r} f(x,y)dxdy + O(r^{4+n}). \tag{6}$$

Combining (4), (5) and (6) yields (2).

This lemma will prove very useful for the rest of the paper and in particular in the next section where analysis are given for various curvature estimates.

A methodological objection to the asymptotic analysis Lemma 1, as well as all theorems in the remainder of this paper will assume that the surface is a uniform Lebesgue measure. Thus the theoretical analysis will be performed as though the surface were a very smooth object with dense uniform Lebesgue sampling. This is very far from reality, and could cast doubts on the pertinency of such a theoretical analysis. However, as will be explained later, the objection will prove invalid, in that the chosen local integral operators will always be robust to irregular sampling. For example the local area could definitely not be computed by a local sample density. In the same way the local barycenter of the existing sampled will be heavily biased by the irregular sampling and would have little to do with the actual local barycenter of the underlying surface. Nevertheless, the moments we shall consider are far more robust, in theory and in practice, to irregular sampling. This is the case for example for the normal when estimated as the normal to the local regression plane or, as we shall see, the mean curvature vector estimated as the projection of the sample on its regression plane. On the numerical side, however, it is recommended to compensate for the irregular sampling by an adequate sample reweighing in the computed local moments. This intrinsic density is simply approximated on discrete data by weighting each point by a weight inversely proportional to its initial density. More precisely, let p be a point and $\mathcal{N}_r(p) = \mathcal{M}_s \cap B_r(p)$. Each point q should ideally have a weight $0 \leq w(q) \leq 1$ such that $\sum_{q \in \mathcal{N}_r(p)} w(q) = 1$. This amounts to solve a huge linear system. For this reason, we shall be contented in the experimental section with ensuring $\sum_{q \in \mathcal{N}_r(p)} w(q) \simeq 1$ by taking $w(p) = \frac{1}{\sharp(B_p(r))}$, as proposed in [50].

3 Curvature estimates by covariance matrix methods

This section contains some of the main contributions of the present paper. It finds the form of the differential operators underlying four different discrete schemes based on local cloud point statistics, and proposing discrete analogues of the "second fundamental form matrix" or of the "principal curvatures". These discrete schemes have a very simple and robust form, being based on the computation of local moments and eigenvalues of the point cloud or of its normals. We shall see that all of the methods asymptotically compute nonlinear differential operators linked to the principal curvatures. Their principle is to replace the matrix of the second fundamental form by some symmetric matrix that can be deduced from the local statistics of the point cloud. We shall consider four matrices (2 or 3-dimensional)that are the simplest of such covariance matrices:

- 2D covariance matrix of the projections of $\overrightarrow{\mathbf{p}_i\mathbf{p}}$ on the tangent plane where $\mathbf{p}_i$ are the points of the neighborhood (section 3.1) ;
- 2D covariance matrix of the projections of the unit normals $\mathbf{n}(\mathbf{p}_i)$ on the tangent plane (section 3.2) ;
- 3D covariance matrix of the unit normals $\mathbf{n}(\mathbf{p}_i)$ (section 3.3) ;
- 3D centered covariance matrix of the $\overrightarrow{\mathbf{p}_i\mathbf{o}}$ where is the barycenter of the neighborhood (section 3.4).

3.1 A discrete "second fundamental form" [7]

Let$(\mathbf{p}_i)_{i\in 1\cdots N}$ be the set of neighbors of a point $\mathbf{p}$ with normal $\mathbf{n}$. This paper proposes to build the "second fundamental form matrix" as follows. (Although this covariance matrix is not, as we shall see, consistent with the second fundamental form, it is thus called in this paper, and actually has asymptotically, as we shall see, the principal directions as eigenvectors.) Let $s_i = (\mathbf{p}_i - \mathbf{p})^T \cdot \mathbf{n}$, let $\mathbf{t}_1$, $\mathbf{t}_2$ be two orthonormal vectors in the tangent plane of $\mathbf{p}$, and

$$\alpha_i = s_i \cdot \begin{pmatrix} (\mathbf{p}_i - \mathbf{p}) \cdot \mathbf{t}_1 \\ (\mathbf{p}_i - \mathbf{p}) \cdot \mathbf{t}_2 \end{pmatrix} = ((\mathbf{p}_i - \mathbf{p})^T \cdot \mathbf{n}) \cdot \begin{pmatrix} (\mathbf{p}_i - \mathbf{p}) \cdot \mathbf{t}_1 \\ (\mathbf{p}_i - \mathbf{p}) \cdot \mathbf{t}_2 \end{pmatrix}.$$

The α_i are the projections of the vectors $(\mathbf{p}_i - \mathbf{p})$ onto the tangent plane to $\mathbf{p}$, weighted by their distance to this plane. The "second fundamental form matrix" is the covariance of these vectors, namely

$$\Sigma_d = \sum_{i=1}^{N} (\alpha_i - \alpha_m) \cdot (\alpha_i - \alpha_m)^T \tag{7}$$

where $\alpha_m = \frac{1}{N}\sum_{i=1}^{N} \alpha_i$ and in Σ_d the d stands for "discrete". To compute the underlying differential operators, two assumptions will be made throughout this paper. The first one is that the surface sampling is uniform with respect to the area measure on the surface. The second one is that this sampling is dense enough, so that the averages taken on neighborhoods can be interpreted as integrals. Under this interpretation, we can reinterpret the sum in (7) as an integral on a cylindrical neighborhood of $\mathbf{p}$, assuming the data point set to be a locally smooth manifold. In the local intrinsic surface coordinate system at point $\mathbf{p}$, $(\mathbf{p}, \mathbf{t}_1, \mathbf{t}_2, \mathbf{n})$, the surface can be written as a graph $z = \frac{1}{2}(k_1 x^2 + k_2 y^2) + o(r^2)$. Thus the vectors α_i are replaced by a continuous vector $\alpha(x, y)$ defined by

$$\alpha(x,y) = \frac{1}{2}(k_1 x^2 + k_2 y^2) \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{2} \cdot \begin{pmatrix} k_1 x^3 + k_2 y^2 x \\ k_1 x^2 y + k_2 y^3 \end{pmatrix} + o(r^3). \tag{8}$$

Under the interpretation taken above the "second fundamental matrix" rewrites

$$\Sigma = \int_{\mathcal{B}_r} (\alpha(x,y) - \alpha_m) \cdot (\alpha(x,y) - \alpha_m)^T d\mathbf{m}(x,y) \tag{9}$$

where

$$\alpha_m = \frac{1}{\text{meas}(\mathcal{B}_r)} \int_{\mathcal{B}_r} \alpha(x,y) d\mathbf{m}(x,y). \tag{10}$$

The proposition made in [7] is to extract the surface principal curvatures and their corresponding directions at $\mathbf{p}$ from this covariance matrix, as its eigenvalues and eigenvectors. The next theorem checks if this works asymptotically in the continuous model.

Theorem 1 *The eigenvectors of the "second fundamental form matrix" Σ give the principal directions with error $o(r^8)$. But the eigenvalues of Σ are not the principal curvatures as they satisfy*

$$\lambda_1 = \frac{\pi r^8}{256}(5k_1^2 + 2k_1k_2 + k_2^2) + o(r^8) \text{ and } \lambda_2 = \frac{\pi r^8}{256}(k_1^2 + 2k_1k_2 + 5k_2^2) + o(r^8)$$

where k_1 and k_2 are the principal curvatures at $\mathbf{p}$

Proof In the continuous model α_m therefore is close to zero because the integrated function is odd on a symmetric domain. More precisely, using Lemma 1 in (10) and writing $\alpha_m = (\alpha_{mx}, \alpha_{my})$,

$$\alpha_{mx} = \frac{1}{2\pi r^2} \int_{x^2+y^2<r^2} (k_1 x^3 + k_2 y^2 x_i + o(r^5)) dxdy = o(r^3)$$

and similarly

$$\alpha_{my} = o(r^3).$$

By Lemma 1 again, the covariance matrix (9) satisfies $\Sigma = \int_{x^2+y^2<r^2} \alpha(x,y) \cdot \alpha(x,y)^T dxdy + o(r^8)$, and, using (8), we can calculate its four components as follows.

$$\Sigma_{11} = \frac{1}{4} \int_{x^2+y^2<r^2} (k_1 x^3 + k_2 y^2 x)^2 dxdy + o(r^8) = \frac{\pi r^8}{256}(5k_1^2 + 2k_1 k_2 + k_2^2) + o(r^8)$$

By exchanging the roles of k_1, k_2, and x, y respectively, we get

$$\Sigma_{22} = \frac{\pi r^8}{256}(k_1^2 + 2k_1 k_2 + 5k_2^2) + o(r^8).$$

Σ being a symmetric matrix, $\Sigma_{12} = \Sigma_{21}$ and the integrated function being odd,

$$\Sigma_{12} = \frac{1}{4} \int_{x^2+y^2<r^2} (k_1 x^3 + k_2 y^2 x)(k_1 x^2 y + k_2 y^3) dxdy + o(r^8) = o(r^8).$$

Thus, Σ is equivalent to a diagonal matrix whose principal directions are t_1 and t_2, which validates the theoretical requirements, t_1 and t_2 being the principal directions at point p. However, the corresponding eigenvalues are

$$\lambda_1 = \frac{\pi r^8}{256}(5k_1^2 + 2k_1 k_2 + k_2^2) + o(r^8)$$

$$\lambda_2 = \frac{\pi r^8}{256}(k_1^2 + 2k_1 k_2 + 5k_2^2) + o(r^8)$$

which are definitely different from $\lambda_1 = k_1$ and $\lambda_2 = k_2$. Only the absolute values of k_1 and k_2 can actually be deduced from Σ.

