SINEFITTING: ROBUST CURVATURE ESTIMATION ON SURFACE TRIANGULATION

Jérôme Charton, Stefka Gueorguieva and Pascal Desbarats
LaBRI (UMR 5800), Université Bordeaux 1
351, Cours de la Libération, F-33405 Talence cedex

ABSTRACT

A novel algorithm for robust curvature estimation based on sinusoidal curve fitting is proposed. The evaluation of this algorithm is presented on analytical surface triangulations by comparing it with other recognized fitting methods according to three criteria: convergence, precision and robustness. By experimenting on various data, we show that the *Sinefitting* algorithm is less affected by errors in vertex normal estimation.

KEYWORDS

triangular meshes, generalized curvature estimation, surface fitting, curvature measures

1. RELATED WORK

Knowledge about the geometric shape of an object is based on the understanding of the differential structure of the object boundary surface: the principal curvatures and directions, the Gauss and the Mean curvatures of the boundary surface. We are interested in estimating this differential structure of the underlying smooth surface from a given triangulation. Since the pioneer works of [1, 21, 7], curvature estimation is a central issue in a great number of research. For surveys we refer the reader to [15, 19, 23]. Generalized curvatures, convergence and measure stability are covered by [12, 18, 3]. The approaches considered in this paper are based on the fitting paradigm: first, a quadratic form is constructed at each vertex of the surface triangulation and then the local differential structure is derived. Chen et al [4] use the normal curvature as stated by the Meusnier and Euler's theorem and locally fit a set of circles through the neighbour vertices. McIvor et al [16] use a quadratic surface fitting for determining the principal frame and the rotated principal quadric. Different variants of this method are applied depending on the type of the quadratic form (simple SQFA, extended or full). More generally, Cazals et al [2] make use of osculating jets defined as truncated Taylor expansions. For Taubin [24] the quadratic form is expressed as an integral representation that is used to obtain the curvature tensor. Langer et al [14] use integral representations of the Gauss and the Mean curvatures. Cohen-Steiner et al [5] use integrals of specific differential forms on their normal cycles. Fitting enables precise estimation of curvatures but it is very sensitive to the surface discretization and to the distribution of the edge directions in the vertex neighbourhood. Indeed, these methods make use of the surface normal approximation which strongly depends on the regularity of the triangles in this neighbourhood.

In the current paper, a novel method for curvature estimation, called *SineFitting*, is presented. At each vertex of the surface we construct a sinusoidal curve to fit the directional angles from the target vertex to its neighbours. Therefore, in contrast to Chen's method, there is no need to restrict the choice of neighbours to specific configurations (pairs of geometrically opposite vertices). Moreover, computations in under or oversampled neighbourhoods remain robust while maintaining high precision. The elaborated method has a linear convergence rate and is not acutely affected by errors in estimation of the normal.

2. OVERVIEW OF THE SINEFITTING METHOD

We refer the reader to [22] for detailed discussion on surface differential geometry. Let S be a surface, P a target vertex from S and $\vec{N}$ the unit normal vector to S in P. Let τ be the tangent plane of S in P, see Fig.1. Given a unit vector $\vec{T}$ in τ, the osculating plane ι through $\vec{N}$ and $\vec{T}$ intersects S in a curve c. Let $\vec{n}$ be the unit normal vector of c in P and ϑ the angle between $\vec{n}$ and $\vec{N}$, $\vartheta = (\vec{n}, \vec{N})$.

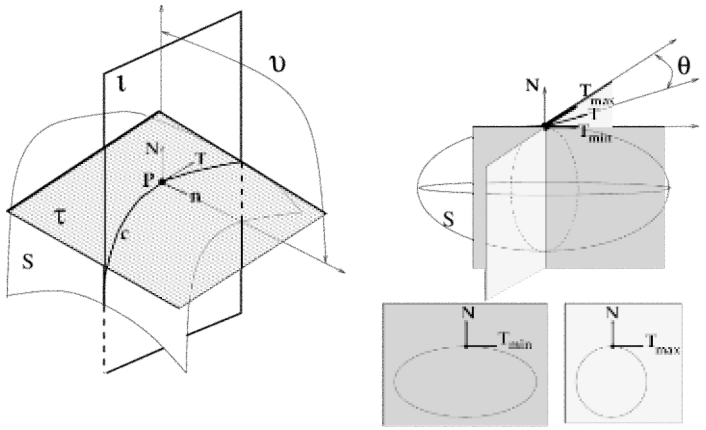


Figure 1: Local surface geometry around point P

According to Meusnier & Euler's theorem, the normal curvature k_T of S in P along $\vec{T}$ could be defined as

