A Survey on Differential Geometry

Thitiwan Srinark, Ph.D.
Computer Engineering Department
Faculty of Engineering, Kasetsart University

Abstract

In this paper, we survey the basic differential geometry of surfaces which include shape operator, first fundamental form, second fundamental form and curvatures. We also demonstrate how some significant equations are derived. We explain various practical techniques for curvature estimation on different surface representations such as parametric surfaces, implicit surface and polygon mesh surfaces. Finally we present a surface classification of point.

1 Differential Geometry on Surface

Let M be an orientable surface embedded in the three-dimensional space $\mathcal{R}^3$ such that M is described by an arbitrary parameterization of two variables. For each point p on the surface M, the surface can be approximated by its tangent plane, that is orthogonal to the normal vector N. Curvatures are defined to measure local bending of the surface. At a point p on the surface, for every unit direction u, the normal curvature $k(u)$ is defined as the curvature of the curve that belongs to both the surface itself and a perpendicular plane containing both N and u. Formally, the normal curvature is defined by [1]

$$k(\mathsf{u}) = S_{\mathsf{p}}(\mathsf{u}) \cdot \mathsf{u}, \quad (1)$$

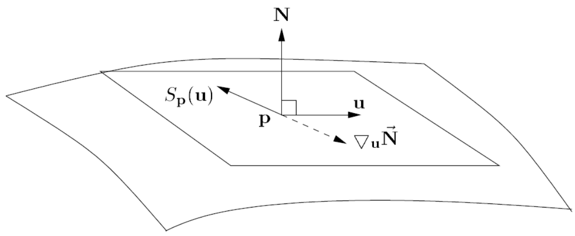


Figure 1. The shape operator at a point p

where S_{p} denotes the shape operator. Note that Eq. 1 can be derived by the following.

$$\begin{aligned} k(\mathsf{u}) &= S_{\mathsf{p}}(\mathsf{u}) \cdot \mathsf{u} \\ &= S_{\mathsf{p}}(-\mathsf{u}) \cdot (-\mathsf{u}) \\ &= k(-\mathsf{u}) \end{aligned} \quad (2)$$

The shape operator S_{p} of the surface M at a point p with u as a tangent vector is defined as

$$S_{\mathsf{p}}(\mathsf{u}) = - \nabla_{\mathsf{u}} \mathsf{N}, \quad (3)$$

where N is a unit normal vector field on a neighborhood of p in M. $\nabla_{\mathsf{u}}\mathsf{N}$ is considered the rate of change of N in the u direction, in other words, it tells how the tangent planes of M are varying in the u direction. Figure 1 illustrates the relationship between the shape operator, normal, and tangent at the point p on the surface.

The maximum and minimum values of the normal curvature $k(\mathsf{u})$ of M at p are called the ``principal curvatures,'' and are denoted by k_1 and k_2. The

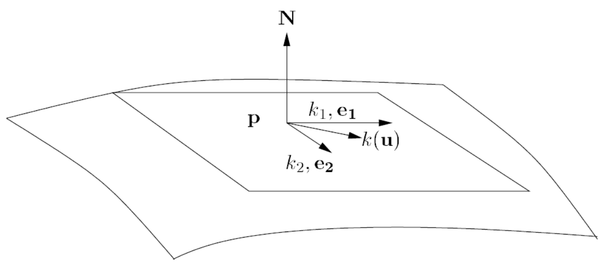


Figure 2. Normal curvature, and principal curvatures at a point p**.**

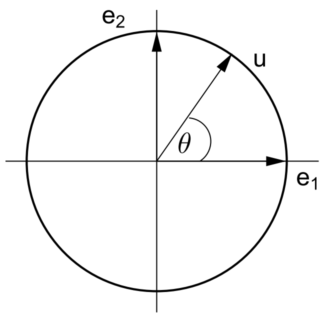


Figure 3. Two principal vectors, e_1 **and** e_2**, and the tangent vector** u**.**

associated orthogonal directions of k_1 and k_2 are called ``principal directions.'' Unit vectors in these directions are called ``principal vectors,'' denoted by $\mathbf{e}_1$ and $\mathbf{e}_2$. Figure 2 shows the normal curvature, and the two principal curvatures at the point **p**. From the Euler's theorem, the normal curvature of M in the **u** direction is

$$k(\mathbf{u}) = k_1 \cos^2 \theta + k_2 \sin^2 \theta. \tag{4}$$

Figure 3 shows how the principal vectors and the tangent vector are related.

From [1], a point **p** of M is ``umbilic,'' if its normal curvature $k(\mathbf{u})$ is constant on all unit tangent vectors **u** at **p**. Then the following two statements are hold. Firstly, if **p** is an umbilic point of M, then the shape operator S at **p** is scalar multiplication by $k = k_1 = k_2$. Secondly, if **p** is a nonumblic point, then there are exactly two principal directions which are orthogonal to each other, and if $\mathbf{e}_1$ and $\mathbf{e}_2$ are principal vectors in these two directions then

$$S(\mathbf{e}_1) = k_1 \mathbf{e}_1, \tag{5}$$
$$S(\mathbf{e}_2) = k_2 \mathbf{e}_2. \tag{6}$$

Namely, the principal curvatures are the eigenvalues of S, and the principal vectors are the eigenvectors of S.

The Gaussian curvature of M at **p** is defined as the determinant of the shape operator S of M at **p**, or

$$K = \det(S). \tag{7}$$

The mean curvature of M at **p** is defined as half of the trace of the shape operator S of M at **p**, or

$$H = \frac{1}{2} \operatorname{trace}(S). \tag{8}$$

From the fact that the principal curvatures are the eigenvalues of S, and the principal vectors are the eigenvectors of S, thus the matrix of S at **p** with respect to $\mathbf{e}_1$ and $\mathbf{e}_2$ can be formed as

$$S = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}. \tag{9}$$

Therefore, the Gaussian curvature and the mean curvature are $K = k_1 k_2$, and $H = (k_1 + k_2)/2$.

In curvature computation, the first fundamental form and second fundamental form are used. Suppose the surface M is represented by parametric equations of $r = r(u, v)$, or $x = x(u, v)$, $y = y(u, v)$ and $z = z(u, v)$, where u and v are coordinate parameters determining position of each point on the surface patch, and all functions are assumed to be continuously differentiable [2]. The tangent vectors of the patch r are shown as follows:

$$r_u = \frac{\partial r}{\partial u} = \begin{bmatrix} \frac{\partial x}{\partial u} \\ \frac{\partial y}{\partial u} \\ \frac{\partial z}{\partial u} \end{bmatrix}, \tag{10}$$

$$r_v = \frac{\partial r}{\partial v} = \begin{bmatrix} \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial v} \\ \frac{\partial z}{\partial v} \end{bmatrix}. \tag{11}$$

The first fundamental form (I) is defined as the dot product of the surface tangent.

$$\begin{aligned} \mathsf{I} &= (r_u du + r_v dv) \cdot (r_u du + r_v dv) \\ &= (r_u)^2 (du)^2 + 2 r_u r_v du dv + (r_v)^2 (dv)^2 \\ &= E du^2 + 2F du dv + G dv^2 \end{aligned} \tag{12}$$

E, F, and G are coefficients of the first fundamental form, which are computed by the following.

