Proceedings of the 2003 IEEE
International Conference on Robotics & Automation
Taipei, Taiwan, September 14-19, 2003

A Comparison of Gaussian and Mean Curvatures Estimation Methods on Triangular Meshes

Tatiana Surazhsky*, Evgeny Magid†, Octavian Soldea‡,
Gershon Elber§ and Ehud Rivlin¶

Center for Graphics and Geometric Computing,
Technion, Israel Institute of Technology, Haifa 32000, Israel.

ABSTRACT

Estimating intrinsic geometric properties of a surface from a polygonal mesh obtained from range data is an important stage of numerous algorithms in computer and robot vision, computer graphics, geometric modeling, industrial and biomedical engineering. This work considers different computational schemes for local estimation of intrinsic curvature geometric properties. Five different algorithms and their modifications were tested on triangular meshes that represent tesselations of synthetic geometric models. The results were compared with the analytically computed values of the Gaussian and mean curvatures of the non uniform rational B-spline (NURBs) surfaces, these meshes originated from. This work manifests the best algorithms suited for total (Gaussian) and mean curvature estimation, and shows that indeed different alogrithms should be employed to compute the Gaussian and mean curvatures.

Keywords

Geometric modeling, principal curvatures, Gaussian curvature, total curvature, mean curvature, polygonal mesh, triangular mesh, range data

1. INTRODUCTION

A number of approaches have been proposed to represent a 3D object for the purposes of reconstruction, recognition, and identification. The approaches are generally classified into two groups: a volumetric and a boundary-based method. A volumetric description utilizes global characteristics of a 3D object : principal axes, inertia matrix [8], tensor-based moment functions [5] etc. Boundary-based methods describe an object based on distinct local properties of the boundary and their relationship. This method is well suited for recognition purposes because local properties are still available with only partial view of an object.

The differential invariant properties such as total (Gaussian) and mean curvatures are one of the most essential features in boundary-based methods, extensively used for segmentation, recognition and registration algorithms [17, 19]. Regrettably, these significant geometric quantities are defined only for twice differentiable (C^2) surfaces. In contrast, geometric data sets are frequently available as polygonal, piecewise linear, approximations, typically as triangular meshes. Such data sets are common output of, for examples, 3D scanners.

There exist a whole plathora of work [1,2,4,6,7,9–14,16,18] describing algorithms for curvature estimation from polygonal surfaces. Unfortunately, no rigorous comparison of these methods exists, nor any errors analysis for these different algorithms ever provided. The reason for the absence of such work might be found in the difficulty of comparing the computed curvature values with the exact curvature information.

In [9,14] the authors furnish us with the general overview of three algorithms on four types of primitive surfaces: spheres, planes, cylinder and trigonometric surfaces. No general surface are considered.

In [12], there is an asymptotic analysis of the paraboloid fitting scheme and an algorithm based on Gauss-Bonnet theorem [3, 15], we refer to as the Gauss-Bonnet scheme, that also known as the *angle deficit* method [12]. Yet again, no error analysis for general surfaces is provided.

In [6, 10, 14], the principal curvatures and principal directions of a triangulated surface is estimated at each vertex by a least square fitting of an osculating paraboloid to the vertex and its neighbors. These references use linear approximation methods where the approximated surface is obtained by solving an over-determined system of linear equations.

In [13], the authors present a nonlinear functional minimization algorithm that is implemented as an iterative con-

*Applied Mathematics Department,
E-mail: tess@cs.technion.ac.il
†Applied Mathematics Department,
E-mail: evgenue@cs.technion.ac.il
‡Faculty of Computer Science,
E-mail: octavian@cs.technion.ac.il
§Faculty of Computer Science,
E-mail: gershon@cs.technion.ac.il
¶Faculty of Computer Science,
E-mail: ehudr@cs.technion.ac.il

straint satisfaction procedure based on local surface smoothness properties.

In [2,11], circular cross sections, near the examined vertex, are fitted to the surface. Then, the principal curvatures are computed using Meusnier's and Euler's theorems (see [3, 15]).