$$k_T = k.\cos(\vartheta) \quad (1)$$

where k denotes the curvature of c in P and $\cos(\vartheta) = \vec{n} \cdot \vec{N}$. While $\vec{T}$ is rotated around $\vec{N}$, an infinite number of normal curvatures k_T could be defined. The extreme values of k_T, k_{min} and k_{max}, are achieved along the principal directions, $\overrightarrow{T_{min}}$ and $\overrightarrow{T_{max}}$. Euler's theorem gives the relation between k_T, k_{min}, k_{max} and the angle θ, $\theta = \angle(\vec{T}, \overrightarrow{T_{max}})$.

$$k_T = k_{max} \cdot cos^2(\theta) + k_{min} \cdot sin^2(\theta) \quad (2)$$

The proposed algorithm, called *SineFitting* algorithm, is based on (1) and (2) in order to evaluate the principal directions $\overrightarrow{T_{min}}$ and $\overrightarrow{T_{max}}$, the principal curvatures k_{min} and k_{max}, the Gauss and the Mean curvatures, k_G and k_H, in P. Our approach is based on a two steps procedure: First, the normal curvature k_T is evaluated according to (1) for curves locally fitting the normal sections of the surface. Second, a sinusoidal curve is constructed to approximate the computed values of k_T following (2). Principal curvatures and directions are calculated for specified values of the sinusoidal amplitude and frequency. Let T_S be a triangulation of S, $N(P)$ a neighbourhood of P, $N=\{P_i, PP_i \in T_S, i=0,...,m-1\}$ and k_i the normal curvature of S along PP_i. Let P_i^* be the projection of P_i on τ, M_i the middle point of PP_i and M_i the median of PP_i, shown in Fig.1. Let us construct a circle σ passing through P and P_i, and centred at O_i, $O_i = l_{\vec{N}} \cap m_i$, where $l_{\vec{N}}$ is the straight line supporting $\vec{N}$. Let $\vec{n}$ be the unit normal of σ, $\vec{n_\sigma} = \dfrac{\overrightarrow{PO_i}}{\left\|\overrightarrow{PO_i}\right\|}$, and k_σ be the normal curvature of σ, $k_\sigma = \dfrac{1}{\left\|\overrightarrow{PO_i}\right\|}$.

Let us now apply (1) to k_i and k_σ:

$$k_i = k_\sigma.\cos(\angle(\vec{N}, \vec{n})) \quad (3)$$

The unit vectors $\vec{N}$ and $\vec{n}$ are aligned with opposite directions and thus $\cos(\angle(\vec{N}, \vec{n})) = -1$. The curvature k_i can be expressed then as:

$$k_i = -k_\sigma = \frac{-1}{\left\|\overrightarrow{PO_i}\right\|} \quad (4)$$

Let ψ be the angle $\angle(M_i, P, O_i)$. From the right-angled triangle $\Delta(M_i, P, O_i)$ it follows that

$$\cos(\psi) = \frac{\left\|\overrightarrow{PM_i}\right\|}{\left\|\overrightarrow{PO_i}\right\|} = 2 \cdot \frac{\left\|\overrightarrow{PP_i}\right\|}{\left\|\overrightarrow{PO_i}\right\|} \quad (5)$$

The value of $cos(\psi)$ can be estimated from the scalar product of $-\vec{N}$ and $\overrightarrow{PP_i}$ and by the substitution in (4) we finally compute the estimation of k_i:

$$k_i = \frac{2 \cdot \vec{N} \cdot \overrightarrow{PP_i}}{\left\|\overrightarrow{PP_i}\right\|^2} \quad (6)$$

It should be noticed that the estimation (6) is similar to the estimation in [24]. The above construction gives a geometrical insight of the normal curvature estimation.