$$E = (r_u)^2 = r_{uu} \tag{13}$$
$$F = (r_u)(r_v) = r_{uv} = r_{vu} \tag{14}$$
$$G = (r_v)^2 = r_{vv} \tag{15}$$

The second fundamental form (II) is defined as the negation of the dot product of the tangent and differential of the normal N.

$$\begin{aligned} \mathsf{II} &= -(r_u du + r_v dv) \cdot (\mathsf{N}_u du + \mathsf{N}_v dv) \\ &= -(r_u \mathsf{N}_u du^2 + r_u \mathsf{N}_v du dv \\ &\quad + r_v \mathsf{N}_u dv du + r_v \mathsf{N}_v dv^2 \\ &= -(r_u \mathsf{N}_u du^2 + 2 r_u \mathsf{N}_v du dv \\ &\quad + r_v \mathsf{N}_v dv^2 \\ &= L du^2 + 2M du dv + N dv^2 \end{aligned} \tag{16}$$

The following shows why $r_u \mathsf{N}_v = r_v \mathsf{N}_u$. Since the dot product of tangent vectors and the normal is equal to zero, $r_u \mathsf{N} = 0$, and $r_v \mathsf{N} = 0$, therefore,

$$\begin{aligned} \frac{\partial}{\partial v}(r_u \mathsf{N}) &= \frac{\partial}{\partial u}(r_v \mathsf{N}), \\ \mathsf{N} r_{uv} + r_u \mathsf{N}_v &= \mathsf{N} r_{vu} + r_v \mathsf{N}_v, \\ \therefore r_u \mathsf{N}_v &= r_v \mathsf{N}_u. \end{aligned} \tag{17}$$

e, f, and g are coefficients of the second fundamental form, and they are computed by the following.

$$e = -r_u \mathsf{N}_u = r_{uu} \mathsf{N} \tag{18}$$
$$f = -r_u \mathsf{N}_v = r_{uv} \mathsf{N} = -r_v \mathsf{N}_u \tag{19}$$
$$g = -r_v \mathsf{N}_v = r_{vv} \mathsf{N} \tag{20}$$

Also from $r_u \mathsf{N} = r_v \mathsf{N} = 0$, we can show that $-r_u \mathsf{N}_u = \mathsf{N} r_{uu}$, $-r_u \mathsf{N}_v = r_{uv} \mathsf{N}$, and $-r_v \mathsf{v} = r_{vv} \mathsf{N}$ by the following.

$$\begin{aligned} r_u \mathsf{N} &= 0 \\ \frac{\partial}{\partial u}(r_u \mathsf{N}) &= 0 \\ \mathsf{N} \frac{\partial^2 r}{\partial u^2} + r_u \frac{\partial \mathsf{N}}{\partial u} &= 0 \\ \mathsf{N} \frac{\partial^2 r}{\partial u^2} &= -r_u \frac{\partial \mathsf{N}}{\partial u} \\ \therefore \mathsf{N} r_{uu} &= -r_u \mathsf{N}_u \end{aligned} \tag{21}$$

$$\begin{aligned} r_v \mathsf{N} &= 0 \\ \frac{\partial}{\partial v}(r_v \mathsf{N}) &= 0 \\ \mathsf{N} \frac{\partial^2 r}{\partial v^2} + r_v \frac{\partial \mathsf{N}}{\partial v} &= 0 \\ \mathsf{N} \frac{\partial^2 r}{\partial v^2} &= -r_v \frac{\partial \mathsf{N}}{\partial v} \\ \therefore \mathsf{N} r_{vv} &= -r_v \mathsf{N}_v \end{aligned} \tag{22}$$

$$\begin{aligned} r_u \mathsf{N} &= 0 \\ \frac{\partial}{\partial v}(r_u \mathsf{N}) &= 0 \\ \mathsf{N} \frac{\partial^2 r}{\partial u \partial v} + r_u \frac{\partial \mathsf{N}}{\partial v} &= 0 \\ \mathsf{N} \frac{\partial^2 r}{\partial u \partial v} &= -r_u \frac{\partial \mathsf{N}}{\partial v} \\ \therefore \mathsf{N} r_{uv} &= -r_u \mathsf{N}_v \end{aligned} \tag{23}$$

Now come back in computation of e, f, and g. The unit normal vector N is computed from the cross product of tangent vectors,

$$\mathsf{N} = \frac{r_u \times r_v}{\|r_u \times r_v\|}, \tag{24}$$

where

$$\begin{aligned} \|r_u \times r_v\|^2 &= (r_u r_u)(r_v r_v) - (r_u r_v)^2, \\ &= EG - F^2. \end{aligned} \tag{25}$$

Therefore, e, f, and g are computed by the following.

$$
\begin{aligned}
e &= r_{uu} \cdot \frac{(r_u \times r_v)}{\|r_u \times r_v\|} \\
&= \frac{1}{\sqrt{EG - F^2}} \cdot \begin{vmatrix} x_{uu} & y_{uu} & z_{uu} \\ x_u & y_u & z_u \\ x_v & y_v & z_v \end{vmatrix} \qquad (26)
\end{aligned}
$$

$$
\begin{aligned}
f &= r_{uv} \cdot \frac{(r_u \times r_v)}{\|r_u \times r_v\|} \\
&= \frac{1}{\sqrt{EG - F^2}} \cdot \begin{vmatrix} x_{uv} & y_{uv} & z_{uv} \\ x_u & y_u & z_u \\ x_v & y_v & z_v \end{vmatrix} \qquad (27)
\end{aligned}
$$

$$
\begin{aligned}
g &= r_{vv} \cdot \frac{(r_u \times r_v)}{\|r_u \times r_v\|} \\
&= \frac{1}{\sqrt{EG - F^2}} \cdot \begin{vmatrix} x_{vv} & y_{vv} & z_{vv} \\ x_u & y_u & z_u \\ x_v & y_v & z_v \end{vmatrix} \qquad (28)
\end{aligned}
$$

Other than the above parametric form, $r = r(u,v) = (x(u,v), y(u,v), z(u,v))$, the surface M is also defined as $r(u,v) = (u, v, \zeta(u,v))$, where ζ is the differentiable real-valued function. All patches of this type are called ``Monge patches.'' We can say that the surface M is the image of the Monge patch [1]. The first and second derivatives of $r(u,v)$ are shown by the following.

$$
r_u = \frac{\partial r}{\partial u} = \begin{bmatrix} 1 \\ 0 \\ \zeta_u \end{bmatrix} \qquad (29)
$$

$$
r_v = \frac{\partial r}{\partial v} = \begin{bmatrix} 0 \\ 1 \\ \zeta_v \end{bmatrix} \qquad (30)
$$

$$
r_{uu} = \frac{\partial^2 r}{\partial u^2} = \begin{bmatrix} 0 \\ 0 \\ \zeta_{uu} \end{bmatrix} \qquad (31)
$$

$$
r_{vv} = \frac{\partial^2 r}{\partial v^2} = \begin{bmatrix} 0 \\ 0 \\ \zeta_{vv} \end{bmatrix} \qquad (32)
$$

$$
r_{uv} = \frac{\partial^2 r}{\partial u \partial v} = \begin{bmatrix} 0 \\ 0 \\ \zeta_{uv} \end{bmatrix} \qquad (33)
$$

$$
r_{vu} = \frac{\partial^2 r}{\partial v \partial u} = \begin{bmatrix} 0 \\ 0 \\ \zeta_{vu} \end{bmatrix} \qquad (34)
$$