In [16], the principal curvatures are estimated with the aid of an eigenvalues/vectors analysis of 3×3 symmetric matrices that are defined via an integral formulation.

In this work, we attempt to quantitatively compare five of the above methods for estimating the total and mean curvature, for triangular meshes. We test these methods on triangular meshes that represent tesselations of the non uniform rational B-spline (NURBs) surfaces. The results are compared to the analytic evaluation of these curvature properties on the NURBs surfaces.

The paper is organized as follows: a short review of differential geometry of surfaces is provided in Section 2. In Section 3, a concise description of the five algorithms we considered is given. Then, results of our comparison could be found in Section 4. Finally, we conclude in Section 5.

2. DIFFERENTIAL GEOMETRY OF SURFACES

Let $S(r,t)$ be a regular C^2 continuous freeform parametric surface in $\boldsymbol{R}^3$. The unit normal vector field of $S(r,t)$ is defined by

$$N(r,t) = \frac{\frac{\partial}{\partial r}S(r,t) \times \frac{\partial}{\partial t}S(r,t)}{\left\| \frac{\partial}{\partial r}S(r,t) \times \frac{\partial}{\partial t}S(r,t) \right\|}.$$

The *normal curvature* κ_n of curve $C \subset S$ passing through the point $S(r_0,t_0)$ is defined by the following relation, known as

Theorem 2.1 Meusnier's theorem:

$$\kappa_n = \kappa \cos\varphi \tag{2.1}$$

where κ is the curvature of C at $S(r_0,t_0)$ and φ is the angle between the curve's normal n and the normal $N(r_0,t_0)$ of S.

The principal curvatures, $\kappa_1(r_0,t_0)$ and $\kappa_2(r_0,t_0)$, of S at $S(r_0,t_0)$ are defined as the maximum and the minimum normal curvatures at $S(r_0,t_0)$, respectively. The directions of the tangents of the two curves that are the result of the intersection of the surface $S(r_0,t_0)$ and the planes containing $N(r_0,t_0)$ and having the curvature values of $\kappa_1(r_0,t_0)$ and $\kappa_2(r_0,t_0)$ are denoted the *principal directions* (see [3,15]).

Theorem 2.2 Euler's theorem: *The normal curvature κ_n of surface $S(r,t)$ in tangent direction T is equal to:*

$$\kappa_n = \kappa_1 \cos^2\theta + \kappa_2 \sin^2\theta \tag{2.2}$$

where θ is the angle between the first principal direction and T.

The total and mean curvatures, $K(r,t)$ and $H(r,t)$, are uniquely defined by the principal curvatures of the surface:

$$K(r,t) = \kappa_1(r,t) \cdot \kappa_2(r,t); \tag{2.3}$$

$$H(r,t) = \frac{\kappa_1(r,t) + \kappa_2(r,t)}{2}. \tag{2.4}$$

3. ALGORITHMS FOR CURVATURE ESTIMATION

In this work, we consider five methods for the estimation of the principal curvatures, for triangular meshes. We assume that the given triangular mesh approximates a smooth, at least twice differentiable, surface.

3.1 Paraboloid Fitting

This algorithm approximates a small neighborhood of the mesh around a vertex v by an osculating paraboloid. The principal curvatures of the surface are considered to be identical to the principal curvatures of the paraboloid (see [6,10, 14]).

Vertex v_i is considered an immediate neighbor of vertex v if edge $e_i = \overline{v\,v_i}$ belongs to the mesh. Denote the set of immediate neighboring vertices of v by $\{v_i\}_{i=0}^{n-1}$ and the set of the triangles containing the vertex v by $\{T_i^v\}_{i=0}^{n-1}$,

$$T_i^v = \triangle(v_i\ v\ v_{(i+1) \bmod n}), \quad 0 \le i \le n-1. \tag{3.1}$$

