

# Mesh Segmentation Based on Local Geometric Properties

Hyoungeok B. Kim<sup>†</sup> and Hosook Kim<sup>††</sup>,

Department of Multimedia Engineering, Dong-Eui University, Busan, Rep. of Korea  
Department of Computer Science, Korea Science Academy of KAIST, Busan, Rep. of Korea

## Summary

In this paper, we present a new method of mesh segmentation based on the local geometric properties such as edge curvature and neighborhood representation. First of all, we review the concept of vertex curvature and convex combination with respect to neighborhood vertices. We propose usable measures to analyze the local shape of a triangular mesh such as vertex normal vector, edge curvature, and face curvature. They play an important role of mesh segmentation.

## Key words:

*Motion Editing, Mesh Segmentation, Local Geometric Properties.*

## 1. Introduction

Cutting up an irregular mesh into simpler sub-meshes benefits any algorithms in computer graphics, in areas as diverse as modeling, compression, simplification, 3D shape retrieval, and texture mapping [1-5]. In general, the most of complicated objects dealt in the field of computer graphics is obtained by using 3D scanners, or generated by operating 3D modeling tools. Each object is represented by a mesh which includes the position information of vertices and the connectivity of them. According to the demand of users, the geometric operations such as mesh segmentation, simplification, and smoothing are applied to the objects. In order to accomplish such geometric operations efficiently, we have to find out rightly the local properties of the mesh such as normal vector and curvature.

The right estimation of curvatures in triangular meshes plays important role in many applications such as surface segmentation, anisotropic re-meshing or rendering. A lot of efforts have been devoted to this problem [6-12]. Popular methods include quadratic fitting, where the estimated curvature is the one of the quadratic which locally fits a certain neighborhood of a vertex [6,7]. Goldfeather proposed the use of a cubic approximation scheme that takes into account vertex normal vectors in the 1-ring [7]. The accuracy of these curvature estimations is dependent of that of fitting. If the one-ring neighborhood has many vertices or has an oscillated shape,

then the approximated surface does not resemble the local shape and these estimations may yield a high error. Other methods typically consider some definition of curvature that can be extended to the polyhedral setting. These methods compute Gaussian curvature and Mean curvature based on the Gauss-Bonnet theory and Euler theory. Taubin presented a method to estimate the tensor of curvature of a surface at vertices of a mesh [6]. Watanabe proposed a simple method of estimating the principal curvatures of a discrete surface [9]. Meyer et. al proposed a discrete analog of the Laplace-Beltrami operator to estimate the discrete curvature [10]. Although the most of previous normal vector computation algorithms may approximate the normal vector in a given tolerance, they are not sufficient to be used as a measure in order to analyze the local shape. In this paper, we propose a concept of curvature, face curvature, which is useful to analyze the local shape efficiently and thus apply them to mesh segmentation.

Rendering is the process of generating an image from a model, by means of computer programs. It would contain geometry, viewpoint, texture, lighting, and shading information. Especially, the shading technology plays an important role of speeding up the rendering pipeline. In the shading methodology, there are three standard shading methods, flat shading, Gouraud shading, and Phong shading. Among them, Phong shading is effective to highlighting. In general, the traditional algorithm of Phong Shading is sensitive to the order of interpolation and the computation of normal vector of vertices. In this paper, we propose a new algorithm of Phong shading, which is based on the normal vector computation [13] and mean value coordinates [14].

The remainder of the paper is as follows. In Section 2 we review the properties of normal vector, vertex curvature, and convex combination with respect to neighborhoods. In Section 3 we introduce edge curvature which may play an important role in a variety of geometric operations. The new algorithm of mesh segmentation is explained in Section 4. Section 5 provides a summary and discusses some future work.

## 2. Geometric Tools

### 2.1 Vertex Normal Vector

We have been often asked a question “Can we represent a point inside a polygon as a linear combination of the vertices of the polygon uniquely?” If possible, we may represent all of information of points inside a polygon with the information of the vertices. There is a convex combination approach as a general solution of the problem. Specially, a representative solution of the approach is the mean value coordinate proposed by Floater [14].

$$P = \sum_{i=1}^m \lambda_i U_i, \sum_{i=1}^m \lambda_i = 1, \lambda_i \geq 0. \quad (1)$$

Equation (1) shows a general form of the convex combination approach that is a linear combination of the vertices  $U_i, i = 1, \dots, m$  of a given polygon for an arbitrary interior point  $P$  of a polygon as shown in Figure 3. Floater defined the coefficients of the linear combination satisfying the above equation by the following manner:

$$\lambda_i = \frac{\omega_j}{\sum_{k=1}^m \omega_k}, \quad \omega_j = \frac{\tan(\frac{\alpha_{j-1}}{2}) + \tan(\frac{\alpha_j}{2})}{\|U_j - P\|},$$

where  $\alpha_j = \angle U_j P U_{j+1}$ .