Note that:

$$
r_u \times r_v = \begin{bmatrix} -\zeta_u \\ -\zeta_v \\ 1 \end{bmatrix}. \qquad (35)
$$

The coefficients of the first fundamental and second fundamental form are computed by the following.

$$
\begin{aligned}
E &= (r_u)^2 = 1 + \zeta_u^2 && (36) \\
F &= (r_u)(r_v) = 1 + \zeta_u \zeta_v && (37) \\
G &= (r_v)^2 = 1 + \zeta_v^2 && (38)
\end{aligned}
$$

$$
e = r_{uu} \cdot \frac{(r_u \times r_v)}{\|r_u \times r_v\|} = \frac{1}{\sqrt{EG - F^2}} \cdot \zeta_{uu} \qquad (39)
$$

$$
f = r_{uv} \cdot \frac{(r_u \times r_v)}{\|r_u \times r_v\|} = \frac{1}{\sqrt{EG - F^2}} \cdot \zeta_{uv} \qquad (40)
$$

$$
g = r_{vv} \cdot \frac{(r_u \times r_v)}{\|r_u \times r_v\|} = \frac{1}{\sqrt{EG - F^2}} \cdot \zeta_{vv} \qquad (41)
$$

From [3], the normal curvature at a point p on the surface M with the tangent direction **u** is expressed in terms of first and second fundamental forms as

$$
k(u) = \frac{Ldu^2 + 2Mdudv + Ndv^2}{Edu^2 + 2Fdudv + Gdv^2}. \qquad (42)
$$

Also for the Gaussian curvature and mean curvature, they are expressed as follows:

$$
\begin{aligned}
K &= \frac{eg - f^2}{EG - F^2}, && (43) \\
H &= \frac{eG - 2fF + gE}{2(EG - F^2)}. && (44)
\end{aligned}
$$

2 Curvature Estimation by Surface Fitting Based Methods

Suppose there is a set of data points in 3D space which represents the surface of a 3D object. The problem is that we would like to calculate curvatures of each data point. One approach to solve the problem is to approximate surface patches along the surface with some surface equations, and compute differential geometry information from the equations that lead to curvatures. In this section, techniques to approximate parametric equations, and implicit equations are explained and also methods to calculate curvatures.

2.1 Parametric Equations

Let M be a surface, which is represented by the graph of a function of two variables $z = f(x, y)$. The surface M can be defined by parametric equations of $r = r(u, v)$ or

$$x = x(u, v), \tag{45}$$

$$y = y(u, v), \tag{46}$$

$$z = z(u, v). \tag{47}$$

where parameters u and v determine the position of each point on the patch, and all functions are assumed to be continuously differentiable [2]. Goldgof presented calculation of curvatures for a point when a set of 3D data points is given in [4] as shown in the following steps:

1. Fit a plane to the set of 3D points such that the sum of squares of distances from points to the plane is minimized.
2. Rotate all data points such that the plan normal is aligned with the Z-axis, and then translate all data points by $-T_f$, where T_f is the vector from the mean of all data points to the origin. Therefore, the plane is coincided with the XY-plane.
3. Perform the least-squares error fitting (on transformed data) in the Z-direction to fit the second order surface of the form,

$$\begin{aligned} z(x,y) &= \gamma_1 + \gamma_2 x + \gamma_3 y + \gamma_4 xy \\ &\quad + \gamma_5 x^2 + \gamma_6 y^2. \end{aligned} \tag{48}$$

4. Calculate all necessary derivatives at the point $p(x_i, y_i, z_i)$ for calculating of the Gaussian and mean curvatures according to the aforementioned equations:

$$z_x = \gamma_2 + \gamma_4 y_i + 2\gamma_5 x_i, \tag{49}$$

$$z_y = \gamma_3 + \gamma_4 x_i + 2\gamma_6 y_i, \tag{50}$$

$$z_{xx} = 2\gamma_5, \tag{51}$$

$$z_{yy} = 2\gamma_6, \tag{52}$$

$$z_{xy} = \gamma_4. \tag{53}$$

Suppose there are n data points. In the first step of fitting a plane of the equation $Ax+By+Cz+D = 0$ to the 3D points, it is performed by first creating a scatter matrix S, where $S = \mathrm{VV^t}$. Let $v_i = (x_i, y_i, z_i)^t$ be the coordinate vector of the point p_i with the mean subtracted, and V is a matrix of size $3 \times n$, and V is defined as

$$\mathrm{V} = \begin{bmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \\ z_1 & \cdots & z_n \end{bmatrix}. \tag{54}$$

The plane coefficients are equal to the coefficients of the eigenvector with the smallest eigenvalue of the scatter matrix, and the normal of the plane is the vector $[A, B, C]$. The transformation of the data point can also be considered as transformation of the normal vector to the positive Z-axis.

To fit the second order surface to the transformed data, the least squares fit can also be applied [5]. With n transformed data points, $(x_i, y_i, z_i) = (x_i, y_i, z(x_i, y_i))$, where $1 \leq i \leq n$, we get following n equations.

$$\begin{aligned} \gamma_1 + \gamma_2 x_1 + \gamma_3 y_1 + \gamma_4 x_1 y_1 + \gamma_5 x_1^2 &= z_1 \\ \gamma_1 + \gamma_2 x_2 + \gamma_3 y_2 + \gamma_4 x_1 y_2 + \gamma_5 x_2^2 &= z_2 \\ &\vdots \\ \gamma_1 + \gamma_2 x_n + \gamma_3 y_n + \gamma_4 x_1 y_n + \gamma_5 x_n^2 &= z_n \end{aligned} \tag{55}$$

The above equations can also be written in matrix form as $V\Gamma = Z$, where

$$
\begin{aligned}
V &= \begin{bmatrix} 1 & x_1 & y_1 & x_1y_1 & x_1^2 & y_1^2 \\ 1 & x_2 & y_2 & x_2y_2 & x_2^2 & y_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & y_n & x_ny_n & x_n^2 & y_n^2 \end{bmatrix}, \quad (56) \\
\Gamma &= \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \gamma_5 & \gamma_6 \end{bmatrix}^T, \quad (57) \\
Z &= \begin{bmatrix} z_1 & z_2 & \dots & z_n \end{bmatrix}^T. \quad (58)
\end{aligned}
$$

Therefore, all coefficients of the second order surface equation are computed as follows:

$$
\begin{aligned}
V\Gamma &= Z, \\
V^TV\Gamma &= V^TZ, \\
\therefore \Gamma &= (A^TA)^{-1}A^TZ.
\end{aligned} \quad (59)
$$

2.2 Implicit Equation

Implicit surfaces are defined by a polynomial of three variables $F(x, y, z) = 0$. We call surfaces which have polynomial (implicit) forms ``algebraic surfaces.'' The highest degree of all terms is the degree of the algebraic surface, for example, spheres and all quadric surfaces are algebraic surfaces of degree two. In our study, the implicit function is in the quadric form of

$$
\begin{aligned}
F(x, y, z) &= ax^2 + by^2 + cz^2 \\
&+ 2exy + 2fyz + 2gzx \\
&+ 2lx + 2my + 2nz + d \\
&= 0,
\end{aligned} \quad (60)
$$

where $a, b, c, e, f, g, l, m, n$, and d are coefficients.

