

## Surface Parameterization Based on Circular Centering Property

Hyoungeok Kim

*Department of Applied SW engineering, Dongeui University, Korea*  
*hskim@deu.ac.kr*

### **Abstract**

*The parametrization of the surface plays an important role in the 3D computer graphics and computer vision. Especially, it is essential in 3D mesh editing that manipulates the position of 3D objects on a computer monitor. Because computer monitors are two-dimensional, while vertices of objects that actually want to change are in a three-dimensional virtual space. In this study, we propose a new type of parametrization method that takes advantage of the special properties of circles that are common to local parameterization. Experimental results show that our parameterization method is more efficient in terms of time and space than the standard parameterization method that was designed by Floater et al.*

**Keywords:** *Surface Parameterization, Shape-preserving Parameterization, Circular Centering Property*

### **1. Introduction**

The parametrization of a surface can be seen as a one-to-one mapping from the surface of the curved surface in three-dimensional space to the appropriate domain on the two-dimensional plane. Generally, the surface of a three-dimensional surface is represented by a triangular mesh, and the mapping is piecewise linear. Therefore, from the viewpoint of graph theory, it can be thought of as a mapping between graphs. .

Surface parametrization plays an important role in geometric modeling, computer graphics, and computer vision. In order to improve the visual quality of a polygon model, it is applied as a very important technical element in texture mapping used in computer graphics. Surface approximation and reconstruction are drawing attention in the field of computer vision due to the demand for 3D scanning techniques and efficient compression methods of increasingly complex triangulation. Bump mapping stores small deviations of the point deflection normal of the smooth underlying surface and uses a shaken version during shading [1].

Normal mapping [2, 3] is a similar technique that directly replaces normal vectors rather than storing perturbations. The mapping between the surfaces of two surfaces is used to transfer the details from one surface to another, or to allow interpolation between the shapes and shapes of several objects [4, 5, 6]. Biermann et al. [7] used local parameterization to facilitate cutting and pasting details between models in the field of mesh editing. Sorkine et al. L'evy [8] uses local parameterization for mesh synthesis in a similar way.

Most of the studies related to surface parameterization use Floater's algorithm [9] basically. However, even though the algorithms have excellent results in both geometric and mathematical aspects, they are still lacking in terms of computer science, namely time complexity and spatial complexity. In this paper, we propose a new solution which may improve these problems. The local parameterization is based on the relationship between a given point and its neighbors. We use a centering property of a circle in order to reduce the time and space complexities.

## 2. Preliminaries

Before describing the research related to shape preservation parameterization, we will briefly introduce the basic definitions of graph theory that will be used in the research. A simple graph  $G = G(V, E)$  is defined as a set of vertices  $V$  and a set of edges  $E$ , where  $V = \{i \mid i = 1, \dots, N\}$  and  $E = \{(i, j) \mid i \neq j, i, j = 1, \dots, N\}$ . When an edge  $(i, j) \in E$  is present, both ends of the segment are called *neighbors* of the other end, and *the degree* of a vertex is defined as the number of neighbors of the vertex. If all the vertices and edges of a graph  $H$  are the vertices and edges of  $G$ ,  $H$  is called a *subgraph* of  $G$ .

When a graph satisfies the following conditions, it is called a plane graph:

- (i) Each vertex corresponds to a point on the plane.
- (ii) Each line segment  $(i, j)$  corresponds to a curve whose end point is  $i$  and  $j$ .
- (iii) The intersection of the curves occurs only at the end point.

In a plane graph, each partial area divided by curves is called a *face*, and a plane graph having a set of vertices  $V$ , a set of edges  $E$ , and a set of faces  $F$  is denoted by  $P(V, E, F)$ . The face with an infinite area is referred to as the *outer face*, and the curved lines touching this face are referred to as the *boundary* of the graph and it is denoted by  $\partial G$ . In particular, when  $\partial G$  is a single closed curve, it is called a *simply connected graph*. When all faces of a graph consist of three edges, the graph is called a *triangulated graph*. It is assumed that all graphs to be covered in this research will be triangulated. *Surface Triangulation* refers to a simple connected graph of triangles in  $R^3$ , and is called *Plane Triangulation*, especially when the graph is on  $R^2$ .

The boundary of a plane graph is a single closed curve on the plane, so one can define the orientation of the boundary. In general, there are two orientations: clockwise and counterclockwise. We choose the direction of all plane graphs as counterclockwise. This can be used to define the isomorphic relationship between two triangulated plane graphs  $G_1$  and  $G_2$ . The two graphs are called isomorphic when the vertices, edges, and faces of all the subsets correspond to each other as follows: The corresponding edges connect the corresponding vertices, the corresponding edges of the corresponding faces have corresponding edges, and the counterclockwise boundaries of the two graphs have the same array of vertices.