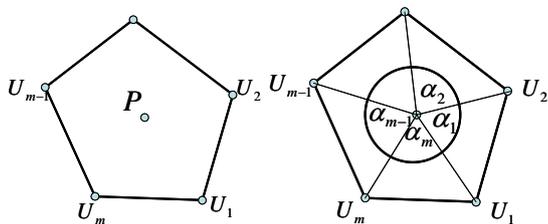


Fig. 1: Configuration of mean value coordinates

Now, we are ready to explain the computation of normal vector of a vertex based on the mean value coordinate. In general, the normal vector of a vertex in meshes cannot be determined uniquely because there are infinitely many tangent planes. So we have to approximate the normal vector. Let us consider the configuration of the local geometry of the vertex. Each face adjacent to the vertex has only one unit normal vector and the vertex is surrounded with the faces. This configuration becomes a clue to derive a new computation of a normal vector of a vertex. The new method runs as follows. We assume that a vertex  $V$  has  $m$  adjacent vertices  $V_1, V_2, \dots, V_m$ . Then the vertex  $V$  is surrounded with faces  $f_j = \Delta V_j V V_{j+1}, i = 1, \dots, m$ . We denote the unit normal vector and the

center of a face  $f_j$  by  $nf_j$  and  $cf_j$ , respectively, as shown in Figure 2.

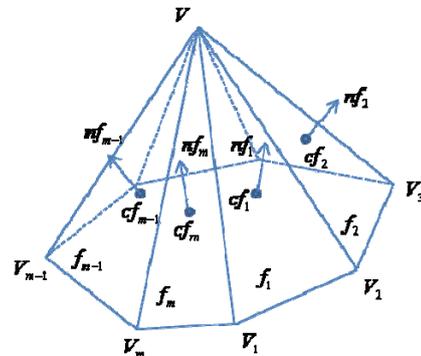


Fig. 2: Configuration of one-ring neighborhood of  $V$

In general, it may be understood that the center of a face is suited for the post where the normal vector of the face appears. So, we construct a mesh by connecting the vertex  $V$  with each center  $cf_j$  and the center  $cf_j$  with the consecutive center  $cf_{j+1}$  in order to obtain the relationship between normal vectors of the vertex and the faces, as shown in Figure 3. Here,  $\beta_j$  is the angle generated by the three consecutive vertices  $cf_j, V, cf_{j+1}$ .

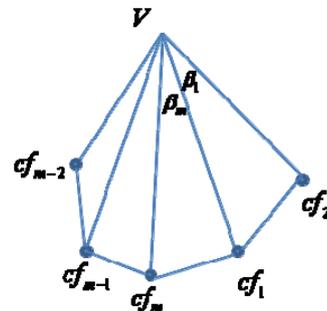


Fig. 3: New local mesh based on graph duality

Now, we apply a conformal mapping to the new local mesh by the following manner: the vertices  $V$  and  $cf_i$  are mapped to the vertices  $P$  and  $U_i$ , respectively, satisfying the following conditions

$$\|PU_i\| = \|Vcf_i\|, \alpha_i = \frac{2\pi}{\sum_{j=1}^m \beta_j}.$$

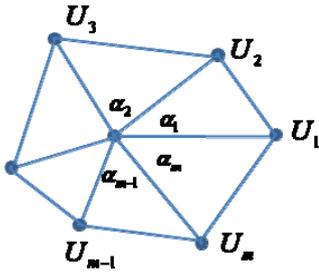


Fig. 4: Conformal mapped configuration

We define a new normal vector based on the mean value coordinates and a local conformal mapping by the following manner:

$$nV = \frac{\sum_{i=1}^m \lambda_i n f_i}{\left\| \sum_{i=1}^m \lambda_i n f_i \right\|}, \quad \lambda_i = \frac{\omega_i}{\sum_{k=1}^m \omega_k},$$

where

$$\omega_k = \frac{\tan\left(\frac{\alpha_{k-1}}{2}\right) + \tan\left(\frac{\alpha_k}{2}\right)}{\|U_k - P\|}$$

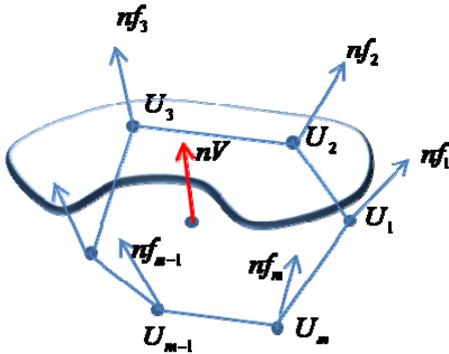


Fig. 5: Vector field interpolation

Figure 5 shows a vector field interpolation which may be understood as the meaning of new normal vector computation.