3.2 Another discrete "second fundamental form"

Another method was also introduced in [7] which, in a nutshell, computes the covariance matrix of the unit normal vectors projections onto the local tangent plane. By applying again the continuous asymptotic analysis of section 3.1, we shall see in Theorem 2 that this method actually computes discrete approximations of the squares of the principal curvatures. The discrete algorithm is as follows. Let $\mathcal{M}$ be a $\mathcal{C}^2$ surface and $\mathbf{p}$ be a point of $\mathcal{M}$. Let $(\mathbf{p}_i)_i$ be the neighbors of $\mathbf{p}$ in a ball neighborhood of radius r. Denote by $\mathbf{n}_i$ the normal at $\mathbf{p}_i$ and define v_i as the projection of $\mathbf{n}_i$ onto the tangent plane at $\mathbf{p}$, then the computed "curvatures" are

defined as the eigenvalues of the covariance matrix of the vectors v_i. The vector v_i being the projection of $\mathbf{n}_i$ onto the tangent plane, we have

$$v_i = \begin{pmatrix} \mathbf{n}_i \cdot \mathbf{t}_1 \\ \mathbf{n}_i \cdot \mathbf{t}_2 \end{pmatrix}.$$

Set $v_m = \frac{1}{N}\sum_{i=1}^N v_i$. Then this new discrete covariance matrix writes $\Sigma_d = \sum_{i=1}^N (v_i - v_m)\cdot(v_i - v_m)^T$. In the continuous framework, the local points on the surface have coordinates $\mathbf{m}(x,y) = (x, y, \frac{1}{2}(k_1x^2 + k_2y^2) + o(r^2))$ and the normal vector to this surface is $\frac{\partial \mathbf{m}}{\partial x}(x,y) \wedge \frac{\partial \mathbf{m}}{\partial y}(x,y) = (-k_1x, -k_2y, 1) + o(r)$. It follows that

$$v(x,y) = \frac{1}{\sqrt{1 + k_1^2x^2 + k_2^2y^2}} \begin{pmatrix} -k_1x \\ -k_2y \end{pmatrix} + o(r),$$

$$v_m = \frac{1}{\text{meas}(\mathcal{B}_r)} \int v(x,y) d\mathbf{m}(x,y),$$

and the continuous covariance matrix is

$$\Sigma := \int_{\mathcal{B}_r} (v(x,y) - v_m)\cdot(v(x,y) - v_m)^T.$$

Theorem 2 *The eigenvalues of the covariance matrix Σ of the vectors $v(x,y)$ in the spherical neighorhood $\mathcal{B}_r$ are*

$$\frac{k_1^2 r^4 \pi}{4} + o(r^4) \text{ and } \frac{k_2^2 r^4 \pi}{4} + o(r^4).$$

Proof Let us compute the mean v_m of $v(x,y)$ on the spherical neighborhood. By Lemma 1 the integral on a spherical neighborhood is asymptotically equivalent to the integral on a cylindrical neighborhood and more precisely, $(\pi r^2) v_m \cdot t_1 = o(r^3)$ and similarly

$$(\pi r^2) v_m \cdot t_2 = \int_{x^2+y^2<r^2} \frac{-k_2 y}{\sqrt{1 + k_1^2x^2 + k_2^2y^2}} dxdy + o(r^4) = o(r^3).$$

Thus the coefficients of Σ satisfy, again by Lemma 1,

$$\begin{aligned} \Sigma_{11} &= \int_{x^2+y^2<r^2} \frac{k_1^2x^2}{1 + k_1^2x^2 + k_2^2y^2} dxdy + o(r^4) = \int_{x^2+y^2<r^2} k_1^2x^2 + o(r^4) \\ &= \frac{k_1^2 r^4 \pi}{4} + o(r^4) \end{aligned}$$

Similarly, $\Sigma_{22} = \frac{k_2^2 r^4 \pi}{4} + o(r^4)$ and $\Sigma_{12} = \Sigma_{21} = o(r^4)$.

Thus Σ is asymptotically diagonal and its eigenvalues Σ_{11} and Σ_{22} are asymptotically obtained for the principal directions t_1 and t_2. Yet, these eigenvalues asymptotically give an approximation of each one of the squared principal curvatures, but not of their sign.

3.3 A third discrete "fundamental form"

The methods analyzed in sections 3.1 and 3.2 are akin to the original method introduced in [33]. Indeed, in [33] it was proposed to compute the covariance matrix of the normal vectors of the neighborhood (without projecting them in the local regression plane) and therefore get a 3×3 matrix instead of a 2×2 matrix. This is actually the simplest imaginable method and we shall see that it gives a result similar to section 3.2.

Theorem 3 *Let $\mathcal{M}$ be a $\mathcal{C}^2$ surface, let $\mathbf{p}$ be a point of $\mathcal{M}$. Then the three eigenvalues of the covariance matrix C of the unit normals in a neighborhood of radius r around $\mathbf{p}$ are asymptotically respectively equal to 1 and to the squares of the principal curvatures at $\mathbf{p}$.*

Proof A normal vector writes

$$N = \frac{1}{\sqrt{1 + k_1^2 x^2 + k_2^2 y^2}} \begin{pmatrix} -k_1 x \\ -k_2 y \\ 1 \end{pmatrix} + o(r).$$

As in the previous sections, we easily obtain by Lemma 1, $N_{mx} = o(r), N_{my} = o(r), N_{mz} = 1 + o(r)$. Thus again by Lemma 1,

$$C = \int_{x^2+y^2 \leq r^2} \frac{1}{1 + k_1^2 x^2 + k_2^2 y^2} \begin{pmatrix} k_1^2 x^2 & k_1 k_2 xy & k_1 x(1 - N_{mz}) \\ k_1 k_2 xy & k_2^2 y^2 & k_2 y(1 - N_{mz}) \\ k_1 x(1 - N_{mz}) & k_2 y(1 - N_{mz}) & (1 - N_{mz})^2 \end{pmatrix} dxdy + o(r^4)$$

and, by calculations exactly analogous to Section 3.2, $C_{11} = \frac{k_1^2 r^4 \pi}{4} + o(r^4)$, $C_{22} = \frac{k_2^2 r^4 \pi}{4} + o(r^4)$, $C_{12} = C_{21} = k_1 k_2 \int_{x,y} xydxdy = o(r^4)$, $C_{13} = C_{31} = C_{23} = C_{32} = C_{33} = o(r^4)$. Thus the eigenvalues are asymptotically equal to $\frac{k_1^2 r^4 \pi}{4}$ and $\frac{k_2^2 r^4 \pi}{4}$, which also gives back the squares of the principal curvatures of the surface, but again not their sign.

3.4 A fourth discrete fundamental form: the surface variation

We shall now analyze a last variant introduced in [40], the so called *surface variation*. It is again based on a local covariance analysis. Unlike the previous methods, the *surface variation* was not claimed to be a curvature estimate, but to be a measure of the neighborhood shape. This subsection establishes again a link between this discrete quantity and the principal curvatures of the surface.