The second step of our algorithm includes the construction of a sinusoidal curve that approximates the normal curvature in $N(P)$. Our goal is to rise the constraint of using $\overrightarrow{T_{min}}$ and $\overrightarrow{T_{max}}$ in (2), and fit a linear sinusoidal expression. According to (2) we have:

$$k_i = k_{max} \cdot cos^2(\theta_i) + k_{min} \cdot sin^2(\theta_i) \quad (7)$$

where $\theta_i = \angle(\vec{T_i}, \overrightarrow{T_{max}})$. We introduce the angle ϕ

$$\phi = \angle(\vec{T_0}, \overrightarrow{T_{max}}) \quad (8)$$

where $\vec{T_0}$ is any vector from $N^*(P)$, $N^*(P) = \{PP_i^*, i=0, ..., m-1\}$. In (7), θ_i is substituted by α_i, $\alpha_i = \theta_i - \phi$. Then (7) is rewritten as:

$$k_i = k_{max} \cdot cos^2(\alpha_i + \phi) + k_{min} \cdot sin^2(\alpha_i + \phi) \quad (9)$$

Next we substitute ϕ, k_{min} and k_{max} as:

$$\phi = -\frac{\arctan\left(\frac{b}{a}\right)}{2}, a > 0 \quad (10) \qquad k_{min} = c - \sqrt{(a^2 + b^2)} \quad (11) \qquad k_{max} = c + \sqrt{(a^2 + b^2)} \quad (12)$$

The equation (9) is rewritten as:

$$k_i = a \cdot \cos(2\alpha_i) + b \cdot \sin(2\alpha_i) + c \quad (13)$$

Now we are ready to compute a sinusoidal curve to approximate in a least square way the estimated from (6) normal curvatures. The intuition of this approach is to first estimate a sine wave shape of arbitrary phase. The peaks of its deviation will correspond to the values of k_{min} and k_{max}. The principal directions $\overrightarrow{T_{min}}$ and $\overrightarrow{T_{max}}$ will correspond to the phases of the leftmost peaks in relation to the origin.

The *SineFitting* algorithm is given in Algo. 1

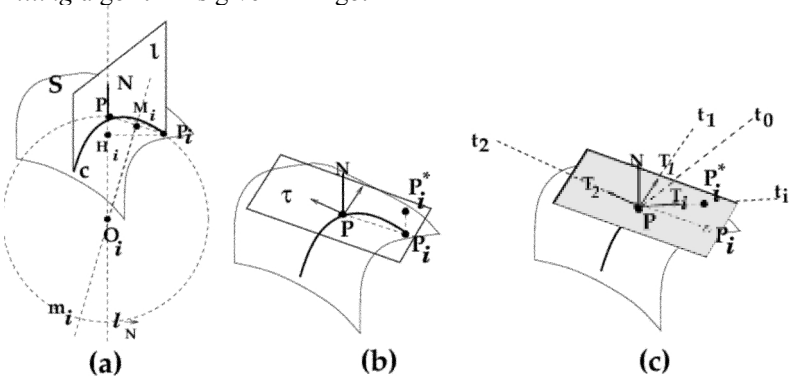


Figure 2: *SineFitting* curvature estimation over one ring neighbourhood

Algorithm 1 *SineFitting*

Require: P and T_S

Ensure: $\overrightarrow{T_{min}}$ and $\overrightarrow{T_{max}}$, k_{min} and k_{max}, k_G and k_H at P

1: $N(P) \leftarrow$ *extract neighbour vertices* P_i *of* P
Let $m=|N(P)|$, *the number of neighbours*

2: $N \leftarrow$ *the unit normal vector in* P

3: $\tau \leftarrow$ *the tangent plane in* P *with normal* $\vec{N}$

4: **for** $i = 1$ **to** $m - 1$ **do**

5: $P_i^* \leftarrow$ *projection of* P_i *on* τ

6: $\vec{T}_i = \dfrac{\overrightarrow{PP_i^*}}{\left\|\overrightarrow{PP_i^*}\right\|}$