Curvatures of points on a surface can be approximated by fitting of quadric surface patches. For example, in [6], Douros and Buxton applied fitting of quadric surface patches to estimate curvature maps of 3D human body surface data, which are acquired by Body scanning hardware. The curvature estimation for every point, p, is processed as follows:

1. Select a neighborhood of N points around p
2. Fit a quadric patch $F(x, y, z) = 0$ to those selected points
3. Calculate a projection p_0 of p onto the patch, such that $F(p_0) = 0$
4. Compute the Gaussian and mean curvatures from the coefficients of the first and second fundamental forms at p_0

In the first step of neighborhood selection, neighboring points are selected from points which are within a close area that includes the point p. If the number of included neighboring points is less than the number of necessary coefficients of the implicit equation, then the close area will be enlarged, and neighboring points will be re-selected. The close area will be enlarged until the number of neighboring points is larger than the number of necessary coefficients. For example, more than 9 points are required for fitting of the quadric equation, because the equation has 9 necessary coefficients.

After selection of neighboring points, then fitting of a quadric patch is performed. The quadric form that is used in fitting is shown by the following.

$$
\begin{aligned}
F(x, y, z) &= ax^2 + by^2 + cz^2 \\
&+ 2exy + 2fyz + 2gzx \\
&+ 2lx + 2my + 2nz + d \\
&= 0
\end{aligned} \quad (61)
$$

Suppose (x_i, y_i, z_i) are selected neighboring points.

$$
\begin{aligned}
F(x_i, y_i, z_i) &= ax_i^2 + by_i^2 + cz_i^2 \\
&+ 2ex_iy_i + 2fy_iz_i \\
&+ 2gz_ix_i + 2lx_i \\
&+ 2my_i + 2nz + i + d \\
&= \varepsilon_i,
\end{aligned} \quad (62)
$$

where ε_i are called ``algebraic distance.'' In case that points (x_i, y_i, z_i) lie on the surface, $\varepsilon_i = 0$. However, points may always not lie on the surface. Therefore, in order to find the solution of coefficients, we compute the following.

$$
\min\left\{\sum_i \varepsilon_i^2\right\} = \min\left\{\sum_i F^2(x_i, y_i, z_i)\right\} \quad (63)
$$

However, without any constraints, the minimum of the above is given by $a = b = c = e = f = g = l = m = n = d = 0$, thus a constraint is defined as $a^2+b^2+c^2+d^2+2e^2+2f^2+2g^2+2l^2+2m^2+2n^2 = 1$.

By homogeneous coordinates, the quadric equation can be represented as follows:

$$X^T U X = 0, \tag{64}$$

$$U = \begin{bmatrix} a & e & g & l \\ e & b & f & m \\ g & f & c & n \\ l & m & n & d \end{bmatrix}, \tag{65}$$

where X^T is transpose of X, $X = [x\ y\ z\ 1]^T$, and U is the real symmetric matrix containing the coefficients. The minimization problem is thus

$$\begin{aligned} \min\{\textstyle\sum_i \varepsilon_i^2\} &= \min \textstyle\sum_i F^2(x_i, y_i, z_i) \\ &= \min\{\textstyle\sum_i [X^T U X]^2\} \end{aligned} \tag{66}$$

subject to trace$(U^T U) = 1$. The problem can be reduced to

$$\min\left\{\sum_i \left(\sum_{\alpha\leq\beta} x_\alpha u_{\alpha\beta} x_\beta\right)^2\right\} \tag{67}$$

subject to $\sum_{\alpha\leq\beta} u_{\alpha\beta}^2 = 1$; where $u_{\alpha\beta}$ are elements at row α and column β of the matrix U such that $\alpha \leq \beta$ and $\alpha, \beta = 1, 2, 3, 4$. Therefore, we have

$$\begin{aligned} u_{\alpha\beta} \in \{&u_{11}, u_{12}, u_{13}, u_{14}, \\ &u_{22}, u_{23}, u_{24}, \\ &u_{33}, u_{34}, u_{44}\}, \end{aligned} \tag{68}$$

which is equal to $\{a, e, g, l, b, f, m, c, n, d\}$, and

$$x_\alpha, x_\beta \in \{x_1, x_2, x_3, x_4\}, \tag{69}$$

which is equal to $\{x, y, z, 1\}$. Therefore,

$$\begin{aligned} &\textstyle\sum_{\alpha\leq\beta} x_\alpha u_{\alpha\beta} x_\beta = x_1 u_{11} x_1 \\ &+ x_1 u_{12} x_2 + x_1 u_{13} x_3 + x_1 u_{14} x_4 \\ &+ x_2 u_{22} x_2 + x_2 u_{23} x_3 + x_2 u_{24} x_4 \\ &+ x_3 u_{33} x_3 + x_3 u_{34} x_4 + x_4 u_{44} x_4 \\ &= ax^2 + by^2 + cz^2 + exy + fyz + gxz \\ &+ lx + my + nz + d. \end{aligned} \tag{70}$$

In order to solve the minimization problem, the Lagrange Multiplier method is applied. In general, the method is used to find the extremum of a function $f(x_1, x_2, ..., x_n)$ subject to the constraint function $g(x_1, x_2, ..., x_n) = C$, where f and g are functions with continuous first partial derivatives. Then for an extremum to exist,

$$\begin{aligned} df &= \frac{\partial f}{\partial x_1} dx_1 + \frac{\partial f}{\partial x_2} dx_2 + \cdots \\ &+ \frac{\partial f}{\partial x_n} dx_n = 0, \end{aligned} \tag{71}$$

$$\begin{aligned} dg &= \frac{\partial g}{\partial x_1} dx_1 + \frac{\partial g}{\partial x_2} dx_2 + \cdots \\ &+ \frac{\partial g}{\partial x_n} dx_n = 0. \end{aligned} \tag{72}$$

Multiply dg by λ and add it to df, so we get the following.

$$\begin{aligned} &\left(\tfrac{\partial f}{\partial x_1} + \lambda \tfrac{\partial g}{\partial x_1}\right) dx_1 + \left(\tfrac{\partial f}{\partial x_2} + \lambda \tfrac{\partial g}{\partial x_2}\right) dx_2 \\ &+ \cdots + \left(\tfrac{\partial f}{\partial x_n} + \lambda \tfrac{\partial g}{\partial x_n}\right) dx_n = 0 \end{aligned} \tag{73}$$

Note that the differentials are all independent, so we can set any combination equal to zero, and the remainder still gives zero. Therefore,

$$\left(\frac{\partial f}{\partial x_k} + \lambda \frac{\partial g}{\partial x_k}\right) dx_k = 0, k = 1, 2, ..., n. \tag{74}$$

The constant λ is called the *Lagrange multiplier*.