Let $\mathbf{p}$ be a point with given neighborhood $\mathcal{B}_r$. Let $\mathbf{o}$ be the barycenter of the neighborhood. In $\mathbb{R}^3$, the coordinates are written with superscripts e.g. the coordinates of a point u are (u^1, u^2, u^3). Thus, for $i = 1, 2, 3$, $\mathbf{o}^i = \frac{1}{\operatorname{card} \mathcal{B}_r} \sum_{\mathbf{p}_k \in \mathcal{B}_r} \mathbf{p}_k^i$. The centered covariance matrix $\Sigma = (m_{ij})_{i,j=1,\cdots,3}$ is defined as $m_{ij} = \sum_{\mathbf{p}_k \in \mathcal{B}_r} (\mathbf{p}_k^i - \mathbf{o}^i) \cdot (\mathbf{p}_k^j - \mathbf{o}^j)$ for $i, j = 1, 2, 3$. Let $\lambda_0 \leq \lambda_1 \leq \lambda_2$ be the eigenvalues of Σ with corresponding eigenvectors v_0, v_1, v_2. For $k = 0, \cdots, 2$,

$$\lambda_k = \sum_{\mathbf{p}_i \in \mathcal{B}_r} \langle (\mathbf{p}_i - \mathbf{o}), v_k \rangle^2. \tag{11}$$

Each eigenvalue gives the variance of the point set in the direction of the corresponding eigenvector. Since v_1 and v_2 are the vectors that capture most variations, they define the PCA regression plane. The normal v_0 to this plane is the direction v minimizing $\sum_{\mathbf{p}_i \in \mathcal{B}_r} \langle (\mathbf{p}_i - \mathbf{o}), v\rangle^2$. [40] defines the surface variation by

$$\sigma = \frac{\lambda_0}{\lambda_0 + \lambda_1 + \lambda_2}. \tag{12}$$

This quantity measures the ratio of variance along the normal to the total variance. If the neighborhood is highly curved, its surface variation will be high and if the neighborhood is flat the surface variation will be small. This quantity has the property to be bounded between 0 (flat case) and 1/3 (isotropic distribution case).

Lemma 2 *[43] In the local intrinsic coordinate system, the barycenter of a neighborhood $\mathcal{B}_r$ of point $\mathbf{p}$ has coordinates $x_\mathbf{o} = o(r^2)$, $y_\mathbf{o} = o(r^2)$ and $z_\mathbf{o} = \frac{Hr^2}{4} + o(r^2)$, where $H = \frac{k_1+k_2}{2}$ is the mean curvature at $\mathbf{p}$.*

Proof We give the proof for the sake of completeness. By Lemma 1 applied to the numerator and denominator of the following fraction, we have

$$\begin{aligned}
z_\mathbf{o} = \frac{\int_{\mathcal{B}_r} z d\mathbf{m}}{\int_{\mathcal{B}_r} d\mathbf{m}} &= \frac{\int_{x^2+y^2<r^2} z(x,y)dxdy + O(r^5)}{\int_{x^2+y^2<r^2} dxdy + O(r^3)} \\
&= \frac{\int_{x^2+y^2<r^2} \left[\frac{1}{2}(k_1x^2 + k_2y^2) + o(x^2+y^2)\right] dxdy}{\int_{x^2+y^2<r^2} dxdy} + O(r^3) \\
&= \frac{1}{2\pi r^2} \int_{\rho=0}^{r} \int_{\theta=0}^{2\pi} \rho^2 (k_1 \cos^2\theta + k_2 \sin^2\theta)\rho d\rho d\theta + o(r^2) \\
&= \frac{r^2}{8\pi}(k_1\pi + k_2\pi) + o(r^2) = \frac{Hr^2}{4} + o(r^2).
\end{aligned}$$

A similar but simpler computation yields the estimates of $x_\mathbf{o}$ and $y_\mathbf{o}$.

Theorem 4 *In the local coordinate system the surface variation σ satisfies*

$$\sigma = \frac{r^2}{16}\left(\frac{k_1^2 + k_2^2}{2} - \frac{1}{3}k_1k_2\right) + o(r^2) \tag{13}$$

Proof We need to explain what the covariance eigenvalues stand for. Each eigenvector v_i and associated eigenvalue λ_i represent a principal direction and the variation along this principal direction,

$$\lambda_i = \int_{\mathbf{m} \in \mathcal{B}_r} \langle \mathbf{om}, v_i\rangle^2 d\mathbf{m}.$$

Since we have $\lambda_0 \leq \lambda_1 \leq \lambda_2$, we can see that λ_0 is associated to the direction with the least variation namely the normal direction to the surface $\mathbf{o}z$. Since the eigenvectors form an orthonormal basis, we have

$$\lambda_0 + \lambda_1 + \lambda_2 = \int_{\mathbf{m} \in \mathcal{C}_r} \langle \mathbf{om}, v_0\rangle^2 + \langle \mathbf{om}, v_1\rangle^2 + \langle \mathbf{om}, v_2\rangle^2 d\mathbf{m} = \int_{\mathbf{m} \in \mathcal{C}_r} \|\mathbf{om}\|^2 d\mathbf{m}$$

This yields $\lambda_0 + \lambda_1 + \lambda_2 = \int_{x^2+y^2<r^2} x^2 + y^2 + (z - z_\mathbf{o})^2 dxdy$ and

$$\lambda_0 + \lambda_1 + \lambda_2 = \frac{\pi r^4}{2} + \lambda_0 + o(r^6) \tag{14}$$

We first compute λ_0, applying again Lemma 1 to get back to the easy cylindrical neighborhood.

$$\begin{aligned}
\lambda_0 &= \int_{x^2+y^2\leq r^2} (z - z_\mathbf{o})^2 dxdy \\
&= \int_{x^2+y^2\leq r^2} z^2 dxdy + z_\mathbf{o}^2 \int_{x^2+y^2\leq r^2} dxdy - 2z_\mathbf{o} \int_{x^2+y^2\leq r^2} z dxdy \\
&= \int_{x^2+y^2\leq r^2} z^2 dxdy + \frac{H^2 r^4}{16} * \pi r^2 - 2\frac{Hr^2}{4}\frac{r^4}{4}\pi H + o(r^6) \\
&= \frac{1}{4}(k_1^2 \int_{x^2+y^2\leq r^2} x^4 dxdy + k_2^2 \int_{x^2+y^2\leq r^2} y^4 dxdy + 2k_1k_2 \int_{x^2+y^2\leq r^2} x^2y^2 dxdy \\
&- \frac{H^2 r^6}{16}\pi + o(r^6) \\
&= \frac{1}{4}\frac{r^6}{6}(\frac{3\pi}{4}(k_1^2 + k_2^2) + k_1k_2\frac{\pi}{2}) - \frac{H^2 r^6}{16}\pi + o(r^6)
\end{aligned}$$

where $H = \frac{k_1+k_2}{2}$ is the mean curvature. Thus

$$\lambda_0 = \frac{\pi r^6}{32}(\frac{k_1^2 + k_2^2}{2} - \frac{1}{3}k_1k_2) + o(r^6) \tag{15}$$

Using (14) and (15) we get

$$\sigma = \frac{\frac{r^2}{16}(\frac{k_1^2+k_2^2}{2} - \frac{1}{3}k_1k_2) + o(r^2)}{1 + \frac{r^2}{16}(\frac{k_1^2+k_2^2}{2} - \frac{1}{3}k_1k_2) + o(r^2)}$$

which finally yields:

$$\sigma = \frac{r^2}{16}\left(\frac{k_1^2 + k_2^2}{2} - \frac{1}{3}k_1k_2\right) + o(r^2)$$

The formula of the surface variation given by Theorem 4 indeed measures a sort of curvature. To interpret it we can notice that

- the surface variation is symmetric in k_1,k_2;
- in the case of a point lying on a sphere, $k_1 = k_2 = k$ so $\sigma_{sphere} = \frac{r^6}{24}k^2$;
- in the case of a saddle point $k_1 = k = -k_2$, $\sigma_{saddle} = \frac{r^6}{12}k^2$ so $\sigma_{sphere} < \sigma_{saddle}$;
- in the case of a cylinder $k_1 = k$, $k_2 = 0$, $\sigma_{cylinder} = \frac{r^6}{32}k^2$.

It follows from that the surface variation is not a discriminating enough information about the surface curvature. It is unable to discriminate very different local shapes.

4 Asymptotic behavior of MLS1 and MLS2

The simplest statistics that can be computed in a spherical neighborhood are the barycenter and the regression plane. Lemma 2 stated that sending each point onto the barycenter of its neighborhood approximates the mean curvature motion. The next simplest statistic is the regression plane, and the second next is the local degree 2 regression surface. The main tool of the scale space proposed in [17] is the projection of each surface point $\mathbf{p}$ on the local regression plane. This PCA regression plane is defined as the plane orthogonal to the least eigenvector of the centered local covariance matrix, and passing through the centroid of the neighborhood. The projection of $\mathbf{p}$ on this plane will be called $\mathbf{p}' = MLS1(\mathbf{p})$ where MLSn stands for *moving least square of degree* n. Indeed, this projection method is the simplest instance of the *moving least square* method by which each point of a surface is projected to a local degree n polynomial regression. (The local barycenter can actually be considered as an MLS of order 0, MLS0.) There is some particular interest in MLS1, because a recent meshing method uses it as the simplest reversible smoothing tool for point clouds [17]. On the other hand many cloud point processing methods involve some variant of the MLS2 method to smooth, interpolate, or sub-sample a point cloud. MLS1 and MLS2 are smoothing operators and therefore could be used as *scale spaces*, that is, as iterative smoothing operators. But, following [17] MLS1 indeed is a scale space. MLS2 is not, as illustrated in the experiments of Section 5. The theorems of this section clarify what happens with these local polynomial regressions by first recalling briefly why MLS1 implements a mean curvature motion, and second by showing that MLS2 is insensitive to first, second, and third order intrinsic derivatives, and has an order 4 difference to the original surface. The study reveals the fourth order intrinsic differential operator associated with MLS2.