7: **end for**

8: **for** i=1 **to** m-1 **do**

9: $\alpha_i = arccos\left(\dfrac{\vec{T}_0 \cdot \vec{T}_i}{\left\|\vec{T}_0\right\| \cdot \left\|\vec{T}_i\right\|}\right)$

10: $k_i^* = 2 \cdot \dfrac{\overrightarrow{P_iP} \cdot \vec{N}}{\overrightarrow{P_iP}}$ *{estimation of the normal curvature along* $\overrightarrow{P_iP}(6)$*}*

11: **end for**

12: *Least square sinusoidal fitting to calculate a , b and c such that*

$$\begin{bmatrix} \cos(2\alpha_1) & \sin(2\alpha_1) & 1 \\ \cos(2\alpha_2) & \sin(2\alpha_2) & 1 \\ \dots & \dots & 1 \\ \cos(2\alpha_m) & \sin(2\alpha_m) & 1 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} k_1^* \\ k_2^* \\ \dots \\ k_m^* \end{bmatrix}$$

13: *Calculate* k_{min} (11), k_{max} (12)

14: *Calculate* $\overrightarrow{T_{max}}$ (8) & (10)

15: *Calculate* $k_G = k_{min} \cdot k_{max}, k_H = \dfrac{k_{min} + k_{max}}{2} = c$

3. EXPERIMENTAL SETUP

The discussion below is detailed on three particular examples, a sphere, S_{sphere}, a right circular cylinder, $S_{cylinder}$ and a trigonometric bivariate function, $S_{trigonometric}$ defined as follows:

S_{sphere}: $x^2 + y^2 + z^2 = 2^2$ (14) $S_{cylinder}$: $x^2 + y^2 = 2^2$, $-2 \le z \le 2$ (15) $S_{trigonometric}$: $0.1(cos(x\pi) + cos(x\pi))$ (16)

The full experimentation data set covering all geometric configurations according to the surface classification given in [7], is accessible at http://dept-info.labri.fr/~charton/curvature_analysis/. The surface triangulation in use is the square split sampling defined in [13].

The proposed comparative analysis includes methods acknowledged as representative in both discrete and continuous approaches: the discrete approach formulates a closed form for differential geometry operators that works directly on the discrete representation as the method proposed by Desbrun, Meyer et al [6, 17] and abbreviated as SDA. The continuous approach includes a two stage procedure. First an entity fitting is processed: fitting of surfaces, as the simple quadratic fitting method of McIvor et al [16] abbreviated SQFA, or fitting of curves, as [4], or fitting of the curvature tensor, as [24] and [14]. Then fitted entities are "interrogated" in order to evaluate the principal curvatures and directions, the Gauss and the Mean

curvatures. We study the performances of the *SineFitting* method with respect to three criteria: the pointwise convergence, the precision and the robustness.

Pointwise convergence
Let P be a target vertex on the surface S, $B(P, r)$ a ball centred at P with radius r and $c_{B(P, r)}$ an intersection curve, $c_{B(P, r)} = S \cap B(P, r)$. Let us consider $N(P)$ with central vertex at P and neighbour vertices $\{P_i\}_{i=0}^{5}$ on $c_{B(P, r)}$. The pointwise convergence tests for the target vertex P consist in checking if the estimated values of the curvatures converge to the exact values when $r \rightarrow 0$.

Precision
The estimated curvatures are compared with the exact ones computed from the curvature formulas for implicit surfaces given in see in [11].
For S_{sphere} curvatures are constant, $k_{min} = k_{max} = k_H = 0.5$, and $k_G = 0.25$. For $S_{cylinder}$ any point not belonging to the bottom and the up circle sides has $k_{min} = 0$, $k_{max} = 0.5$, $k_H = 0.25$ and $k_G = 0$. For $S_{trigonometric}$ the symbolic computations are performed using ***Maple17***.