In our problem, we have

$$f(u_{\alpha\beta}) = \sum_i \left(\sum_{\alpha\leq\beta} x_\alpha u_{\alpha\beta} x_\beta\right)^2, \tag{75}$$

$$g(u_{\alpha\beta}) = 1 - \sum_{\alpha\leq\beta} u_{\alpha\beta}^2. \tag{76}$$

Compute partial derivatives of f and g by $u_{\gamma\delta}$, where $\gamma \leq \delta$.

$$\frac{\partial f}{\partial u_{\gamma\delta}} = \sum_i \left(2 x_\gamma x_\delta \left(\sum_{\alpha\leq\beta} x_\alpha u_{\alpha\beta} x_\beta\right)\right) \tag{77}$$

$$\frac{\partial g}{\partial u_{\gamma\delta}} = -2 u_{\gamma\delta} \tag{78}$$

Therefore we get the following equations.

$$\begin{array}{c} \sum_i \left(2x_\gamma x_\delta \left(\sum_{\alpha\le\beta} x_\alpha u_{\alpha\beta} x_\beta\right)\right) \\ -\lambda.2u_{\gamma\delta} = 0 \\ \sum_{\alpha\le\beta}\left(\sum_i x_\gamma x_\delta x_\alpha x_\beta\right) u_{\alpha\beta} \\ = \lambda u_{\gamma\delta} \end{array} \tag{79}$$

The above equation is in a matrix eigenvalue-eigenvector form $Au = \lambda u$, where A is the 10×10 matrix, and u is the 10×1 column vector constructed from the independent elements of the symmetric matrix U.

$$u = \begin{bmatrix} u_{11} \\ u_{12} \\ u_{13} \\ u_{14} \\ u_{22} \\ u_{23} \\ u_{24} \\ u_{33} \\ u_{34} \\ u_{44} \end{bmatrix} = \begin{bmatrix} a \\ e \\ g \\ l \\ b \\ f \\ m \\ c \\ n \\ d \end{bmatrix} \tag{80}$$

$$A = \begin{bmatrix} A_1 & A_2 & \cdots & A_{10} \end{bmatrix} \tag{81}$$

$$A_1 = \begin{bmatrix} \sum_i x_1x_1x_1x_1 \\ \sum_i x_1x_2x_1x_1 \\ \vdots \\ \sum_i x_4x_4x_1x_1 \end{bmatrix} \tag{82}$$

$$A_2 = \begin{bmatrix} \sum_i x_1x_2x_1x_1 \\ \sum_i x_1x_2x_1x_2 \\ \vdots \\ \sum_i x_4x_4x_1x_2 \end{bmatrix} \tag{83}$$

$$A_{10} = \begin{bmatrix} \sum_i x_4x_4x_1x_1 \\ \sum_i x_4x_4x_1x_2 \\ \vdots \\ \sum_i x_4x_4x_4x_4 \end{bmatrix} \tag{84}$$

Alternatively, A can be expressed as

$$A = D^T D, \tag{85}$$

where D is the $N \times 10$ design matrix, and N is the number of neighboring points. Each row of D is defined as

$$D_i = \begin{bmatrix} x_i^2 \\ y_i^2 \\ z_i^2 \\ x_iy_i \\ y_iz_i \\ x_iz_i \\ x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}^T, \tag{86}$$

where $i = 1, \ldots, N$.

Then 10 real eigenvectors and 10 corresponding eigenvalues of A are computed. Eigenvectors u is then normalized to unity, so that the constraint is satisfied. The following will show how $\sum_i \varepsilon_i^2$ is related to λ.

$$\begin{array}{l} \sum_i \varepsilon_i^2 = \sum_i F^2(x_i, y_i, z_i) \\ = \sum_i \left(\sum_{\alpha\le\beta} x_\alpha u_{\alpha\beta} x_\beta\right)^2 \\ = \sum_i \left(\sum_{\alpha\le\beta} x_\alpha u_{\alpha\beta} x_\beta\right)\left(\sum_{\gamma\le\delta} x_\gamma u_{\gamma\delta} x_\delta\right) \\ = \sum_{\alpha\le\beta} u_{\alpha\beta} \sum_{\gamma\le\delta}\left(\sum_i x_\alpha x_\beta x_\gamma x_\delta\right) u_{\gamma\delta} \\ = \sum_{\alpha\le\beta} u_{\alpha\beta}\left(\sum_{\gamma\le\delta} A_{\alpha\beta\gamma\delta} u_{\gamma\delta}\right) \\ = \sum_{\alpha\le\beta} \lambda u_{\alpha\beta} u_{\alpha\beta} \\ = \lambda\left(\sum_{\alpha\le\beta} u_{\alpha\beta} u_{\alpha\beta}\right) \\ \therefore \sum_i \varepsilon_i^2 = \lambda \end{array} \tag{87}$$

We can see that the eigenvalues represent the error of fit of the eigenvector coefficients. Since we want the solution which gives the smallest sum of squared errors, the solution coefficients of the equation is selected from the coefficients of the eigenvector with the smallest corresponding eigenvalue. The confidence of fit is then equal to λ_2/λ_1, where λ_1 and λ_2 are the first and second smallest eigenvalues. However, if the matrix A has the matrix rank lower than 10, the solutions can be unreliable. Therefore instead of using the original data points, we should preprocess these data to avoid low rank. For example, in [4], the data points are transformed by Tf, where Tf is the vector from the original point to the mean point of the data points. Now these transformed points are applied to fit the quadric equation. Moreover, the Householder and QL methods are suggested to use to calculate eigenvalues and eigenvectors, because they are efficient and reliable.

Since the implicit equation is homogeneous, there can be two possible solutions which signs of coefficients are negated to each other. For example, solution coefficients can be either $a, b, c, -d$, or $-a, -b, -c, d$. This ambiguity can later cause incorrect signs of mean curvatures. Therefore, after we get the coefficients from the quadric surface fitting, all coefficients must be divided by the first coefficient, a.

After getting correct coefficients, we approximate the point on the surface that is close to our interesting point. There are three possible ways to do by the following:

- The first way is to assume that $F(p)$ is very close to zero, or $p_0 = p$. The method produces some errors depending on how well the surface is fitted.

- The second way is to calculate the normal line on F that passes through p, and compute p_0 as the intersection of the line and F. However, the result formulation of the problem has no closed-form solution.

- The third way is an approximation method [6], which is processed as follows:

  1. select three lines n_1, n_2, and n_3 that pass through p. Note that these three lines should be orthogonal to each other, and parallel to the axes of the orthogonal coordinate system.

  2. compute the intersection p_i of each line with F. There can be either zero, one, or two solutions for each intersection problem. In case of zero solutions, that n_i is rejected. In case of two solutions, the intersection point that is closer to p by distance is selected.

  3. create a new line n_4 from the remaining lines such that n_4 is the mean of them.

  4. calculate the intersection of n_4 with F.

  5. now there are at most four possible intersection points, the point that gives the smallest distance to p is selected as p_0.