4.1 The asymptotic behavior of MLS1

The next lemma compares the normal to the PCA regression plane with the normal to the surface, $\mathbf{n}$ at $\mathbf{p}$.

Lemma 3 *The normal* $\mathbf{v}$ *to the PCA regression plane in a spherical neighborhood* $\mathcal{B}_r$ *at* $\mathbf{p} \in \mathcal{M}$ *is equal to the surface normal at point* $\mathbf{p}$*, up to a negligible factor:* $\mathbf{v} = \mathbf{n} + O(r)$.

Proof The local PCA regression plane of point $\mathbf{p}$ is characterized as the plane passing through the barycenter of the neighborhood $\mathcal{B}_r$ and with normal $\mathbf{v}$ minimizing:

$$I(\mathbf{v}) = \int_{\mathcal{B}_r} |\langle \mathbf{v}, \mathbf{pp}' \rangle|^2 d\mathbf{p}' \text{ s.t. } \|\mathbf{v}\| = 1$$

Denoting by (v_x, v_y, v_z) the coordinates of $\mathbf{v}$,

$$I(\mathbf{v}) = \int_{\mathcal{B}_r} (v_x x + v_y y + v_z \frac{1}{2}(k_1 x^2 + k_2 y^2) + o(r^2))^2 dxdy.$$

Considering the particular value $\mathbf{v} = (0, 0, 1)$ shows that the minimal value I_{min} of $I(\mathbf{v})$ satisfies $I_{min} \leq O(r^6)$. In consequence the minimum (v_x, v_y, v_z) satisfies

$v_x \leq O(r)$ and $v_y \leq O(r)$. Thus, since $||\mathbf{v}|| = 1$, $v_z \geq 1 - O(r)$ and therefore $\mathbf{v} = \mathbf{n} + O(r)$.

Theorem 5 *Let T_r be the operator defined on the surface $\mathcal{M}$ transforming each point $\mathbf{p}$ into its projection $\mathbf{p}' = T_r(\mathbf{p})$ on the local regression plane. Then*

$$T_r(\mathbf{p}) - \mathbf{p} = \frac{Hr^2}{4}\mathbf{n} + o(r^2). \tag{16}$$

Proof By Lemma 2 the barycenter $\mathbf{o}$ of $\mathcal{B}_r$ has local coordinates $\overrightarrow{\mathbf{po}} = (o(r^2), o(r^2), \frac{Hr^2}{4} + o(r^2))$. On the other hand $\overrightarrow{\mathbf{pp'}}$ is proportional to the normal to the regression plane, $\mathbf{v}$. Thus by Lemma 3, $\overrightarrow{\mathbf{pp'}} = \lambda(O(r), O(r), 1 - O(r))$. To compute λ, we use the fact that $\mathbf{p}'$ is the projection on the regression plane of $\mathbf{p}$, and that $\mathbf{o}$ belongs to this plane by definition. This implies that $\overrightarrow{\mathbf{pp'}} \perp \overrightarrow{\mathbf{op'}}$ and therefore

$$\lambda^2 O(r^2) + \lambda(1 - O(r))(H\frac{r^2}{4} + o(r^2) + \lambda(1 - O(r))) = 0,$$

which yields $\lambda = \frac{Hr^2}{4} + o(r^2)$ and therefore

$$\overrightarrow{\mathbf{pp'}} = (O(r^3), O(r^3), \frac{Hr^2}{4} + o(r^2)) = \frac{Hr^2}{4}\mathbf{n} + o(r^2).$$

4.2 The asymptotic behavior of MLS2

Among the many versions of MLS2 proposed in the literature, we shall pick one which is a common denominator, and prone to a simple asymptotic analysis. In MLS2 a first intrinsic reference frame is first calculated, and the mean square approximation by order 2 polynomials is made in this reference frame. The most natural frame is found by applying MLS1, and the coordinates (x, y) are therefore the coordinates in the regression plane in a spherical neighborhood $\mathcal{B}_r$. The second step is to find the closest order 2 polynomial in the spherical neighborhood for the quadratic distance. Because of Lemma 1 we can specify, without loss of generality or precision, that this minimization is made in the cylindrical neighborhood $\mathcal{C}_r$. In that way, all integrals computed in the approximation process are integrals on the disk $x^2 + y^2 \leq r^2$, which is numerically and formally convenient. Thus the MLS2 algorithm which we shall analyze works in the two steps:

1. compute the regression plane of the manifold in the spherical neighborhood $\mathcal{B}_r = B_r(\mathbf{p}) \cap \mathcal{M}$;
2. call (x, y) the reference coordinates in the regression plane. Consider the restriction of the smooth manifold to the disk $D_r := x^2 + y^2 \leq r^2$, $z = f(x, y)$. Then find the order 2 polynomial $g(x, y)$ that best approximates f for the $L^2(D_r)$ distance;
3. set (in the reference frame) $MLS2(\mathbf{p}) := (0, 0, g(0, 0))$.

The next theorem shows that unlike MLS1, which reveals the mean curvature, the difference between a point smoothed by MLS2 and its original position is very small (of order 4) and actually reveals a fourth order intrinsic operator of bi-Laplacian type. Thus the evolution by an iterated MLS2 is a fourth order equation that is intuitively well-posed, at least for short times.

Theorem 6 *Consider a smooth manifold and its intrinsic coordinates* $(\tilde{x}, \tilde{y}, \tilde{z})$ *around a point* $\mathbf{p}(0,0)$*, so that the Taylor expansion in a neighborhood of* $\mathbf{p}$ *satisfies*

$$\tilde{z} = \tilde{f}(\tilde{x}, \tilde{y}) = \frac{1}{2}k_1\tilde{x}^2 + \frac{1}{2}k_2\tilde{y}^2 + \tilde{f}_3(x,y) + \tilde{f}_4(\tilde{x}, \tilde{y}) + \tilde{f}_5(\tilde{x}, \tilde{y}) + O(r^6)$$

where $\tilde{f}_i$ *are homogeneous polynomials in* $\tilde{x}, \tilde{y}$ *of global degree* i*. The second order approximation* $MLS2(\mathbf{p})$ *of* $\mathbf{p}$ *in a cylindrical neighborhood of* $\mathbf{p}$ *with radius* r *satisfies*

$$< MLS2(\mathbf{p}) - \mathbf{p}, \mathbf{n} >= -\frac{r^4}{48}(3\tilde{a}_{04} + \tilde{a}_{22} + 3\tilde{a}_{40})) + O(r^5)$$

where $\tilde{x}$ *and* $\tilde{y}$ *are the coordinates associated with the principal curvatures,* $\tilde{a}_{40} = \frac{1}{4!}\frac{\partial^4 f}{\partial \tilde{x}^4}$*,* $\tilde{a}_{04} = \frac{1}{4!}\frac{\partial^4 f}{\partial \tilde{y}^4}$*,* $\tilde{a}_{22} = \frac{1}{4!}\frac{\partial^4 f}{\partial \tilde{x}^2 \partial^2 \tilde{y}}$ *are the fourth derivatives of the intrinsic equation at* $\mathbf{p}$ *in the directions of* $\tilde{x}$*,* $\tilde{y}$ *and* $\tilde{x}, \tilde{y}$ *respectively, and* $\mathbf{n}$ *is the normal to the surface at* $\mathbf{p}$*, oriented towards the concavity.*

Lemma 4 *One can choose the coordinates* x *and* y *in the regression plane at* $\mathbf{p}$ *so that,* z *being the coordinate in the direction of the normal plane, the equation of the manifold around* $\mathbf{p}$ *has the form* $z = f(x,y) = \sum_{i,j=0}^{5} a_{ij}x^i y^j + o(|x^2 + y^2|^3)$*, and in addition* $a_{ij} = \tilde{a}_{ij} + O(r)$ *where* $\tilde{z} = \tilde{f}(\tilde{x}, \tilde{y}) = \sum_{i,j=0}^{5} \tilde{a}_{ij}\tilde{x}^i\tilde{y}^j + o(|\tilde{x}^2 + \tilde{y}^2|^3)$ *is the equation of the manifold in the intrinsic coordinates* $(\tilde{x}, \tilde{y}, \tilde{z})$ *defined by the normal at* $\mathbf{p}$ *and the directions of the principal curvatures.*