### 2.2 Vertex Curvature

In general, the curvature at a vertex on a mesh may not be computed easily because there are many surfaces interpolating the vertex and its one-ring neighbors. Moreover we cannot understand that the surfaces reflect well the local shape of the one-ring neighborhood though the surfaces interpolate the vertex and the neighbors. Most of the previous curvature estimations directly compute the sectional discrete curvature according to the adjacent edges [6,810]. All of them derive the sectional curvature

by using a Taylor series of a curve which interpolates two vertices. Let  $p$  and  $p_i$  be a given vertex and its adjacent vertex, respectively and let  $g(s)$  be a continuous curve passing through the two vertices such that  $g(0) = p$ ,  $g(s) = p_i$ .

Then the curve may be represented by its Taylor series:

$$g(s) = g(0) + \frac{g'(0)}{1!}s + \frac{g''(0)}{2!}s^2 + O(s^3)$$

$$g(s) = g(0) + T \cdot s + \frac{N}{2} \cdot \kappa(p) \cdot s^2 + O(s^3).$$

Here,  $T$  and  $N$  are the unit tangent vector and the unit normal vector, respectively, and  $\kappa(p)$  is the sectional curvature of  $g(s)$  at point  $p$  to the direction  $pp_i$ . By applying the inner product with  $N$  to both sides, we may get the following equation:

$$N \cdot (g(s) - p) \approx s(N \cdot T) + s^2 \frac{\kappa(p)}{2}(N \cdot N).$$

Because  $N \cdot T = 0$  and  $N \cdot N = 1$ , the above equation may be changed to the following simple equation:

$$\kappa(p) \approx \frac{2[N \cdot (g(s) - p)]}{s^2}.$$

Since  $g(s) = p_i$  the above equation becomes to the following equation

$$\kappa(p) \approx \frac{2[N \cdot (p_i - p)]}{s^2}.$$

Here, we have a problem how to derive the parameter  $s$ . The previous methods use the parameter as the length of edge  $pp_i$  such that

$$\kappa_c(p) \approx \frac{2[N \cdot (p_i - p)]}{|p_i - p|^2}.$$

However it is not right. We picked a hole of such parameterization and proposed a new method in [KIM]. Now, we review the parameterization shortly. The new discrete curvature estimation is based on the parabola interpolation. We adopt a quadratic Bezier curve as an interpolating curve because it has a good geometric property. Let  $p_i^-$  be the point such that

$$pp_i^- + pp_i = -2(pp_i \cdot N)N,$$

as shown in Fig 6, where  $N$  is the normal vector of the vertex  $p$ .

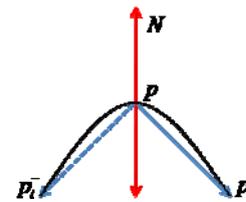


Fig. 6: Configuration of local shape

The general form of a quadratic Bezier curve is as follows:

$$g(t) = p_i^- B_0^2(t) + p B_1^2(t) + p_i B_2^2(t)$$

Here,  $B_i^2(t)$  are the Bernstein polynomials such that

$$B_i^n(t) = \binom{n}{i} (1-t)^{n-i} t^i$$

The curvature at  $p$  is the same as one of  $g(t)$  at  $t=0.5$ , so it is computed by the following manner:

$$\kappa_p(0.5) = \frac{\|g''(0.5) \times g'(0.5)\|}{\|g'(0.5)\|^3}$$

$$\kappa_p(0.5) = \frac{\|4(p_i^- - 2p + p_i) \times (p_i - p_i^-)\|}{\|(p_i - p_i^-)\|^3}$$

By exploiting the geometric properties, the curvature of the vertex  $p$  becomes

$$\kappa_p(p) = \frac{2\|N \cdot (p_i - p)\|}{\|(p_i - p) - N \cdot (p_i - p)N\|^2} \quad (2)$$

### 3. Edge Cross Curvature

Now, we are ready to define the edge cross curvature with respect to edges of a triangular mesh in 3D space. First of all, for a given edge  $AB$  we consider two adjacent triangles  $ABC$ ,  $ABD$  which have the common edge  $AB$  as shown in Figure 7.