Let N_v be the normal of surface S at vertex v. Normals are, in many cases, provided with the mesh toward Gouraud and/or Phong shading. Otherwise, let

$$N_i^v = \frac{(v_i - v) \times (v_{(i+1) \bmod n} - v)}{\left\|(v_i - v) \times (v_{(i+1) \bmod n} - v)\right\|}. \tag{3.2}$$

be the unit normal of triangle T_i^v. Then, N_v could be estimated as an average of normals N_i^v:

$$\overline{N_v} = \frac{1}{n}\sum_{i=0}^{n-1} N_i^v; \quad N_v = \frac{\overline{N_v}}{\left\|\overline{N_v}\right\|}. \tag{3.3}$$

v is now transformed along with its immediate neighboring vertices, $\{v_i\}_{i=0}^{n-1}$, to the origin such that N_v coalesce with the z axes. Assume an arbitrary direction x (and $y = z \times x$). Then, the osculating paraboloid of this canonical form equals,

$$z = ax^2 + bxy + cy^2. \tag{3.4}$$

The coefficients a, b and c are found by solving a least square fit to v and the neighboring vertices $\{v_i\}_{i=0}^{n-1}$. Then, the total and mean curvatures are computed as,

$$K = 4ac - b^2; \quad H = a + c. \tag{3.5}$$

3.2 Circular Fitting

This algorithm from [2, 11] uses the Meusnier's 2.1 and Euler's 2.2 theorems for the estimation of the principal curvatures. The justification for this algorithm comes from Equation (2.2) that is equivalent to

$$\kappa_n = \frac{1}{2}(\kappa_1 + \kappa_2) - \frac{1}{2}(\kappa_1 - \kappa_2)(\cos 2\theta_0 \cos 2\alpha + \sin 2\theta_0 \sin 2\alpha), \tag{3.6}$$

where θ_0 is the angle between some arbitrary chosen reference direction T_0 in the tangent plane of vertex v and the principal direction that corresponds to κ_1. α is the angle between the tangent direction corresponding to the normal curvature κ_n and the reference direction T_0. We can rewrite Equation (3.6) as

$$\kappa_n = A - B\cos 2\alpha + C \sin 2\alpha. \tag{3.7}$$

One can derive the values of A, B and C using a least squares fit of at least three circles, where each one passes

through point $v = S(r_0, t_0)$, and two of v's immediate neighbors. Then, the principal curvatures and principal directions could be computed as:

$$\begin{aligned} \kappa_1 &= A + \sqrt{B^2 + C^2}, \\ \kappa_2 &= A - \sqrt{B^2 + C^2}, \\ \theta_0 &= \frac{1}{2} arctan\left(\frac{C}{B}\right). \end{aligned} \tag{3.8}$$

The method proposed in [2, 11] constructs the fitted circles through the vertex v and a pair of v's neighbors, v_i and v_j, such that the angle between vectors $(v_i - v)$ and $(v_j - v)$ is close to π. Thus, by selecting $k \geq 3$ pairs of such neighbors, the k constructed circles can prescribe using Meusnier's theorem (see Equation (2.1)) the values of A, B, C, and consequently, κ_1 and κ_2.

3.3 The Gauss-Bonnet Scheme

Consider again vertex v and its immediate neighborhood $\{v_i\}_{i=0}^{n-1}$. Then, for $i = 0 \ldots n-1$, let $\alpha_i = \angle(v_i, v, v_{(i+1) \bmod n})$ be the angle at v between two successive edges $e_i = \overline{v\, v_i}$. Further, let $\gamma_{i+1} = \angle(v_i, v_{(i+1) \bmod n}, v_{(i+2) \bmod n})$ be the outer angle between two successive edges of neighboring vertices of v. Then, simple trigonomy can show that

$$\sum_{i=0}^{n-1} \alpha_i = \sum_{i=0}^{n-1} \gamma_i. \tag{3.9}$$

The Gauss-Bonnet [3, 15] theorem reduces, in the polygonal case, to

$$\iint_A K dA = 2\pi - \sum_{i=0}^{n-1} \gamma_i, \tag{3.10}$$

which, by Equation (3.9) equals to,

$$\iint_A K dA = 2\pi - \sum_{i=0}^{n-1} \alpha_i, \tag{3.11}$$

where A is the accumulated areas of triangles T_i^v (Equation (3.1)) around v.