Proof Consider $(\tilde{x}, \tilde{y}, \tilde{z})$ the coordinates in the intrinsic frame such that $\tilde{x}$ and $\tilde{y}$ are the coordinates associated with the principal curvatures at $\mathbf{p}$, and the plane $\tilde{x}\mathbf{p}\tilde{y}$ is the tangent plane. Consider now coordinates (x, y, z) associated with the regression plane in a spherical neighborhood. Because the normal at the regression plane tends to the real normal when the spherical neighborhood shrinks, we can choose the coordinate axes (x, y) in the regression plane so that the rotation R which sends one frame to the other is close to the identity, namely

$$(\tilde{x}, \tilde{y}, \tilde{z}) = R(x, y, z) \tag{17}$$

with $R \to Id$ when $r \to 0$. More precisely, by Lemma 3, the normal $\mathbf{v}(r)$ to the PCA regression plane in a spherical neighborhood $\mathcal{B}_r$ at $\mathbf{p} \in \mathcal{M}$ is equal to the surface normal at point $\mathbf{p}$, up to a negligible factor: $\mathbf{v}(r) = \mathbf{n} + O(r)$. Thus we can pick $R(r)$ satisfying

$$R = R(r) = I + O(r). \tag{18}$$

Consider now the order 4 asymptotic expansion of $\tilde{z}$ as a function of $\tilde{x}, \tilde{y}$, where $\tilde{g}$ is a degree 4 polynomial. (We assume the manifold to be at least C^5):

$$\tilde{z} - \tilde{g}(\tilde{x}, \tilde{y}) - O((\tilde{x}^2 + \tilde{y}^2)^{\frac{5}{2}}) = 0.$$

By substituting in it the relation (17) the above equation becomes an implicit equation in z, x, y, R,

$$Q(x, y, z, R) - O((x^2 + y^2 + z^2)^{\frac{5}{2}}) = 0. \tag{19}$$

However, by the chain rule we have $\frac{\partial Q}{\partial z}(0,0,0,Id) = 1$. Thus by the implicit function theorem, there is a function h of class C^5 such that in a neighborhood of $(0,0,0,Id)$, (19) is equivalent to

$$z = h(x,y,R).$$

Since h is C^5 we can make a Taylor expansion of h and therefore get

$$z = g(x,y,R) + O(||R-Id||^5 + (x^2+y^2)^{\frac{5}{2}}).$$

In particular for $R = Id$ we have $(x,y,z) = (\tilde{x},\tilde{y},\tilde{z})$ and we obtain by identification of the terms with degree lower or equal to 4 that $g(x,y,Id) = \tilde{f}(x,y)$. Thus, all monomials $a_{ij}(R)x^iy^j$ in the expansion of f with respect to x,y satisfy $a_{i,j}(R) = \tilde{a}_{i,j}(Id) + O(I-R)$, which by (18) yields $a_{i,j}(r) = \tilde{a}_{i,j} + O(r)$.

Proof of Theorem 6. Let us write $f(x,y) = f_1(x,y)+f_2(x,y)+f_3(x,y)+f_4(x,y)+f_5(x,y)+o(|x^2+y^2|^{5/2})$ where

$$f_1(x,y) = a_{10}x + a_{01}y, \quad f_2(x,y) = a_{20}x^2 + a_{11}xy + a_{02}y^2$$
$$f_3(x,y) = a_{30}x^3 + a_{21}x^2y + a_{12}xy^2 + a_{03}y^3,$$
$$f_4(x,y) = a_{40}x^4 + a_{31}x^3y + a_{22}x^2y^2 + a_{13}xy^3 + a_{04}y^4,$$
$$f_5(x,y) = a_{50}x^5 + a_{41}x^4y + a_{32}x^3y^2 + a_{23}x^2y^3 + a_{14}xy^4 + a_{05}y^5.$$

We look for the order 2 polynomial g that best fits this surface in the least squares sense,

$$g(x,y) = \alpha x^2 + \beta y^2 + \gamma xy + \delta x + \epsilon y + \theta.$$

We therefore must find the parameters $\Theta = \begin{pmatrix}\alpha & \beta & \gamma & \delta & \epsilon & \theta\end{pmatrix}$ which minimize

$$\int_{x^2+y^2<r^2} (g(x,y) - f(x,y))^2 dxdy = \int_{x^2+y^2<r^2} (X\Theta^T - f(x,y))^2 dxdy$$

where $X = \begin{pmatrix}x^2 & y^2 & xy & x & y & 1\end{pmatrix}$. This is a quadratic minimization and differentiating this integral with respect to Θ yields

$$\int_{x^2+y^2<r^2} X^T(X\Theta^T - f(x,y))dxdy = 0.$$

Writing $M = \int_{x^2+y^2<r^2} X^TX$, the minimizer Θ satisfies

$$\Theta^T = \left(\int_{x^2+y^2<r^2} (X^TX)\right)^{-1} \int_{x^2+y^2<r^2} (X^Tf(x,y));$$

$$\Theta^T = M^{-1}\int_{x^2+y^2<r^2} X^T(f_1(x,y)+f_2(x,y)+f_3(x,y)+f_4(x,y)+f_5(x,y)+O((x^2+y^2)^3);$$

$$\text{where} \quad X^TX = \begin{pmatrix} x^4 & x^2y^2 & x^3y & x^3 & x^2y & x^2 \\ x^2y^2 & y^4 & xy^3 & xy^2 & y^3 & y^2 \\ x^3y & xy^3 & x^2y^2 & x^2y & xy^2 & xy \\ x^3 & xy^2 & x^2y & x^2 & xy & x \\ x^2y & y^3 & xy^2 & xy & y^2 & y \\ x^2 & y^2 & xy & x & y & 1 \end{pmatrix}.$$

When integrating on the disk, most terms vanish and we get

$$M = \frac{\pi r^4}{4}\begin{pmatrix} \frac{r^2}{2} & \frac{r^2}{6} & 0 & 0 & 0 & 1 \\ \frac{r^2}{6} & \frac{r^2}{2} & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{r^2}{6} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & \frac{4}{r^2} \end{pmatrix}; \quad M^{-1} = \frac{4}{\pi r^4}\begin{pmatrix} \frac{9}{2r^2} & \frac{3}{2r^2} & 0 & 0 & 0 & -\frac{3}{2} \\ \frac{3}{2r^2} & \frac{9}{2r^2} & 0 & 0 & 0 & -\frac{3}{2} \\ 0 & 0 & \frac{6}{r^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -\frac{3}{2} & -\frac{3}{2} & 0 & 0 & 0 & r^2 \end{pmatrix}.$$

Therefore

$$\begin{aligned} \Theta^T =& M^{-1}\int_{x^2+y^2<r^2} X^T f_1(x,y) + M^{-1}\int_{x^2+y^2<r^2} X^T f_2(x,y) \\ &+ M^{-1}\int_{x^2+y^2<r^2} X^T f_3(x,y) + M^{-1}\int_{x^2+y^2<r^2} X^T f_4(x,y) \\ &+ M^{-1}\int_{x^2+y^2<r^2} X^T f_5(x,y) + M^{-1}\int_{x^2+y^2<r^2} X^T O((x^2+y^2)^4), \end{aligned}$$

$$\text{with} \quad \int_{x^2+y^2<r^2} X^T f_1(x,y) = \frac{\pi r^4}{4}\begin{pmatrix} 0 \\ 0 \\ 0 \\ a_{10} \\ a_{01} \end{pmatrix};$$

$$\int_{x^2+y^2<r^2} X^T f_2(x,y) = \frac{\pi r^4}{4}\begin{pmatrix} \frac{r^2}{6}(3a_{20}+a_{02}) \\ \frac{r^2}{6}(a_{20}+3a_{02}) \\ \frac{r^2}{6}a_{11} \\ 0 \\ 0 \\ a_{20}+a_{02} \end{pmatrix};$$

$$\int_{x^2+y^2<r^2} X^T f_3(x,y) = \frac{\pi r^4}{4}\begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{r^2}{6}(3a_{30}+a_{12}) \\ \frac{r^2}{6}(a_{21}+3a_{03}) \\ 0 \end{pmatrix};$$

$$\int_{x^2+y^2<r^2} X^T f_4(x,y) = \frac{\pi r^4}{4}\begin{pmatrix} \frac{r^4}{16}(5a_{40}+a_{22}+a_{04}) \\ \frac{r^4}{16}(a_{40}+a_{22}+5a_{04}) \\ \frac{r^4}{16}(a_{31}+a_{13}) \\ 0 \\ 0 \\ \frac{r^2}{6}(3a_{40}+a_{22}+3a_{04}) \end{pmatrix};$$

$$\int_{x^2+y^2<r^2} X^T(f_5(x,y)) = \frac{\pi r^4}{4}\begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{r^4}{16}(5a_{50}+a_{32}+a_{14}) \\ \frac{r^4}{16}(a_{41}+a_{23}+5a_{05}) \\ 0 \end{pmatrix};$$

$$\int_{x^2+y^2<r^2} X^T(x^2+y^2)^3 = \frac{\pi r^4}{4}\begin{pmatrix} \frac{2r^6}{5} \\ \frac{2r^6}{5} \\ 0 \\ 0 \\ 0 \\ \frac{r^4}{2} \end{pmatrix}.$$