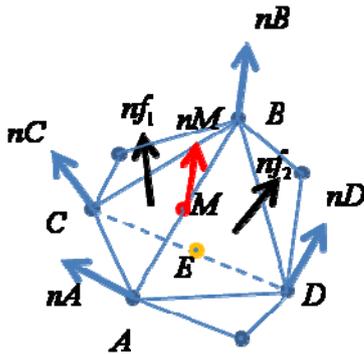


Fig. 7: Edge cross curvature

Let  $M$  be the middle point of the edge  $AB$ . We define the normal vector  $nM$  at the middle point  $M$  by the following manner:

$$nM = \frac{nf_1 + nf_2 + nA + nB}{4}$$

Here,  $nf_1$  and  $nf_2$  are the unit normal vectors of the triangles  $ABC$  and  $ABD$ , respectively.  $nA$  and  $nB$  the unit normal vectors of vertices  $A$  and  $B$ , respectively. Let  $E$  be the middle point of  $C$  and  $D$ . We define a vector  $nE$  as the unit vector from point  $E$  to point  $M$ :

$$nE = \frac{EM}{\|EM\|}$$

By plugging the three points  $C$ ,  $M$ , and  $D$  into the three points form of Equation (2), we obtain the following curvature of  $M$ :

$$\kappa(M) = \frac{\|4(C - 2M + D) \times (D - C)\|}{\|D - C\|^3}$$

Now, we define the edge cross curvature of edge  $AB$  as

$$\kappa_{AB} = \kappa(M)(nM \cdot nE)$$

### 4. Mesh Segmentation

The goal of mesh segmentation is to efficiently analyze the local shape of meshes and thus classify the faces into three clusters: convex, concave, and saddle regions. In general, we may easily compute the curvature at points on smooth surfaces and thus such classification is not hard work. However, it is very difficult to find out what the type of the local region is in discrete space such as meshes. We think that the edge cross curvature may become a clue to region classification.

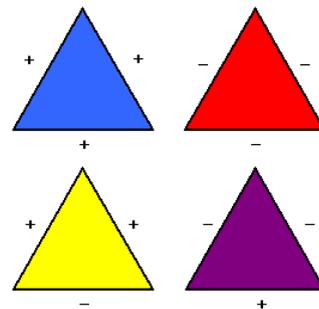


Fig. 8: Triangle classification based on edge cross curvature

The sign of the value of edge cross curvature means the convexity or the concavity of a local region which contains two adjacent triangles with a common edge: if the sign of  $\kappa_{AB}$  is positive, the region is of convex. Otherwise, it is of concave. It is insufficient to analyze the convexity of a face. So we need a measure to compute the convexity of faces. Since each edge has its own edge cross curvature,

we investigate the value for all of three edges of a face. There are only four cases as shown in Figure 8. If all of three edges have the edge cross curvature of positive, then we may understand that the triangle is on a convex region. If all of three edges have the edge cross curvature of negative, then the triangle is on a concave region. If two edges have the same sign and the other has the different sign, then the triangle may be considered to be on a saddle region.

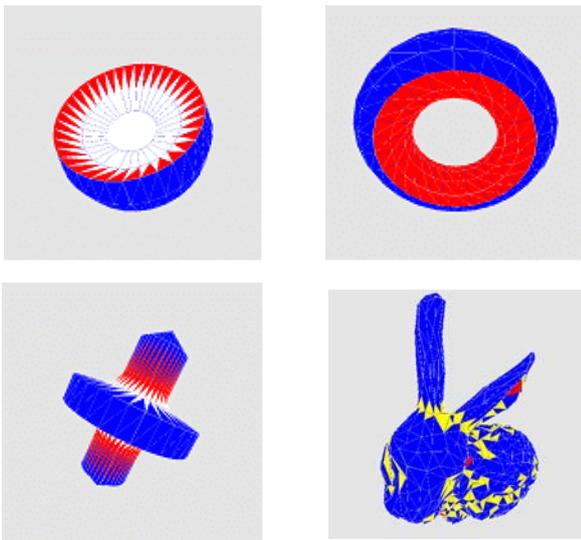


Fig. 9: Examples of mesh segmentation

Figure 9 shows the examples of mesh segmentation. The color of white means a region of concavity, the color of blue means a region of convexity. The colors of red and yellow indicate that the region is on a saddle region. The more dense the mesh is, the more the quality of our mesh segmentation is.