Assuming K is constant in the local neighborhood, Equation (3.11) can be rewritten as

$$K = \frac{2\pi - \sum_{i=0}^{n-1} \alpha_i}{\frac{1}{3}A}. \tag{3.12}$$

This approach for estimating K is used, for example, by [1, 4, 7, 12, 14]). In [4, 7] a similar intergal approach to the computation of the mean curvature is also proposed as

$$H = \frac{\frac{1}{4}\sum_{i=0}^{n-1} \|e_i\| \beta_i}{\frac{1}{3}A}, \tag{3.13}$$

where $\|e_i\|$ denotes the magnitude of e_i and β_i measures normal deviations $\beta_i = \angle(N_i^v, N_{(i+1) \bmod n}^v)$ (see Equation (3.2)).

3.4 The Watanabe and Belyaev Approach

A simple method for estimating the principal curvatures of a surface that is approximated by a triangular mesh was proposed in [18].

Consider an oriented surface S. Let T be a tangent vector and N be the unit normal at a surface point P. A normal section curve $r(s)$ associated with T at P is defined as the intersection between the surface and the plane through P that is spanned by T and N. Let T_1 and T_2 be the principal directions at P associated with principal curvatures κ_1 and κ_2, respectively. $\kappa_n(\varphi)$ denotes the normal curvature of the normal section curve, where φ is the angle between T and T_1. Using Euler's theorem (See Theorem 2.2), integral formulas of $\kappa_n(\varphi)$ and its square are derived:

$$\frac{1}{2\pi}\int_0^{2\pi} k_n(\varphi) d\varphi = H; \qquad \frac{1}{2\pi}\int_0^{2\pi} k_n(\varphi)^2 d\varphi = \frac{3}{2}H^2 - \frac{1}{2}K. \tag{3.14}$$

In order to estimate the integrals of Equation (3.14), one needs to estimate the normal curvature around v, in all possible tangent directions.

Assume P coalesces with mesh vertex v and recall the normal N_v of v (Equation (3.3)). Here, the average of the normals of the adjacent faces to vertex v takes into account the relative areas of the different faces.

v is now transformed along with its immediate neighboring vertices, $\{v_i\}_{i=0}^{n-1}$, to the origin such that N_v coalesces with the z axes. Consider the intersection curve $r = r(s)$ of the surface by a plane through v that is spanned by N_v (the z axis in our canonical form) and edge $e_i = \overline{v\, v_i}$. A Taylor series expansion of $r(s)$ gives.

$$\begin{aligned} r(s) &= r(0) + s r'(0) + \frac{s^2}{2} r''(0) + \ldots, \\ &= r(0) + s T_r + \frac{s^2}{2} \kappa_n N_r + \ldots, \end{aligned} \tag{3.15}$$

where T_r and N_r are the unit tangent and normal of $r(s)$. Recall that $v = r(0)$ and that $v_i = r(s)$. The arclength s could be approximated by the length of edge $e_i = \overline{v\, v_i}$, or $s \approx \|\overline{v\, v_i}\|$. Multiplying Equation (3.15) by $N_v = N_r$ yields,

$$N_v \cdot \overline{v\, v_i} \approx \kappa_n \frac{\|\overline{v\, v_i}\|^2}{2}; \qquad \kappa_n \approx \frac{2 N_v \cdot \overline{v\, v_i}}{\|\overline{v\, v_i}\|^2}. \tag{3.16}$$

The trapezoid approximation of Equation (3.14) leads to

$$2\pi H \approx \sum_{i=0}^{n-1} \kappa_n^i \left(\frac{\varphi_i + \varphi_{(i+1) \bmod n}}{2}\right), \tag{3.17}$$

and

$$2\pi\left(\frac{3}{2}H^2 - \frac{1}{2}K\right) \approx \sum_{i=0}^{n-1} {k_n^i}^2 \left(\frac{\varphi_i + \varphi_{(i+1) \bmod n}}{2}\right). \tag{3.18}$$

3.5 The Taubin Approach

Let T_1 and T_2 be the two principal directions at point P of surface S and let $T_\theta = \cos(\theta) T_1 + \sin(\theta) T_2$ be some unit length tangent vector at P. Taubin, in [16], defines the symmetric matrix $\mathbf{M_p}$ by the integral formula of

$$\mathbf{M_p} = \frac{1}{2\pi}\int_{-\pi}^{+\pi} \kappa_n^p(T_\theta) T_\theta T_\theta^t d\theta. \tag{3.19}$$

where $\kappa_n^p(T_\theta)$ is the normal curvature of S at P in the direction T_θ.