Multiplying all of these results by the matrix M^{-1}, we get

$$M^{-1}\int_{x^2+y^2<r^2} X^T f_1(x,y) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ a_{10} \\ a_{01} \\ 0 \end{pmatrix}; \tag{20}$$

$$M^{-1}\int_{x^2+y^2<r^2} X^T f_2(x,y) = \begin{pmatrix} a_{20} \\ a_{02} \\ a_{11} \\ 0 \\ 0 \\ 0 \end{pmatrix}; \tag{21}$$

$$M^{-1}\int_{x^2+y^2<r^2} X^T(f_3(x,y)) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{r^2}{6}(3a_{30}+a_{12}) \\ \frac{r^2}{6}(a_{21}+3a_{03}) \\ 0 \end{pmatrix}; \tag{22}$$

$$M^{-1}\int_{x^2+y^2<r^2} X^T(f_4(x,y)) = \begin{pmatrix} \frac{r^2}{8}(6a_{40}+a_{22}) \\ \frac{r^2}{8}(a_{22}+6a_{04}) \\ \frac{3r^2}{8}(a_{31}+a_{13}) \\ 0 \\ 0 \\ -\frac{r^4}{48}(3a_{40}+a_{22}+3a_{04}) \end{pmatrix}; \tag{23}$$

$$M^{-1}\int_{x^2+y^2<r^2} X^T(f_5(x,y)) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{r^4}{16}(5a_{50}+a_{32}+a_{14}) \\ \frac{r^4}{16}(a_{41}+a_{23}+5a_{05}) \\ 0 \end{pmatrix}; \tag{24}$$

$$M^{-1}\int_{x^2+y^2<r^2} X^T(x^2+y^2)^3 = \begin{pmatrix} \frac{33r^4}{20} \\ \frac{33r^4}{20} \\ 0 \\ 0 \\ 0 \\ -\frac{7r^6}{10} \end{pmatrix}, \tag{25}$$

and combining equations (20), (21), (22), (23), (24) and (25) we finally obtain the parameter Θ

$$\Theta^T = \begin{pmatrix} a_{20} + \frac{r^2}{8}(a_{22}+6a_{40}) + O(r^4) \\ a_{02} + \frac{r^2}{8}(a_{22}+6a_{04}) + O(r^4) \\ a_{11} + \frac{3r^2}{8}(a_{13}+a_{31}) + O(r^4) \\ a_{10} + \frac{r^2}{6}(a_{12}+3a_{30}) + \frac{r^4}{16}(5a_{50}+a_{32}+a_{14}) + O(r^5) \\ a_{01} + \frac{r^2}{6}(3a_{03}+a_{21}) + \frac{r^4}{16}(a_{41}+a_{23}+5a_{05}) + O(r^5) \\ -\frac{r^4}{48}(3a_{04}+a_{22}+3a_{40}) + O(r^6) \end{pmatrix}$$

so that the MLS2 projection satisfies $g(0,0) = -\frac{r^4}{48}(3a_{04}+a_{22}+3a_{40}) + O(r^6)$. Finally Lemma 4 permits to replace $g(0,0) = -\frac{r^4}{48}(3a_{04}+a_{22}+3a_{40}) + O(r^6)$. by

$$-(\frac{r^4}{48}(3\tilde{a}_{04}+\tilde{a}_{22}+3\tilde{a}_{40}) + O(r^6))(1+O(r)) = -\frac{r^4}{48}(3\tilde{a}_{04}+\tilde{a}_{22}+3\tilde{a}_{40}) + O(r^5).$$

□

We shall now analyze experimentally those results.

5 Numerical experiments

This section performs numerical comparative experiments with the most significant algorithms described in the previous sections. A simulated randomly sampled sphere will play the role of numerical pattern. In particular we evaluate the mean curvatures given on the sphere by MLS1 projection and MLS2 projection followed by polynomial regression. We also compute the curvature estimated by the method described in [7] and by the surface variation of [40]. The results are compared by giving the mean estimated curvature and its standard variation. The input data is a randomly sampled sphere with radius 2 corrupted with added centered Gaussian noise of variance 0.1.

Iteration	MLS1		MLS2	
	mean	standard variation	mean	standard variation
0	0.5828	2.8609	0.052	1.2879
1	0.5158	1.2434	0.4920	1.0053
2	0.5079	0.3196	0.5083	0.1259
3	0.5102	0.0253	0.5073	0.1001
4	0.5136	0.0189	0.5068	0.0855
5	0.5171	0.0165	0.5065	0.0749
10	0.5356	0.0156	0.5058	0.0489

Fig. 2 Comparison of the curvature estimation by iteration of the MLS1 projection and iterations of the MLS2 projection, with the same radius.

By comparing the values in the table of fig 2, two conclusions can be drawn: first, MLS1 is significantly more stable than the MLS2 projection, which can be observed by the standard variation on the estimate. The SNR gain is close to 4 by using MLS1 instead of MLS2 with the same iteration number. In conformity with Theorem 5, MLS1 projection yields an increase of the mean curvature (i.e., the sphere radius decreases, which is expected from a mean curvature motion). These results can be compared to the other two main curvature estimators.

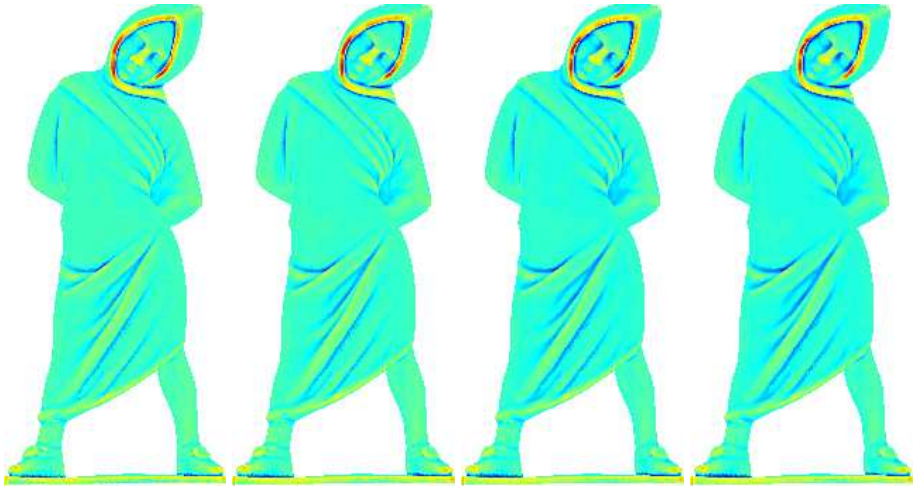
Figs 3, 4 and 5) show various curvature distribution and surface variation distributions illustrating the interest of computing such operators to classify surface points.

(a) Iteration 1 (b) Iteration 2 (c) Iteration 3 (d) Iteration 4

Fig. 3 Curvature evolution by iterative projection on MLS1

Another experiment permits to better judge of the MLS1 smoothing effect. First, a consistently oriented point set was built (see [17] for an efficient way of doing so). This normal orientation yields the sign of the mean curvature, by computing the scalar product of the oriented normal and the displacement vector. Each point was then plotted in a different color according to its sign, blue for positive and red for negative (see Fig 6). This experiment shows that, at the beginning, the curvature sign captures essentially noise and small texture. After several iterations, the shape is smoothed and the sign captures the geometry of the shape (large scale variations), which is the main advantage of the scale space strategy.

To compare the techniques analyzed theoretically in the previous sections, we finally used randomly sampled shapes with added Gaussian noise. We compared between computing the covariance of the points projected onto the local tangent plane, as described in section 3.1 (called 2dcov1 in the remainder of this section); computing the covariance of the unit normals projected on the regression plane, as described in section 3.2 (called 2dcov2 in the remainder of this section); computing



(a) Iteration 1 (b) Iteration 2 (c) Iteration 3 (d) Iteration 4

Fig. 4 Curvature evolution by iterative projection on MLS2 .

(a) Surface variation (b) Curvature by normal covariance analysis

Fig. 5 Other curvature estimates .

the covariance of the unit normals, as described in section 3.3 (called 3dcov in the remainder of this section); and finally MLS2. The 2dcov1 method was immediately discarded, because it does not yield a separate estimate of the principal curvatures. We therefore only compared the other three methods.

To do so the estimators were compared on three kinds of noisy surfaces: a sphere, a cylinder and a torus with added gaussian noise in the normal direction.