Robustness
In order to test robustness, we investigate four strategies for the sampling of $\{P_i\}_{i=0}^{5}$ on $c_{B(P, r)}$ inspired from the experiments of [8], [9], [10] and [14]:
The regular neighbourhood $N_{reg}(P)$ corresponds to a regular sampling around P and enables to study convergence when no degenerate triangles occur in the vicinity of the target vertex.
In the irregular neighbourhood $N_{irreg}(P)$, pairs of vertices P_i and P_{i+2}, $i=0,1,2$, are aligned. This geometric configuration corresponds to the "regular vertex" following [14]. Being less constrained than $N_{reg}(P)$, it focuses on the direction distribution and downplays the distances to the neighbours.
The regular neighbourhood with angle perturbation $N_{reg\delta Angle}(P)$ is constructed from $N_{reg}(P)$ by displacement of a single vertex P_j toward or away from its neighbours on the surface.
The regular neighbourhood with distance perturbation $N_{reg\delta Dist}(P)$ is constructed from $N_{reg}(P)$ by displacement of a single vertex P_j towards or away from P on the surface.

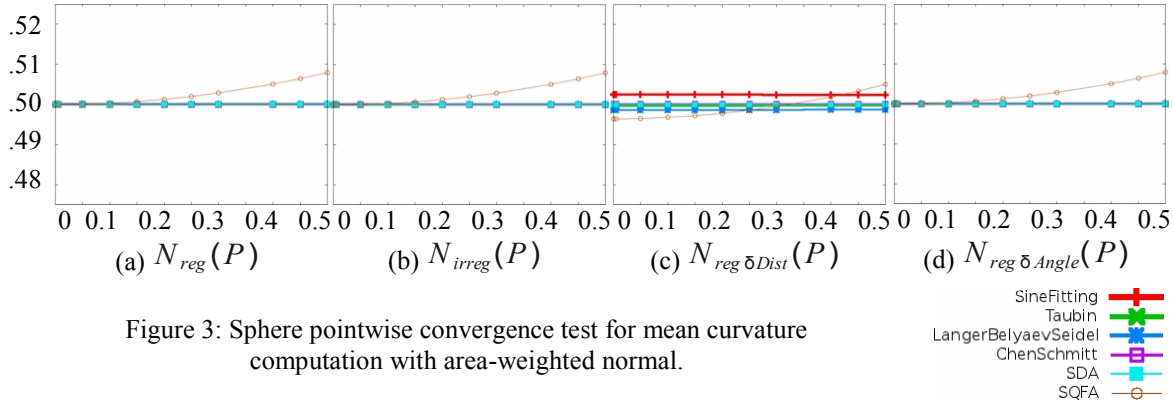


Figure 3: Sphere pointwise convergence test for mean curvature computation with area-weighted normal.

4. RESULTS

Pointwise convergence
Examples of the pointwise convergence, in $P(2, 0, 0)$ on S_{sphere} are provided in Fig.3. The vertical axis indicates the variation of the mean curvature k_H. The center of the k_H variation interval is the exact value of k_H. The horizontal axis corresponds to the radius r of $B(P, r)$. According to our experimentation all methods converge to the exact curvature values when the theoretical normal is used no matter the perturbations on the neighbourhood $N(P)$. The rates of convergence are linear except for the SQFA method.

When the normal is approximated with an area weighted normal and the neighbourhood vertices are displaced for $N_{reg\delta Dist}(P)$, see Fig.3(c), only SDA converges to the exact value.
The pointwise convergence in *P(2, 0, 0)* on $S_{cylinder}$ is shown in Fig.4 and Fig.5. For this example, all methods converge to the exact values with the theoretical normal except for the case of $N_{reg}(P)$, see Fig.4(a). The methods SQFA, Langer's and the *SineFitting* converge to the exact values when the theoretical normal is used for all types of $N(P)$. The best convergence rate is achieved by the *SineFitting* method.
The results on $S_{cylinder}$ when the area weighted normal is used do not converge to the exact values when the neighbourhood is perturbed. Moreover, the type of perturbation, depending on the distance between the target vertex and the neighbours, or on the angles adjacent to the target vertex, does not have the same impact on the algorithms. As for example, Chen's and Langer's methods are precise for k_H with angle perturbation, $N_{irreg}(P)$, while approximating k_H with $N_{reg\delta Angle}(P)$ and $N_{reg\delta Dist}(P)$. The extremely erroneous value for k_H, k_H = 0.277778, is produced by the Taubin's method on $N_{irreg}(P)$.
From a quantitative point of view, by counting the total number of successful tests, SQFA and the *SineFitting* perform better than the others. The convergence rate makes the difference: the quadratic convergence rate of the SQFA makes it less attractive, favouring the *SineFitting* method.