Since the unit length normal vector N to S at P is an eigenvector of $\mathbf{M_p}$ associated with the eigenvalue zero, after the factorization,

$$\mathbf{M_p} = \mathbf{T_{12}}^t \begin{pmatrix} m_p^{11} & m_p^{12} \\ m_p^{21} & m_p^{22} \end{pmatrix} \mathbf{T_{12}} \tag{3.20}$$

where $\mathbf{T_{12}} = [T_1, T_2]$ is the 3×2 matrix constructed by concatenating the column vectors T_1 and T_2. The principal curvatures can then be obtained as functions of the nonzero eigenvalues of $\mathbf{M_p}$:

$$k_1 = 3m_p^{11} - m_p^{22}, \qquad k_2 = 3m_p^{11} - m_p^{22}. \tag{3.21}$$

The first step of the implementation estimates the normal vector N_v at each vertice v of the surface with the help of the Equation (3.2). Then, for each vertex v, matrix $\mathbf{M_v}$ is approximated with a weighted sum over the neighbor vertices, v_i:

$$\tilde{M}_v = \sum_{i=0}^{n-1} w_i \kappa_n(T_i) T_i T_i^t, \tag{3.22}$$

where

$$T_i = \frac{(I - N_v N_v^t)(v - v_i)}{\|(I - N_v N_v^t)(v - v_i)\|} \tag{3.23}$$

is the unit length normalized projection of vector $v_i - v$ onto the tangent plane $\langle N_v \rangle^\perp$. The normal curvature indirection T_i is approximated with the help of Equation (3.16) as $\kappa_n(T_i) = \frac{2N_v^t(v_i - v)}{\|v_i - v\|^2}$.

The weights, w_i, are selected to be proportional to the sum of the surface areas of the triangles incident to both vertices v and v_i (two triangles if the surface is closed, and one triangle if both vertices belong to the boundary of S).

By construction, the normal vector N_v is an eigenvector of the matrix $\tilde{M}_v$ associated with the eigenvalue zero. Then, $\tilde{M}_v$ is restricted to the tangent plane $\langle N_v \rangle^\perp$ and, using a Householder transformation and Givens rotation the remaining eigenvectors T_1 and T_2 of $\tilde{M}_v$ (i.e., the principal directions of the surface at v) are computed. Finally, the principal curvatures are obtained from the two corresponding eigenvalues of $\tilde{M}_v$ using Equation (3.21).

3.6 Modifications

We considered two modifications of the Taubin's algorithm [16]:

- *Taubin A, (Constant integration):* In Equation (3.22), the weights w_i are selected to be proportional to angles $\angle(v_i, v, v_{i+1})$ instead of the surface areas.
- *Taubin B, (Smoothing with a trapezoidal rule):* The directional curvature $\kappa_n(T_i)$ in Equation (3.22) is selected as an average of values $\kappa_n(T_i)$ and $\kappa_n(T_{i+1})$.

We also considered two modifications to the algorithm of Watanabe and Belyaev [18]. In the two following modificated algorithms, the first step was the same. We employ the tangent directions T_i at vertex P produced by Equation (3.23). Having two vertices (v and v_i), tangent direction T_i and the normal in P, we compute the radius of the fitted circle and from that derive $\kappa_n(\varphi_i)$, the normal curvature in the specified direction T_i.

- *Watanabe A:* Having the normal curvatures, we apply Equations (3.17) and (3.18).
- *Watanabe B:* From the set of the normal curvatures of each vertex v, $\{\kappa_n(\varphi_i)\}_{i=0}^{n-1}$, we select the maximal (k_1) and the minimal (k_2) normal curvature values and apply the classic Equations of (2.3) and (2.4).