Note that line equations in 3D space are represented in parametric form is parallel to the vector (v_1, v_2, v_3) so the line equations are $(x, y, z) = (x_0, y_0, z_0) + t(v_1, v_2, v_3)$, or

$$x = x_0 + v_1 t, \tag{88}$$

$$y = y_0 + v_2 t, \tag{89}$$

$$z = z_0 + v_3 t. \tag{90}$$

Therefore, intersection points of a 3D line and a quadric surface are computed by replacing line equations into the quadric equation, and solving for t. Intersection points (x, y, z) are then calculated from t. Remind that the quadric surface equation is expressed by the following.

$$\begin{aligned} F(x,y,z) &= ax^2 + by^2 + cz^2 \\ &+ 2exy + 2fyz + 2gzx \\ &+ 2lx + 2my + 2nz \\ &+ d \\ &= 0 \end{aligned} \tag{91}$$

By replacing line equations into the quadric equation, we get the following.

$$\begin{aligned} & a(x_0 + v_1 t)^2 + 2l(x_0 + v_1 t) \\ + \; & b(y_0 + v_2 t)^2 + 2m(y_0 + v_2 t) \\ + \; & c(z_0 + v_3 t)^2 + 2n(z_0 + v_3 t) \\ + \; & 2e(x_0 + v_1 t)(y_0 + v_2 t) \\ + \; & 2f(y_0 + v_2 t)(z_0 + v_3 t) \\ + \; & 2g(z_0 + v_3 t)(x_0 + v_1 t) + d = 0 \end{aligned} \tag{92}$$

$$At^2 + Bt + C = 0 \tag{93}$$

A, B, and C are computed as follows:

$$\begin{aligned} A &= av_1^2 + bv_2^2 + cv_3^2 \\ &+ 2ev_1v_2 + 2fv_2v_3 + 2gv_3v_1, \quad (94) \\ B &= 2ax_0v_1 + 2by_0v_2 + 2cz_0v_3 \\ &+ 2ey_0v_1 + 2ex_0v_2 + 2fz_0v_2 \\ &+ 2fy_0v_3 + 2gx_0v_3 + 2gz_0v_1 \\ &+ 2lv_1 + 2mv_2 + 2nv_3, \quad (95) \\ C &= ax_0^2 + by_0^2 + cz_0^2 \\ &+ 2ex_0y_0 + 2fy_0z_0 + 2gz_0x_0 \\ &+ 2lx_0 + 2my_0 + 2nz_0 + d, \quad (96) \end{aligned}$$

then

$$t = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}. \quad (97)$$

The next step is to compute curvatures. We thus need to compute coefficients of the first and second fundamental forms [7]. First we transform our implicit function $F(x, y, z) = 0$ into the Monge patch form by letting $z = \zeta(x, y)$, or $F(x, y, \zeta(x, y)) = 0$. By implicit differentiation, the total differential of F is the following.

$$dF = \frac{\partial F}{\partial x}dx + \frac{\partial F}{\partial y}dy + \frac{\partial F}{\partial z}dz \quad (98)$$

Therefore, partial derivatives of ζ are the following.

$$\begin{aligned} \frac{\partial F}{\partial x} &= \frac{\partial F}{\partial x}\cdot\frac{\partial x}{\partial x} + \frac{\partial F}{\partial y}\cdot\frac{\partial y}{\partial x} + \frac{\partial F}{\partial z}\cdot\frac{\partial \zeta}{\partial x} \\ 0 &= \frac{\partial F}{\partial x} + \frac{\partial F}{\partial z}\cdot\frac{\partial \zeta}{\partial x} \\ \therefore \frac{\partial \zeta}{\partial x} &= -\frac{F_x}{F_z} \quad (99) \\ \frac{\partial F}{\partial y} &= \frac{\partial F}{\partial x}\cdot\frac{\partial x}{\partial y} + \frac{\partial F}{\partial y}\cdot\frac{\partial y}{\partial y} + \frac{\partial F}{\partial z}\cdot\frac{\partial \zeta}{\partial y} \\ 0 &= \frac{\partial F}{\partial y} + \frac{\partial F}{\partial z}\cdot\frac{\partial \zeta}{\partial y} \\ \therefore \frac{\partial \zeta}{\partial y} &= -\frac{F_y}{F_z} \quad (100) \end{aligned}$$

The coefficients E, F, and G of the first fundamental forms are expressed by the following.

$$\begin{aligned} E &= 1 + \zeta_x^2 = 1 + \frac{F_x^2}{F_z^2} \quad (101) \\ F &= \zeta_x\zeta_y = \frac{F_xF_y}{F_z^2} \quad (102) \\ G &= 1 + \zeta_y^2 = 1 + \frac{F_y^2}{F_z^2} \quad (103) \end{aligned}$$

The magnitude of the gradient is,

$$|\nabla| = \sqrt{F_x^2 + F_y^2 + F_z^2}, \quad (104)$$

and the relationship between these cofficients are

$$EG - F^2 = (F_x^2 + F_y^2 + F_z^2)/F_z^2. \quad (105)$$

In order to compute, the coefficients L, M, and N of the second fundamental form, second derivatives of ζ must be calculated.

$$\begin{aligned} \zeta_{xx} &= -\frac{(F_zF_{xx} - F_xF_{zx})}{F_z^2} \\ &= \frac{1}{F_z^3}\begin{vmatrix} F_{xx} & F_{xz} & F_x \\ F_{zx} & F_{zz} & F_z \\ F_x & F_z & 0 \end{vmatrix} \quad (106) \\ \zeta_{xy} &= -\frac{(F_zF_{xy} - F_xF_{zy})}{F_z^2} \\ &= \frac{1}{F_z^3}\begin{vmatrix} F_{xy} & F_{yz} & F_y \\ F_{xz} & F_{zz} & F_z \\ F_x & F_z & 0 \end{vmatrix} \quad (107) \\ \zeta_{yy} &= -\frac{(F_zF_{yy} - F_zF_{zy})}{F_z^2} \\ &= \frac{1}{F_z^3}\begin{vmatrix} F_{yy} & F_{yz} & F_y \\ F_{zy} & F_{zz} & F_z \\ F_y & F_z & 0 \end{vmatrix} \quad (108) \end{aligned}$$

Then, e, f, and g are computed by the following.

$$\begin{aligned} e &= \frac{1}{F_z^2|\nabla|}\begin{vmatrix} F_{xx} & F_{xz} & F_x \\ F_{zx} & F_{zz} & F_z \\ F_x & F_z & 0 \end{vmatrix} \quad (109) \\ f &= \frac{1}{F_z^2|\nabla|}\begin{vmatrix} F_{xy} & F_{yz} & F_y \\ F_{zx} & F_{zz} & F_z \\ F_x & F_z & 0 \end{vmatrix} \quad (110) \\ g &= \frac{1}{F_z^2|\nabla|}\begin{vmatrix} F_{yy} & F_{yz} & F_y \\ F_{zy} & F_{zz} & F_z \\ F_y & F_z & 0 \end{vmatrix} \quad (111) \end{aligned}$$

Then we compute curvatures by forming two 2×2 matrices A and B.

$$A = \begin{bmatrix} e & f \\ f & g \end{bmatrix} \quad (112)$$

$$B = \begin{bmatrix} E & F \\ F & G \end{bmatrix} \quad (113)$$

The Gaussian curvature (K) is defined as, $K = \det(A)/\det(B)$, and the mean curvature (H) is defined as $H = \frac{1}{2}\operatorname{trace}(A^{-1}B)$. The principal curvatures are the eigenvalues k_1 and k_2 of the matrix $B^{-1}A$. Therefore, the Gaussian curvature and the mean curvature are also computed as $K = k_1.k_2$ and $H = \frac{1}{2}(k_1 + k_2)$.

3 Curvature Estimation on 3D Meshes

There are many proposed methods to estimate curvatures from surface meshes. A classic way to do is to fit local surface to meshes and then apply partial derivatives to get curvatures. The other method is to use tensor based techniques. Moreover, curvatures can be estimated at each vertex point from the mesh shape in a discrete domain.