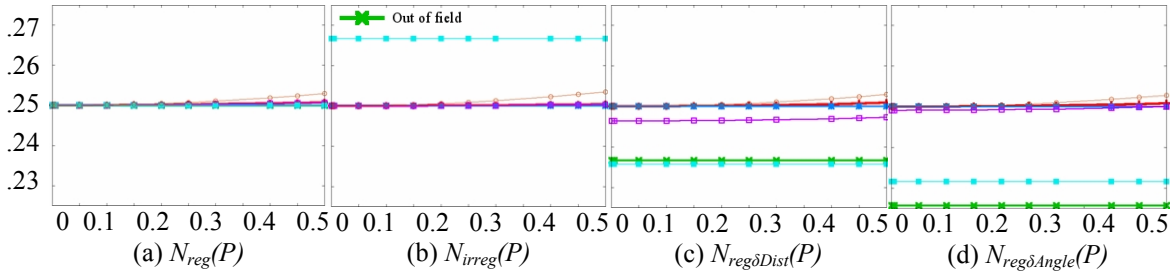


Figure 4: Cylinder pointwise convergence test for mean curvature computation with theoretical normal.

Precision
The precision is illustrated in Fig.7 with respect to the Gauss curvature evaluation on $S_{trigonometric}$. As one can see in Fig.7(b, c, f), Taubin's, Chen's and Langer's methods lack of precision in some points of inflection. The *SineFitting* method surpasses in precision the other methods: the error deviation is minimal for the different neighbourhood perturbations.

Robustness
The sensitivity of curvature estimation to the precision of the normal is illustrated in Fig.6. The regions where the estimated values are equal to the exact ones are coloured in green. With blue and red colours the regions where curvatures are respectively underestimated or overestimated are shown. All examples in Fig.6(a)-(f) use the theoretical normal while those in Fig.6(g)-(l) exploit the area-weighted normal approximation. As one can see the errors in the curvature estimation are structurally related to the quality of the underlying region triangulation, vertex valence for the poles of the sphere, and triangle quality all over the triangulation. The SDA outperforms all algorithms having similar results for both normals, the theoretical