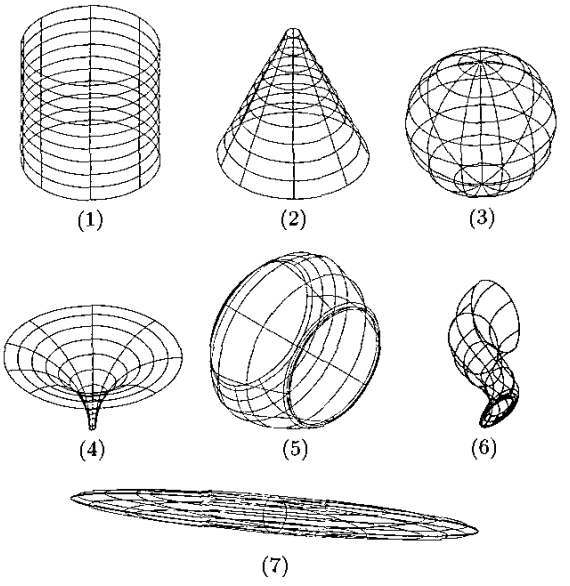

Figure 1: The NURB surfaces that were used for curvature estimation tests.

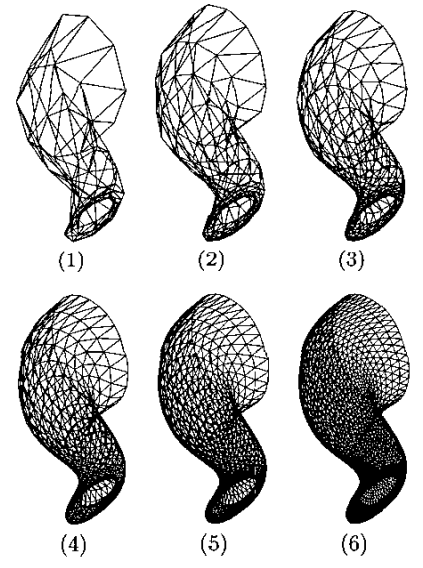

Figure 2: The tesselations of the spout surface were produced for the following resolutions: 128 triangles (1), 288 triangles (2), 512 triangles (3), 1152 triangles (4), 2048 triangles (5) and 5000 triangles (6).

4. EXPERIMENTAL RESULTS

We tested all the algorithms described in Section 3 on a set of synthetic models that represent the tesselations of seven NURB surfaces: a cylinder, a cone, a sphere, a surface of revolution generated by a non circular arc, the body and the spout of the infamous Utah teapot model and ellipsoid,

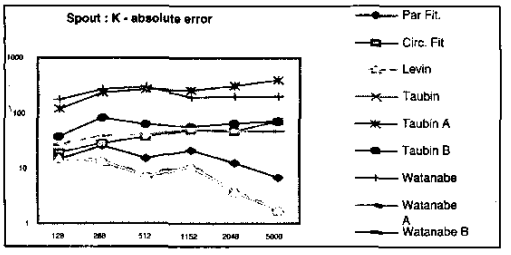

Figure 3: Average of the absolute error for the value of total curvature for the tesselations of the surface of revolution generated by an arc (see Figure 1 (4)).

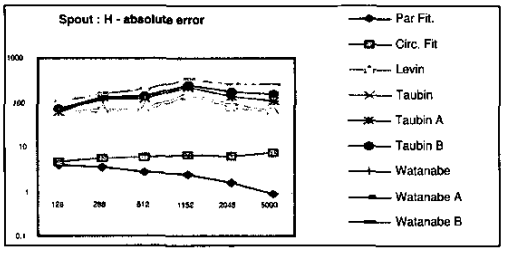

Figure 4: Average of the absolute error for the value of mean curvature for the tesselations of the surface of revolution generated by an arc (see Figure 1 (4)).