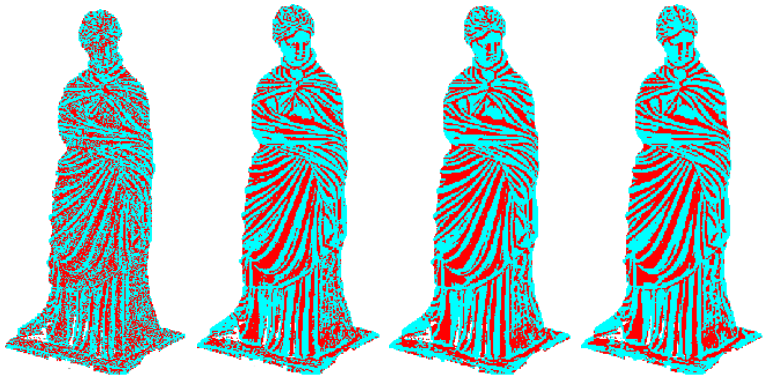


Fig. 6 Evolution of the motion direction with projection iterations.

The goal was not to compute the actual curvatures of the noiseless surface, but to contrast the robustness of these local surface geometric indicators which replace the infinitesimal curvatures. The asymptotic theorems 2, 3, 1 and 2 show that the considered intrinsic local integral estimators retain the same structure properties as the real differential operators. Thus, our goal is to select among them the ones that have the best SNR. On the other hand the SNR depends of course of the radius on which these operators are computed. The smaller the radius, the more faithful these operators will be to local curvature operators. Thus, all things being equal, it is better to compute them with small radii. But of course the SNR decreases with the radius. Thus, to compare the power of these local integral operators, the best way is to compare the SNR's with fixed radius or, equivalently, to compare their radii for fixed SNR. Yet there are two parameters for MLS1: the radius r and the number of iterations N. But both parameters can be an equivalent radius. Indeed, applying N iterations of MLS1 with radius r is roughly equivalent to applying one iteration of the scale space with radius $r_q = \sqrt{N}r$ (this equivalence is drawn by analogy with iterated linear filters). The numerical tables give the equivalent radius value for MLS1.

A sphere and a cylinder were used because they have constant curvatures on the whole surface. We also considered a torus, because one can compute the operators on invariant circles of the torus, where curvatures should be constant in absence of noise. As just explained, the radii were set so that each method gives the same standard variation for the estimate of the same curvature.

One should first notice that 3dcov and 2dcov2 give very similar results. This is not surprising since both methods rely on the normal covariance matrix (actually, even their asymptotic behavior is the same, see theorems 2 and 3).

Tables 7, 8, 9 show the results after this calibration of the experiment by the standard deviation. Two conclusions can be drawn from these experiments. First, the radii needed for getting a small standard variation are slightly larger for 2dcov2, 3dcov and significantly larger for MLS2, than for MLS1. This has the direct consequence that computation times are significantly higher for those methods than for MLS1: it is indeed faster to iterate a method working on a small neighborhood than to do a single iteration of a method requiring a large neighborhood. Nevertheless, MLS1 only provides us with an estimated equivalent

	parameters	stdev	SNR
MLS1	$N = 5, r = 0.14$, $r_{eq} = 0.31$	0.026	20.0
2dcov2	$r = 0.3$	0.027	17.6
3dcov	$r = 0.3$	0.027	17.6
MLS2	$r = 0.6$	0.029	15.6

Fig. 7 Comparison of the mean curvature estimates on a noisy sphere with radius 2 and noise variance 0.05. Calibration is done by setting the parameters so that the standard deviation is similar. The parameters are the neighborhood radius r and the number of iterations N in the case of MLS1. The best SNR is obtained with MLS1, which is also the fastest method. But 2dcov2 and 3dcov have similar performance, while MLS2 is clearly worse (its radius doubles for a lower SNR.)

	parameters	stdev	SNR
MLS1	$N = 5, r = 0.16$, $r_{eq} = 0.36$	0.040	16.2
2dcov2	$r = 0.37$	0.032	16.9
3dcov	$r = 0.3$	0.031	16.9
MLS2	$r = 0.6$	0.036	13.2

Fig. 8 Comparison of the mean curvature estimates on a noisy cylinder with radius 1 and noise variance 0.05. The radius r or equivalent radius r_{eq} are set so that the standard deviation becomes similar. The conclusions are the same as in fig 7.

	parameters	stdev	SNR
MLS1	$N = 4, r = 0.07$, $r_{eq} = 0.14$	0.032	34.2
2dcov2	$r = 0.15$	0.031	31.8
3dcov	$r = 0.15$	0.031	31.8
MLS2	$r = 0.32$	0.030	1.2

Fig. 9 Comparison of the mean curvature estimates for an invariant circle of a noisy torus with radii 2 and 0.5 and noise variance 0.02. Here again, the filtering radii were chosen so that the standard deviation becomes similar, and the SNR's and radii can therefore be compared. Here again MLS1 wins by a small margin on 2dcov2 and 3dcov, and by a large margin over MLS2.

to the mean curvature but does not give an estimated equivalent of the principal curvatures nor of the principal directions. As a matter of fact only MLS2 provides this information: 2dcov2 and 3dcov only provide the principal directions and the squared principal curvatures. Yet we saw that in order for MLS2 to be resilient to noise a large neighborhood must be used which leads to huge computation times.

Since we proved that MLS1 is consistent with an intrinsic heat equation, it plays the special role among the considered operators of simulating a scale space semigroup. Using the scale space paradigm, it can be used previously to the computation of other differential operators. We did again the same computations using the MLS1 iterations (scale space) before applying the more complex methods. We used the same number of iterations for MSLS1 as found in tables 7, 8, 9 and performed the next analysis using the same radius. The new tables (Tabs 10, 11 and 12) show how the scale space makes it possible to compute reliably the same moments with a smaller processing radius. Computation times being the bottleneck of all numerical methods, we compare on Tab. 13 the computation times obtained on the cylinder when applying all the methods (with the same parameters as in

	MLS2	2dcov2	3dcov
without scale space	$276s$	$196s$	$188s$
with scale space	$90s$	$85s$	$87s$

Fig. 13 Computation time for the cylinder experiments.

figs 8 and 11: it is straightforward that applying the MLS1 iterations is a much better strategy to get manageable numerical experiments.

	H		k_1		k_2	
	SNR	std	SNR	std	SNR	std
2dcov2	19.19	0.016	0.349	0.048	0.048	0.037
3dcov	19.19	0.016	0.349	0.048	0.048	0.034
MLS2	0.645	0.053	1.283	0.0871	0.036	0.037

Fig. 10 Sphere example: applying scale space iterations before further analysis. The parameters of MLS1 iterations are the ones found in the calibration process: $r = 0.14$ and $N = 5$.

	H		k_1		k_2	
	SNR	std	SNR	std	SNR	std
2dcov2	55.55	0.026	56.55	0.030	4.03	0.029
3dcov	55.67	0.026	55.70	0.030	3.92	0.029
MLS2	46.58	0.026	59.91	0.029	1.30	0.029

Fig. 11 Cylinder example: applying scale space iterations before further analysis. Parameters of MLS1 iterations are the ones found in the calibration process: $r = 0.16$ and $N = 5$. The SNR for k_2 is of course not meaningful, the asymptotic theoretical mean of k_2 being 0.

	H		k_1		k_2	
	SNR	std	SNR	std	SNR	std
2dcov2	39.56	0.041	34.73	0.0741	12.90	0.035
3dcov	39.54	0.041	34.73	0.0740	12.90	0.036
MLS2	35.00	0.0563	11.09	0.0713	37.67	0.067

Fig. 12 Torus example: applying scale space iterations before further analysis. Parameters of MLS1 iterations are the ones found in the calibration process: $r = 0.07$ and $N = 4$.

These numerical experiments confirm that the only way to recover a robust signed integro-differential operator, equivalent to the mean curvature, is to apply the scale space (iterations of MLS1) and then MLS2. If the sign is not needed, any of the 2dcov2 or 3dcov can be used. Since 3dcov is simpler to compute, this would, in this case, be our best choice.

Conclusion

In this paper, we analyzed the local intrinsic moments of smooth surfaces proposed in the literature, and linked them by an asymptotic analysis to the surface principal curvatures. The interest of such local moments is that they can be computed directly on raw point clouds and therefore allow for a direct numerical analysis of such raw data. We showed that these clever methods only recover the equivalent of squared principal curvatures, and loose their signs.

The alternative method to compute curvatures on the surface is the order 2 regression MLS2. An asymptotic analysis of MLS2 confirms that is is accurate with order 4 and also uncovers a new intrinsic fourth order partial differential operator arising naturally from this order 2 regression.