## 5. Conclusion

The contribution of this paper is to present an efficient method of mesh segmentation. Our method is based on mean value coordinate and edge curvature. First of all, we proposed a new computation of the normal vector of a vertex, which is derived by using the mean value coordinate. The computation is also available to meshes of arbitrary topology. Secondly, we derived a new sectional curvature which uses the different parameter with the previous methods. The parameter explains the right analysis of local configuration of a vertex. Exploiting this sectional curvature, we could derive a new concept of edge curvature and propose a method of mesh

segmentation. In future, we will develop a more efficient algorithm which is applicable to meshes of arbitrary.

## Acknowledgments

This work was supported by Dong-Eui University Grant (2008AA201).

## References

- [1] Funkhouser, T., Kazhdan, M., Shilane, P., Min, P., Kiefer, W., Tal, A., Rusinkiewicz, S., Dobkin, D.: *Modeling by example*. ACM Trans. Graph. (SIGGRAPH) **23**(3), 652–663 (2004)
- [2] Karni, Z., Gotsman, C., *Spectral compression of mesh geometry*. In: Proceedings of SIGGRAPH 2000, pp. 279–286 (2000)
- [3] Garland, M., Willmott, A., Heckbert, P, *Hierarchical face clustering on polygonal surfaces*. In: Proceedings of ACM Symposium on Interactive 3D Graphics, pp. 49–58 (2001)
- [4] Zuckerberger, E., Tal, A., Shlafman, S., *Polyhedral surface composition with applications*, Comput. Graph. **26**(5), 733–743 (2002).
- [5] Levy, B., Petitjean, S., Ray, N., Maillot, J., *Least squares conformal maps for automatic texture atlas generation*. In Proceedings of SIGGRAPH, 2002, pp. 362–371 (2002)
- [6] G. Taubin, *Estimating the tensor of curvature of a surface from a polyhedral approximation*, Proceeding of the Fifth International Conference on Computer Vision, pp 902-907, 1995
- [7] Sylvain Petitjean, *A Survey of Methods for Recovering Quadrics in Triangle Meshes*, ACM Computing Surveys, Vol. 34, No. 2, pp 211–262 (2002).
- [8] J. Goldfeather and V. Interrante, *A novel cubic-order algorithm for approximating principal direction vectors*, ACM Transaction on Graphics, Vol. 23, No. 1, pp 45–63 (2004)
- [9] Kouki Watanabe and Alexander G. Belyaev, *Detection of Salient Curvatures Features on Polygonal Surfaces*, Computer Graphics Forum, Vol. 20, No. 3, (2001)
- [10] Mark Meyer, Mathieu Desbrun, Peter Schroder, and Alan H. Barr, *Discrete Differential Geometry Operators for Triangulated 2-Manifolds*, Proc of VisMath, (2002)
- [11] Holger Theisel, Christian Rossl, Rhaleb Zayer, and Hans-Peter Seidel, *Normal Based Estimation of the Curvature Tensor for Triangular Meshes*, Proc. of Pacific Graphics'04, pp 288–297 (2004)
- [12] Tatiana Surazhsky, Evgeny Magid, Octavian Soldea, Gershon Elber, and Ehud Rivlin, *A Comparison of Gaussian and Mean Curvatures Estimation Methods on Triangular Meshes*, Proc. of 2003 IEEE International Conference on Robotics and Automation, (2003)
- [13] Hyoungseok Kim and Hosook Kim, *New Computation of Normal Vector and Curvatures*, WSEAS Transactions on Computers, Vol. 8, No. 10, pp 1661-1670, 2009
- [14] Michael S. Floater, *Mean value coordinates*, Comp. Aided Geom. Design, Vol. 20, p 19-27, 2003



**Hyoungseok B. Kim** received the M.S. and Ph.D. degrees in department of Mathematics from Korea Advanced Institute of Science and Technology in 1992 and 1998, respectively. During 1998-1999, he was a researcher in Electronics and Telecommunications Research Institute (ETRI). He is an associate professor in department of Multimedia Engineering at Dong-Eui University from 1999. His interesting research fields are computer graphics, geometric modeling, fluid simulation, and applied mathematics.



**Hosook Kim** received the M.S. and Ph.D. degrees in department of Computer Engineering from Ewha Womans University in 1999 and 2005, respectively. She is an associate professor in department of computer science at Korea Science Academy of KAIST. Her interesting research fields are database, data mining, computer graphics, and mobile technology.