3.1 Surface fitting based method

The first way to estimate curvatures on 3D meshes is by local surface fitting of meshes. Curvatures are then calculated from the fitting function. For example, Yuen, Khalili and Mokhtarian presented a technique to estimate curvatures on smoothed 3-D meshes in [8, 9]. According to the literatures, first from triangulated models of 3-D objects, a local parameterization technique such as local surface fitting is applied to get a local parametric representation of a surface in coordinates u and v. This representation is expressed as

$$r(u,v) = (x(u,v), y(u,v), z(u,v)), \quad (114)$$

when $r(u,v)$ corresponds to semigeodesic coordinates, Gaussian curvature K is given by

$$K = \frac{b_{uu}b_{vv} - b_{uv}^2}{x_v^2 + y_v^2 + z_v^2}, \quad (115)$$

and

$$b_{uu} = \frac{Ax_{uu} + By_{uu} + Cz_{uu}}{\sqrt{A^2+B^2+C^2}}, \quad (116)$$

$$b_{vv} = \frac{Ax_{vv} + By_{vv} + Cz_{vv}}{\sqrt{A^2+B^2+C^2}}, \quad (117)$$

$$b_{uv} = \frac{Ax_{uv} + By_{uv} + Cz_{uv}}{\sqrt{A^2+B^2+C^2}}, \quad (118)$$

where $A = y_u z_u - z_u y_v$, $B = x_v z_u - z_v x_u$, and $C = x_u y_v - y_u x_v$. Mean curvature H is

$$H = \frac{b_{vv} + (x_v^2 + y_v^2 + z_v^2)b_{uu}}{2(x_v^2 + y_v^2 + z_v^2)}. \quad (119)$$

For each point of the surface, $P(x(u,v), y(u,v), z(u,v))$, the corresponding neighboring data is convolved with the partial derivatives of the Gaussian function $(G(u,v,\sigma)$.

$$x_u = x * \frac{\partial G}{\partial u} \quad (120)$$

$$y_u = y * \frac{\partial G}{\partial u} \quad (121)$$

$$z_u = z * \frac{\partial G}{\partial u} \quad (122)$$

$$x_v = x * \frac{\partial G}{\partial v} \quad (123)$$

$$y_v = y * \frac{\partial G}{\partial v} \quad (124)$$

$$z_v = z * \frac{\partial G}{\partial v} \quad (125)$$

$$x_{uu} = x * \frac{\partial^2 G}{\partial u^2} \quad (126)$$

$$y_{uu} = y * \frac{\partial^2 G}{\partial u^2} \quad (127)$$

$$z_{uu} = z * \frac{\partial^2 G}{\partial u^2} \quad (128)$$

$$x_{vv} = x * \frac{\partial^2 G}{\partial v^2} \quad (129)$$

$$y_{vv} = y * \frac{\partial^2 G}{\partial v^2} \quad (130)$$

$$z_{vv} = z * \frac{\partial^2 G}{\partial v^2} \quad (131)$$

$$x_{uv} = x * \frac{\partial^2 G}{\partial u \partial v} \quad (132)$$

$$y_{uv} = y * \frac{\partial^2 G}{\partial u \partial v} \quad (133)$$

$$z_{uv} = z * \frac{\partial^2 G}{\partial u \partial v} \quad (134)$$

3.2 Surface Mesh Curvatures

3.2.1 Gaussian Curvature

In [10, 11], calculation of Gaussian curvature is explained. The Gaussian curvature K at a vertex point is computed from its adjacent triangles,

$$K = \frac{\rho \Delta\theta}{A}, \quad (135)$$

$$\Delta\theta = 2\pi - \sum_i \theta_i, \quad (136)$$

$$A = \sum_i A_i, \quad (137)$$

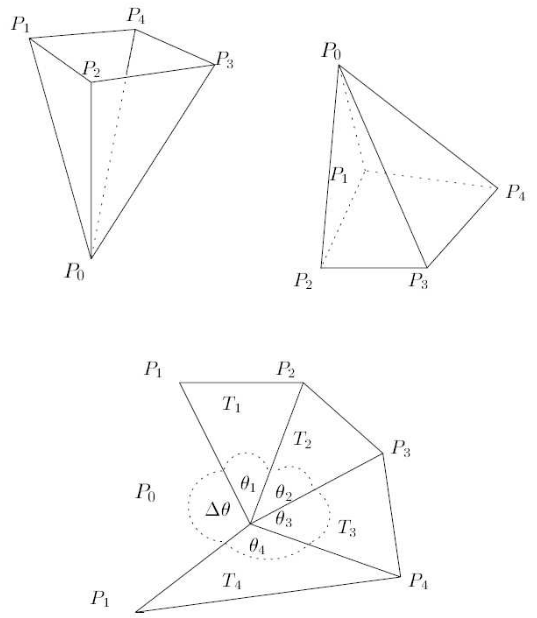
where A is the total area of the adjacent triangles T_i, for $i = 1, 2, 3, ...$ Figure 4 shows example of curvature approximation at different surface regions. Figure 4 (a) illustrates the case of vertex P_0 is in convex and concave regions. Figure 4 (b) illustrates the case of vertex P_0 is in a saddle region. In the first case (Figure 4 (a)), $\sum_i \theta_i < 2\pi$, hence the Gaussian curvature is positive $(K > 0)$. In the second case (Figure 4 (b)), $\sum_i \theta_i > 2\pi$, so the Gaussian curvature is negative $(K < 0)$. In the case of flat surface, $\sum_i \theta_i = 2\pi$, therefore, the Gaussian curvature is equal to zero.

3.2.2 Mean Curvature

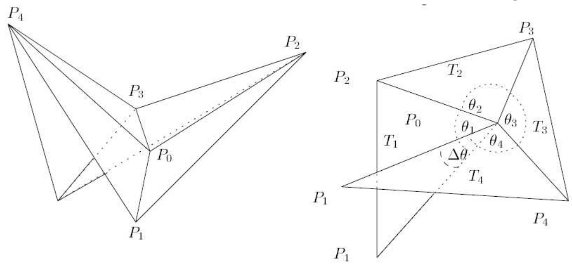
The mean curvature is defined by the divergence of the surface around the normal vector, $H = \nabla \vec{n}$. From [12, 13, 11], the mean curvature normal for a surface mesh is computed by

$$-H\vec{n} = \frac{1}{4A} \sum_{j \in N(i)} (\cot \alpha_j + \cot \beta_j)(x_j - x_i), \quad (138)$$

where $N(i)$ is the vertex X_i's adjacent polygon set, $(x_j - x_i)$ is the edge e_{ij}, α_j and β_j are two angles



(a). Vertex P_0 is in convex or concave regions



(b). Vertex P_0 is in a saddle region

Figure 4. Approximation of Gaussian curvature at vertex P_0

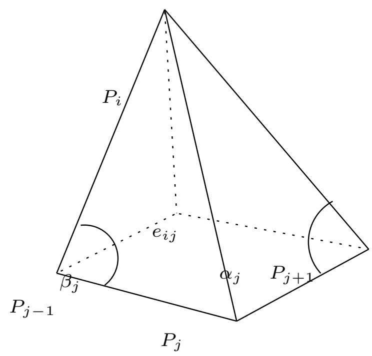


Figure 5. Approximation of mean curvature at vertex P_i

in j^{th} and $(j-1)^{th}$ element in $N(i)$ opposite to the edge e_{ij}, respectively, and A is the sum of the areas of triangles in $N(i)$. Figure 5 shows approximating of mean curvature at vertex P_i.