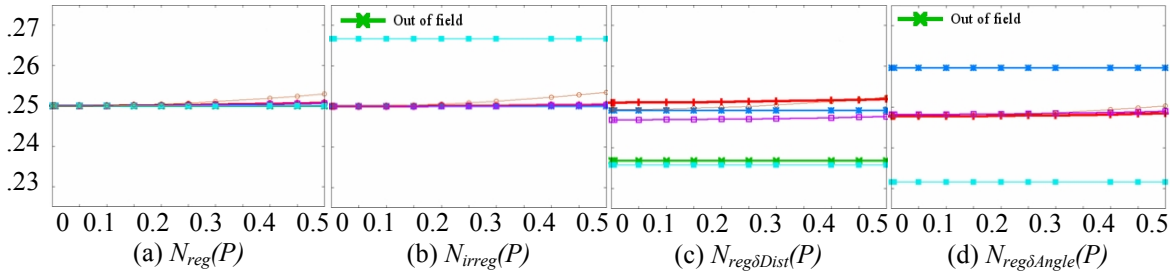
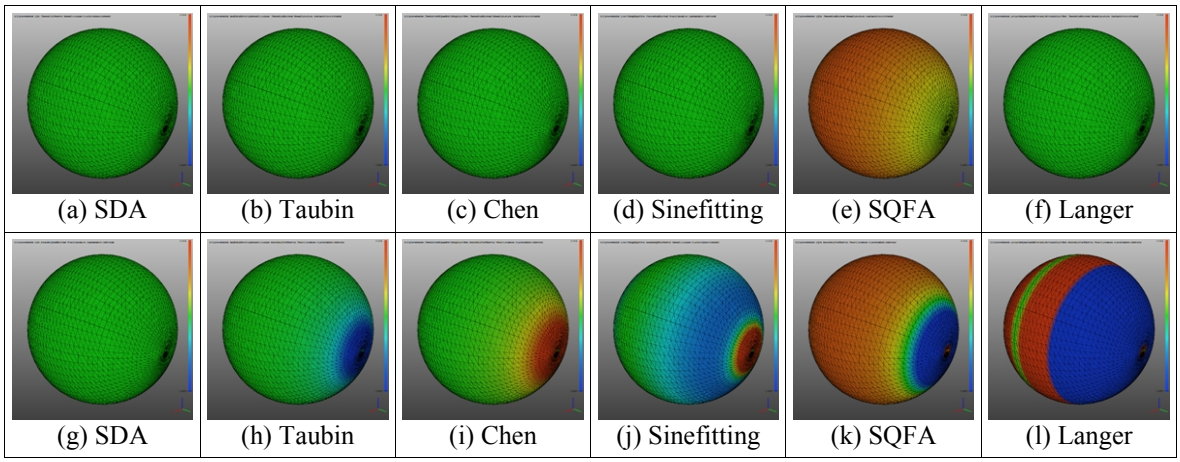


Figure 5: Cylinder pointwise convergence test for mean curvature computation with area-weighted normal.

and the area-weighted ones. SQFA and Langer's methods are the more sensitive to the normal precision as seen in Fig.6(k)(l). The proposed *SineFitting* algorithm handles the irregularities in the valence and the shape

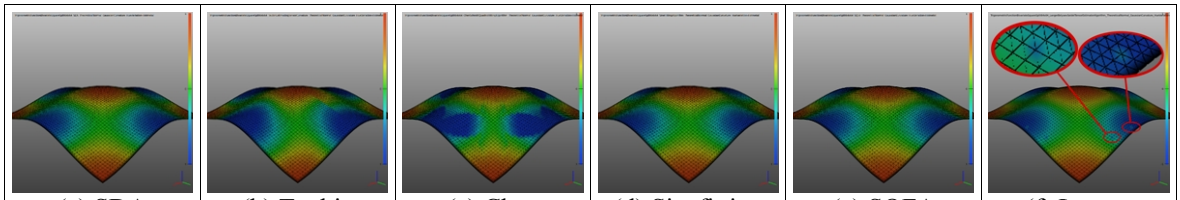
of the surface triangulation when the theoretical normal is available and remains stable when the normal is approximated.



(a) SDA (b) Taubin (c) Chen (d) Sinefitting (e) SQFA (f) Langer

(g) SDA (h) Taubin (i) Chen (j) Sinefitting (k) SQFA (l) Langer

Figure 6: Mean curvature evaluation based on theoretical and area-weighted normal



(a) SDA (b) Taubin (c) Chen (d) Sinefitting (e) SQFA (f) Langer

Figure 7: Gauss curvature evaluation based on theoretical normal

5. CONCLUSION AND PERSPECTIVES

In the present article we propose a method for the evaluation of the principal directions and curvatures, the Gauss and the Mean curvatures of surface triangulations. The elaborated method is based on a curvature *Sinefitting* algorithm and allies the advantages of the curve and surface fitting methods in processing the irregular data sampling. Three fundamental treatments are identified during the curvature evaluation at a target vertex: extraction of the neighbourhood, estimation of the normal and principal directions and curvature computation. Key attention is kept to the role of the normal estimation. A comparative analysis with other widely used methods is provided. Experimental results on chosen data set enhance the *SineFitting* performances with respect to the pointwise convergence, the precision and the robustness of the computation. One future extension of our work is to improve the normal estimation on meshes. In addition, we intend to extend our method to compute curvature on a wider scale by examining multi-ring neighbourhood. Our goal is to revisit geometric saliency and develop segmentation techniques based on precise and robust curvature estimation.