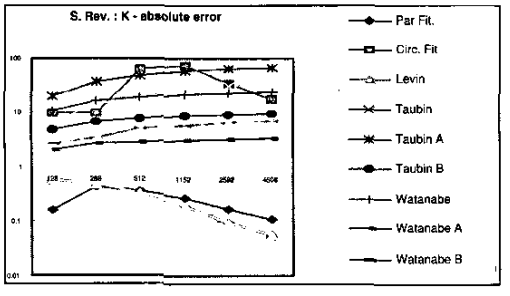

Figure 5: Average of the absolute error for the value of total curvature for the tesselations of the Utah teapot's spout NURBs surface (see Figures 1 (6) and 2).

see Figure 1. These models along with the curvature values for each vertex are also available in

http://www.cs.technion.ac.il/~gershon/poly_crvtr.

The tesselations of each model were produced for several different resolutions: from about a hundred triangles to several thousand triangles for the finest resolution. The different tesselations of the spout surface are shown in Figure 2. These different resolutions helped us gain some insight into the convergence rate of the different examined algorithms as the accuracy of the tesselation is improved.

For all the tested models, the comparison between the different algorithms showed similar results. Denote by $\overline{K_i}$ and $\overline{H_i}$ the values of total and mean curvatures computed by

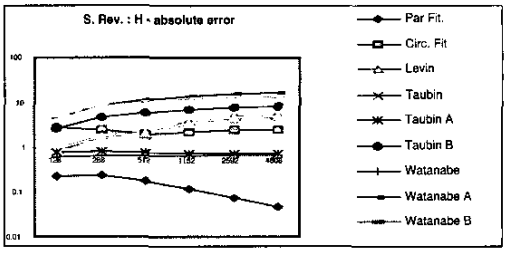

Figure 6: Average of the absolute error for the value of mean curvature for the tesselations of the Utah teapot's spout NURBs surface (see Figures 1 (6) and 2).

one of the methods from the triangular mesh data in vertex v_i, while K_i and H_i are the exact (analytically computed) values of the total and mean curvatures, in the same surface location $v_i = S(r_i, t_i)$ on the corresponding NURBs surface. We considered the following error values:

1. Average of the absolute error value of the total curvature K

$$\frac{1}{m}\sum_{i=1}^{m}|K_i - \overline{K_i}|,$$

2. Average of the absolute error value of the module of the mean curvature $|H|$

$$\frac{1}{m}\sum_{i=1}^{m}||H_i| - |\overline{H_i}||,$$

In the vast majority of previous results, only primitives such as cones and sphere were examined for the accuracy of these curvature approximation algorithms.

In all our tests, two algorithms out performed the rest. It was clear thoughout our tests that the parabolic fitting is the best scheme to derive the mean curvature. Further the Gauss-Bonnet (angle deficit) scheme always provided the best results for the total curvature.

Figures 3 and 4, show the results of the tests for the tesselations of the surface of revolution generated by a non circular arc (see Figure 1 (4)), whereas Figures 5 and 6 demonstrate the comparison of the curvature estimation algorithms for the tesselations of the spout of the Utah teapot (see Figures 1 (6) and 2).

These graphs shows few examples of the results we got throughout our tests. The Gauss-Bonnet scheme shines when K is computed and the parabolic fitting scheme works best for H. Hence, the *optimal approximation scheme for triangular meshes should be based on a synergy of two schemes.*

Another significant result that can be drawn from these graphs is that this synergetic scheme converges as the fineness of the mesh is improved. This convergance was not witnessed by all schemes, yet both the Gauss-Bonnet scheme for K and the parabolic fitting scheme for H always did converge.

Interestingly enough, for K the parabolic fitting scheme was continuously second and followed a stride behind the Gauss-Bonnet scheme. If one must select only one method of choice, it should therefore be the parabolic fitting scheme.

5. CONCLUSION

In this work, we have provided a comparison of five different algorithms for curvature estimation over triangular meshes.

We have built a library of triangular meshes that represent approximations (with different resolutions) of the synthetic models given as NURB surfaces. For each surface we have produced several polyhedral approximation with varying number of the triangles. The library files contain the following information for each vertex v_i:

- 3D coordinates;
- Analytically precomputed values of total curvature K_i and squared value of the mean curvature H_i^2 at point $v_i = S(r_i, t_i)$, computed from the original NURBs surface.