Finally the analysis of the MLS1 projection (recalled from [17]) yields a mean curvature motion. Once iterated this *scale space operator, proven very robust to irregular sampling, gives an alternative way to compute curvatures by combining scale space and MLS2. Numerical experiments herewith have shown this to be the most reliable method, in agreement with the scale space methodology already established in image analysis.*

References

1. *Alexa, M., Adamson, A.: Interpolatory point set surfaces convexity and hermite data. ACM Trans. Graph.* ***28****, 20:1–20:10 (2009)*
2. *Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T.: Point set surfaces. In: Proc. Vis '01, pp. 21–28. IEEE Computer Society, Washington, DC, USA (2001)*
3. *Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T.: Computing and rendering point set surfaces. IEEE TVCG* ***9****(1), 3–15 (2003)*
4. *Alliez, P., Cohen-Steiner, D., Tong, Y., Desbrun, M.: Voronoi-based variational reconstruction of unoriented point sets. In: SGP '07, pp. 39–48. Eurographics, Switzerland (2007)*
5. *Amenta, N., Kil, Y.J.: Defining point-set surfaces. In: SIGGRAPH '04: ACM SIGGRAPH 2004 Papers, pp. 264–270. ACM Press, USA (2004)*
6. *Belkin, M., Sun, J., Wang, Y.: Constructing laplace operator from point clouds in rd. In: Proc. SODA '09, pp. 1031–1040. SIAM, USA (2009)*
7. *Berkmann, J., Caelli, T.: Computation of surface geometry and segmentation using covariance techniques. IEEE PAMI* ***16****(11), 1114–1116 (1994)*
8. *Buades, A., Coll, B., Morel, J.M.: Neighborhood filters and pdes. Numer. Math.* ***105****(1), 1–34 (2006)*
9. *Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C., Evans, T.R.: Reconstruction and representation of 3d objects with radial basis functions pp. 67–76 (2001)*
10. *Cazals, F., Pouget, M.: Estimating differential quantities using polynomial fitting of osculating jets. In: SGP '03, pp. 177–187. Eurographics, Switzerland (2003)*
11. *Cazals, F., Pouget, M.: Topology driven algorithms for ridge extraction on meshes. Tech. rep., INRIA (2005)*
12. *Clarenz, U., Griebel, M., Rumpf, M., Schweitzer, M.A., Telea, A.: Feature sensitive multiscale editing on surfaces. Vis. Comput.* ***20****, 329–343 (2004)*
13. *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****, 516–524 (2004)*
14. *Cohen-Steiner, D., Morvan, J.M.: Restricted delaunay triangulations and normal cycle. In: Proc. SCG '03, pp. 312–321. ACM, USA (2003)*
15. *Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: SIGGRAPH '96, pp. 303–312. ACM Press, USA (1996)*

16. *Desbrun, M., Meyer, M., Schroder, P., Barr, A.H.: Implicit fairing of irregular meshes using diffusion and curvature flow. In: SIGGRAPH '99, pp. 317–324. ACM Press/Addison-Wesley Publishing Co., USA (1999)*
17. *Digne, J., Morel, J.M., Mehdi-Souzani, C., Lartigue, C.: Scale space meshing of raw data point sets. Computer Graphics Forum pp. no–no (2011)*
18. *Fleishman, S., Cohen-Or, D., Silva, C.T.: Robust moving least-squares fitting with sharp features. ACM Trans. Graph.* ***24****(3), 544–552 (2005)*
19. *Gelfand, N., Mitra, N.J., Guibas, L.J., Pottmann, H.: Robust global registration. In: Proceedings of the third Eurographics symposium on Geometry processing. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (2005)*
20. *Gross, M., Hubeli, A.: Eigenmeshes. Tech. rep., ETH Zurich (2000)*
21. *Guennebaud, G., Gross, M.: Algebraic point set surfaces. ACM Trans. Graph.* ***26*** *(2007)*
22. *Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Surface reconstruction from unorganized points. In: SIGGRAPH '92, pp. 71–78. ACM Press, USA (1992)*
23. *Hubeli, A., Gross, M.: Multiresolution feature extraction for unstructured meshes. In: VIS '01: Proceedings of the conference on Visualization '01, pp. 287–294. IEEE Computer Society, Washington, DC, USA (2001)*
24. *Hulin, D., Troyanov, M.: Mean curvature and asymptotic volume of small balls. The American Mathematical Monthly* ***110****(10), pp. 947–950 (2003)*
25. *Kalogerakis, E., Simari, P., Nowrouzezahrai, D., Singh, K.: Robust statistical estimation of curvature on discretized surfaces. In: SGP '07, pp. 13–22. Eurographics, Switzerland (2007)*
26. *Karni, Z., Gotsman, C.: Spectral compression of mesh geometry. In: SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques, pp. 279–286. ACM Press/Addison-Wesley Publishing Co., USA (2000)*
27. *Kazhdan, M.: Reconstruction of solid models from oriented point sets. In: SGP '05, p. 73. Eurographics Association, Switzerland (2005)*
28. *Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: SGP '06, pp. 61–70. Eurographics, Switzerland (2006)*
29. *Kobbelt, L.P., Botsch, M., Schwanecke, U., Seidel, H.P.: Feature sensitive surface extraction from volume data. In: SIGGRAPH '01, pp. 57–66. ACM, USA (2001)*
30. *Lancaster, P., Salkauskas, K.: Surfaces generated by moving least squares methods. Mathematics of Computation* ***37****(155), 141–158 (1981)*
31. *Levin, D.: The approximation power of moving least-squares. Math. Comput.* ***67****(224), 1517–1531 (1998)*
32. *Levin, D.: Mesh-independent surface interpolation. In: H. Brunnett, Mueller (eds.) Geometric Modeling for Scientific Visualization, pp. 37–49. Springer-Verlag (2003)*
33. *Liang, P., Todhunter, J.S.: Representation and recognition of surface shapes in range images: a differential geometry approach. Comput. Vision Graph. Image Process.* ***52****(1), 78–109 (1990)*
34. *Lipman, Y., Cohen-Or, D., Levin, D.: Data-dependent mls for faithful surface approximation. In: SGP '07, pp. 59–67. Eurographics, Switzerland (2007)*
35. *Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. In: SIGGRAPH '87, pp. 163–169. ACM Press, USA (1987)*
36. *Magid, E., Soldea, O., Rivlin, E.: A comparison of gaussian and mean curvature estimation methods on triangular meshes of range image data. CVIU* ***107****(3), 139–159 (2007)*
37. *Mérigot, Q., Ovsjanikov, M., Guibas, L.J.: Robust Voronoi-based Curvature and Feature Estimation. In: SIAM/ACM Joint Conference on Geometric and Physical Modeling. San Francisco, USA (2009)*
38. *Meyer, M., Desbrun, M., Schröder, P., Barr, A.: Discrete differential geometry operators for triangulated 2-manifolds. In: International Workshop on Visualization and Mathematics (2002)*
39. *Oztireli, A.C., Guennebaud, G., Gross, M.: Feature preserving point set surfaces based on non-linear kernel regression. CGF* ***28****, 493–501(9) (2009)*
40. *Pauly, M., Gross, M., Kobbelt, L.P.: Efficient simplification of point-sampled surfaces. In: Proc. VIS '02, pp. 163–170. IEEE, USA (2002)*
41. *Pauly, M., Kobbelt, L.P., Gross, M.: Point-based multiscale surface representation. ACM Trans. Graph.* ***25****(2), 177–193 (2006)*
42. *Pottmann, H., Wallner, J., Huang, Q.X., Yang, Y.L.: Integral invariants for robust geometry processing. CAGD* ***26****(1), 37–60 (2009)*

43. *Pottmann, H., Wallner, J., Yang, Y.L., Lai, Y.K., Hu, S.M.: Principal curvatures from the integral invariant viewpoint. CAGD* ***24****(8-9), 428–442 (2007)*
44. *Rusinkiewicz, S.: Estimating curvatures and their derivatives on triangle meshes. In: 3DPVT '04, pp. 486–493. IEEE, USA (2004)*
45. *Tang, X.: A sampling framework for accurate curvature estimation in discrete surfaces. IEEE TVCG* ***11****(5), 573–583 (2005)*
46. *Taubin, G.: Estimating the tensor of curvature of a surface from a polyhedral approximation. In: ICCV '95, p. 902. IEEE, USA (1995)*
47. *Taubin, G.: A signal processing approach to fair surface design. In: SIGGRAPH '95, pp. 351–358. ACM Press, USA (1995)*
48. *Theisel, H., Rossl, C., Zayer, R., Seidel, H.P.: Normal based estimation of the curvature tensor for triangular meshes. In: Proc. PG' 04, pp. 288–297. IEEE, USA (2004)*
49. *Tong, W.S., Tang, C.K.: Robust estimation of adaptive tensors of curvature by tensor voting. IEEE PAMI* ***27****(3), 434–449 (2005)*
50. *Unnikrishnan, R., Hebert, M.: Multi-scale interest regions from unorganized point clouds. In: Workshop on Search in 3D (S3D), IEEE CVPR (2008)*
51. *Yang, P., Qian, X.: Direct computing of surface curvatures for point-set surfaces. In: Proc. PBG 07 (2007)*
52. *Yang, Y.L., Lai, Y.K., Hu, S.M., Pottmann, H.: Robust principal curvatures on multiple scales. In: SGP '06, pp. 223–226. Eurographics, Aire-la-Ville, Switzerland (2006)*