REFERENCES

[1] P.J. Besl and R. Jain. Invariant surface characteristics for 3d object recognition in range images. Computer Vision, Graphics, and Image Processing, 33(1):33-80, 1986.

[2] F. Cazals and M. Pouget. Estimating differential quantities using polynomial fitting of osculating jets. Computer Aided Geometric Design, 22:121-146, 2005.

[3] Frédéric Chazal et al. Stability of curvature measures. Comput. Graph. Forum, 28(5):1485-1496, 2009.

[4] X. Chen and F. Schmitt. Intrinsic surface properties from surface triangulation. In European Conference on Computer Vision, pages 739-743, 1992.

[5] David Cohen-Steiner and Jean-Marie Morvan. Restricted delaunay triangulations and normal cycle. In Symposium on Computational Geometry, pages 312-321, 2003.

[6] M. Desbrun et al. Discrete differential-geometry operators for triangulated 2-manifolds. In VisMath, pages 35-57. Springer-Verlag, 2002.

[7] P.J. Flynn and A.K. Jain. On reliable curvature estimation. In Computer Vision and pattern recognition, pages 110-116, 1989.

[8] R. V. Garimella and B. K. Swartz. Curvature estimation for unstructured triangulations of surfaces. Technical Report LA-UR-03-8240, Los Alamos National Laboratory, Nov 2003.

[9] T. Gatzke et al. Curvature maps for local shape comparison. In SMI, pages 246-255, 2005.

[10] T. Gatzke and C. M. Grimm. Estimating curvature on triangular meshes. International Journal of Shape Modeling, 12(1):1-28, 2006.

[11] R. Goldman. Curvature formulas for implicit curves and surfaces. Computer Aided Geometric Design, 22(7):632-658, 2005.

[12] X. Guoliang. Convergence analysis of a discretization scheme for gaussian curvature over triangular surfaces. Computer Aided Geometric Design, 23(2):193-207, 2006.

[13] B. Hamann. Curvature approximation for triangulated surfaces. In Geometric Modelling, pages 139-153, 1992.

[14] T. Langer et al. Exact and interpolatory quadratures for curvature tensor estimation. Computer Aided Geometric Design, 24(8-9):443-463, 2007.

[15] J.-L. Maltret and M. Daniel. Discrete curvatures and applications : a survey. Rapport de recherche LSIS.RR.2002.002, Laboratoire des Sciences de l'Information et des Systèmes, 2002.

[16] A.M. McIvor and R.J. Valkenburg. A comparison of local surface geometry estimation methods. Mach. Vis. Appl., 10(1):17-26, 1997.

[17] Mark Meyer. Discrete differential operators for computer graphics. PhD thesis, Pasadena, CA, USA, 2004. AAI3290476.

[18] J.-M. Morvan. Generalized Curvatures. Springer Series in Geometry and Computing, 2008.

[19] S. Petitjean. A survey of methods for recovering quadrics in triangle meshes. ACM Computing Surveys, 2:1-61, 2002.

[20] S. Rusinkiewicz. Estimating curvatures and their derivatives on triangle meshes. In 2nd Int. Symp. on 3D Data Processing, Visualization and Transmission (3DPVT 2004), 6-9 September 2004, Thessaloniki, Greece, pages 1-8. IEEE Computer Society, 2004.

[21] P.T. Sander. Generic curvature features from 3-d images. IEEE Transactions on systems, man, and cybernetics, 19(6):1623-1636, 1989.

[22] M. Spivak. A Comprehensive Introduction to Differential Geometry. Publish or Perish, inc., 1970.

[23] T. Surazhsky et al. A comparison of gaussian and mean curvatures estimation methods on triangular meshes. In ICRA, pages 1021-1026, 2003.

[24] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In ICCV, pages 902-907, 1995.

[25] H. Theisel et al. Normal based estimation of the curvature tensor for triangular meshes. In Pacific Conference on Computer Graphics and Applications, pages 288-297, 2004.