The output of the tests of five different schemes on seven models (Figure 1) shows that:

1. The best algorithm for the estimation the total curvature is the *Gauss-Bonnet scheme.*
2. The best method for the estimation of the mean curvature is the *paraboloid fit method.*

The most stable method that always had a good convergence is that osculating paraboloid fitting scheme. In [12], the authors proved that the paraboloid fit method has a quadratic error bound using asymptotic analysis even for non-uniform meshes (that is usually the case in practice), while the Gauss-Bonnet scheme has quadratic error bound in the case of uniform mesh and linear error bound otherwise.

Acknowledgments

This work was partially supported by CONSIST project.

6. REFERENCES

[1] L. Alboul and R. van Damme. Polyhedral metrics in surface reconstruction: tight triangulations. In *University of Twente, Dep. of Applied Mathematics, Technical Report*, pages 309-336, 1995.

[2] X. Chen and F. Schmitt. Intrinsic surface properties from surface triangulation. In *Proc. 2nd European Conf. on Computer Vision, (ed) G. Sandini*, pages 739– 743. St. Margherita Ligure, Italy, May 1992.

[3] M. DoCarmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall, 1976.

[4] N. Dyn, K. Hormann, S.-J. Kim, and D. Levin. Optimizing 3d triangulations using discrete curvature analysis. In T. Lyche and L. L. Schumaker, editors, *Mathematical Methods in CAGD: Oslo 2000 Nashville, TN, 2001. Vanderbilt University Press*, pages 135–146, 2001.

[5] T. L. Faber and E. Stokely. Orientation of 3d structures in medical images. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 10, pages 626-633, September 1988.

[6] B. Hamann. Curvature approximation for triangulated surfaces. In *Computing Suppl.*, number 8, pages 139–153, 1993.

[7] S. J. Kim, C.-H. Kim, and D. Levin. Surface simplification with discrete curvature norm. In *The Third Israel-Korea Binational Conference on geometric Modeling and Computer graphics, Seoul, Korea.*, October 2001.

[8] Y. Kim and R. Luo. Validation of 3d curved objects: Cad model and fabricated workpiece. In *IEEE Transactions on Industrial Electronics*, volume 41(1), pages 125– 131, February 1994.

[9] P. Krsek, C. Lukács, and R. R. Martin. Algorithms for computing curvatures from range data. In *The Mathematics of Surfaces VIII, Information Geometers*, pages 1–16. A. Ball et al. Eds., 1998.

[10] P. Krsek, T. Pajdla, and V. Hlavác. Estimation of differential parameters on triangulated surface. In *21st Workshop of the Austrian Association for Pattern Recognition*, May 1997.

[11] R. Martin. Estimation of principal curvatures from range data. *International Journal of Shape Modeling*, 4:99–111, 1998.

[12] D. Meek and D. Walton. On surface normal and gaussian curvature approximations given data sampled from a smooth surface. *Computer Aided Geometric Design*, (17):521–543, 2000.

[13] P. Sanders and S. Zucker. Inferring surface trace and differential structure from 3d images. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 12(9), pages 833– 854, September 1990.

[14] E. Stokely and S. Y. Wu. Surface parameterization and curvature measurement of arbitrary 3d-objects: Five practicel methods. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 14(8), pages 833–840, August 1992.

[15] D. Struik. *Lectures on Classical Differentional Geometry.* Addison-Wesley Series in Mathematics, 1961.

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

[17] B. Vemuri, A. Mitiche, and J. Aggarwal. Curvature-based representation of objects from range data. In *Image Vision Comput.*, volume 4, pages 107–114, May 1986.

[18] K. Watanabe and A. G. Belyaev. Detection of salient curvature features on polygonal surfaces. In *EUROGRAPHICS 2001*, volume 20, 2001.

[19] N. Yakoya and M. Levine. Range image segmentation based on differential geometry: A hybrid approach. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 11(6), pages 643-649, June 1989.