3.2.3 Total Curvature

From [14], in case that the Gaussian curvature and the mean curvature are already known, the total curvature of a vertex can be calculated by

$$D = \sqrt{4H^2 - 2K^2}, \tag{139}$$

where D is the total curvature, K is the Gaussian curvature, and H is the mean curvature. The total curvature can also be estimated from the norm of the covariance of the surface normals that are adjacent to that vertex [14]. The covariance matrix is first defined from the variance and covariance in all three cardinal directions. The variance and covariance are computed as follows:

$$\sigma_{xx}^2 = \frac{1}{N}\sum_{t=0}^{N}(x_t - \bar{x})^2, \tag{140}$$

$$\sigma_{yy}^2 = \frac{1}{N}\sum_{t=0}^{N}(y_t - \bar{y})^2, \tag{141}$$

$$\sigma_{zz}^2 = \frac{1}{N}\sum_{t=0}^{N}(z_t - \bar{z})^2, \tag{142}$$

$$\begin{aligned}
\sigma_{xy}^2 &= \sigma_{yx}^2 \\
&= \frac{1}{N}\sum_{t=0}^{N}(x_t - \bar{x})(y_t - \bar{y}), \quad (143)\\
\sigma_{yz}^2 &= \sigma_{zy}^2 \\
&= \frac{1}{N}\sum_{t=0}^{N}(y_t - \bar{y})(z_t - \bar{z}), \quad (144)\\
\sigma_{zx}^2 &= \sigma_{xz}^2 \\
&= \frac{1}{N}\sum_{t=0}^{N}(z_t - \bar{z})(x_t - \bar{x}), \quad (145)
\end{aligned}$$

where N is the number of triangles associated with this vertex, and $[x_t y_t z_t]$ are the normal of triangle t.

	K > 0	K $= 0$	K < 0
H < 0	Peak T=1	Ridge T=2	Saddle Ridge T=3
H $= 0$	none T=4	Flat T=5	Minimal Surface T=6
H > 0	Pit T=7	Valley T=8	Saddle Valley T=9

Table 1. Surface type labels from surface curvature signs

The total curvature D is equal to the norm of the covariance matrix C.

$$D = ||C|| \tag{146}$$

$$C = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} \tag{147}$$

4 Surface Classification

There are many ways to assign the surface type of a point. The first method is based on Gaussian curvature (K) and mean curvature (H) by Besl and Jain in [15].

The surface type, T, is defined as

$$\begin{aligned} T &= 1 + 3(1 + sgn(H,\epsilon)) \\ &\quad + (1 - sgn(K,\epsilon)), \end{aligned} \tag{148}$$

and a tolerance signum function is defined by

$$sgn(x,\epsilon) = \begin{cases} +1 & : \quad x > \epsilon \\ 0 & : \quad |x| \le \epsilon \\ -1 & : \quad x < \epsilon \end{cases} \tag{149}$$

The Table 1 shows the relationship of K, H, and T.

The second method is based on principal curvatures, k_1 and k_2, by Koenderink and Doorn in [16]. In this method, a shape index at each point is computed by

$$s = \frac{2}{\pi}\arctan\frac{k_2 + k_1}{k_2 - k_1}, \tag{150}$$

Shape index range	Surface types
$s \in [-1, -7/8]$	Spherical cup
$s \in [-7/8, -5/8)$	Trough
$s \in [-5/8, -3/8)$	Rut
$s \in [-3/8, -1/8)$	Saddle ruts
$s \in [-1/8, +1/8)$	Saddle
$s \in [+1/8, +3/8)$	Saddle ridges
$s \in [+3/8, +5/8)$	Ridge
$s \in [+5/8, +7/8)$	Dome
$s \in [+7/8, +1)$	Spherical cap

Table 2. Surface type assigned from shape index

where $k_1 \geq k_2$. A surface type is assigned related to the shape index as in the Table 2.

In addition to the shape index, the cuvedness was defined as the distance from the origin in the (k_1, k_2)-plane. It is used to specify the amount of `curvedness' or `intensity' of the surface curvature. The curvedness is computed by

$$c = \sqrt{\frac{k_1^2 + k_2^2}{2}}, \qquad (151)$$

where c is always positive.

5 Conclusions

We presented basic theory of differential geometry on surface; and also surveyed practical computing techniques used to calculate important differential geometric measurements which are related to surface properties. These measurements can be applied in various surface analysis applications, for example, package design, material surface analysis, surface contaction bewteen non-rigid objects, biological surface study, etc.

6 Acknowledgement

The author would like to acknowledge NIH Grant Number RR-15588 from the COBRE Program of the National Center for Research Resources.

References

[1] O'Neill, B., (1997), *Elementary Differential Geometry*. Academic Press, 2 edition.

[2] Dubrovin, B. A., Fomenko, A. T. and Novikov, S. P., (1984), *Modern Geometry - Methods and Applications*. Graduate Texts in Mathematics. Springer-Verlag.

[3] Gray, A., (1998), *Modern Differential Geometry of Curves and Surfaces with MATHEMATICA*. CRC Press.

[4] Goldgof, D. B., (1985), Motion estimation from three-dimensional data. Master's thesis, University of Illinois at Urbana-Champaign.

[5] Leon, S. J., (1986), *Linear Algebra with Applications*. Macmillan Publishing Company.

[6] Douros, I. and Buxton, B. F., (2002) Three-dimensional surface curvature estimation using quadric surface patches. *Numerisation 3D - Scanning*.

[7] Hartmann, E., (1995), Blending an implicit with a parametric surface. *Computer Aided Geometric Design*, Vol. 12.

[8] Yuen, P., Khalili, N. and Mokhtarian, F., (1999), Curvature estimation on smoothed 3-d meshes. *British Machine Vision Conference*, September.

[9] F. Mokhtarian, N. Khalili, and P. Yuen, (2001), Mult-scale free-form 3d object recognition using 3d models. *Image and Vision Computing*, Vol. 19.

[10] Falcidieno, B. and Spagnuolo, M., (1992), Polyhedral surface decomposition based on curvature analysis. *International Workshop on Modern Geometric Computing for Visualization*, Springer-Verlag, June, 263--269.

[11] Zhou, L. and Pang, A., Metrics and visualization tools for surface mesh comparison. http://www.cse.ucsc.edu/research/slvg/mesh.html.

[12] Desburn, M., Meyer, M., Schröder, P. and Barr, A., (1999), Implicit fairing of irregular meshes using diffusion and curvature flow. *SIGGRAPH99*, Addison Wesley, 317--324.

[13] Schneider, R. and Kobbelt, L., (2001), Geometric fairing of irregular meshes for free-form surface design. *Computer Aided Geometric Design*, Vol. 18(4).

[14] Mangan, A. P. and Whitaker, R. T., (1999), Partitioning 3d surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5(4).

[15] Besl, P. J. and Jain, R. C., (1988), Sementation through variable-order surface fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10(2).

[16] Koenderink, J. J. and Doorn, A. J., (1992), Surface shape and curvature scales. *Image and Vision Computing*, Vol. 10(8).