We now define a parameterization, which is the main content of this paper. Parameterization is the process of finding a triangulated planar graph that is the same as a given triangulated surface, or its graph. The name parametrization is a one given to this process by which the surface of space can be represented by its own sculptural notation for both parameters  $u$  and  $v$ . Floater proposed a method of parameterization for arbitrary triangulated surfaces as follows[9]:

Let  $G(S, X)$  be a surface triangulation in  $R^3$  which is isomorphic to a plane graph  $G(V, E, F)$  and has the

set of nodes  $X = \{ \mathbf{x}_i = (x_i, y_i, z_i) \mid i = 1, \dots, N \}$ . Let  $x_1, \dots, x_n$  and  $x_{n+1}, \dots, x_N$  be the interior vertices and the boundary vertices of  $G(S, X)$ , respectively

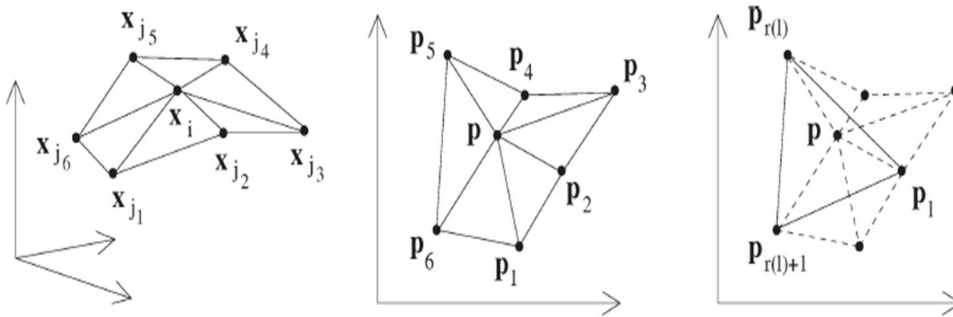
- (1) The boundary vertices correspond to the vertices  $u_{n+1}, \dots, u_N$  which are forming a counterclockwise convex polygon.
- (2) For  $1 \leq i \leq n$ , a real number  $\lambda_{i,j}$  satisfying the following condition is set.

$$\begin{aligned} \lambda_{i,j} &= \mathbf{0} & \text{if } (i,j) \notin E \\ \lambda_{i,j} &> \mathbf{0} & \text{if } (i,j) \in E \end{aligned} \quad \sum_{j=1}^N \lambda_{i,j} = \mathbf{1} \quad (1)$$

- (3) The interior vertices  $u_1, \dots, u_n$  are obtained by solving the following simultaneous linear equations.

$$\mathbf{u}_i = \sum_{j=1}^N \lambda_{i,j} \mathbf{u}_j \quad (2)$$

There is no limit to the exact method of obtaining the coefficients  $\lambda_{i,j}$  of the linear system by the above algorithm. If  $\lambda$  is kept constant regardless of the spatial distance of the original graph, each vertex is represented as the average of neighboring vertices, and this is called uniform parameterization. Since this parameterization is only affected by the adjacency states of the vertices, it does not match our research project, which has to rely on the geometry of 3D meshes. Based on the distance information between the vertices on the mesh, we need to set  $\lambda_{i,j}$  to avoid deforming the shape of the mesh as much as possible.



**Figure 1. Basic idea of Floater method**

The basic idea of shape conservation parameterization proposed by Floater [9] is as follows. Let  $S(G, X)$  be a surface parameterization for  $G = G(V, E, F)$ , and let  $G_i$  be a partial graph of  $G$  which consists of its  $i$ -th vertex  $x_i$  and its neighbors  $x_{j_1}, x_{j_2}, \dots, x_{j_{d_i}}$ , where the neighbors are given in order counterclockwise. For  $X_i = \{x_i, x_{j_1}, x_{j_2}, \dots, x_{j_{d_i}}\}$  we think about the following process.

- (1) For  $S_i = S_i(G_i, X_i)$ , we consider a local shape-preserving parameterization  $P_i$  that maps  $x_i$  to  $p \in R^2$ , and  $x_{j_1}, x_{j_2}, \dots, x_{j_{d_i}}$  to  $p_1, p_2, \dots, p_{d_i}$ .
- (2) For  $j$  such that  $(i, j) \in E$ , we set the coefficient  $\lambda_{i,j}$  satisfying the following condition

$$p = \sum_{k=1}^{d_i} \lambda_{i,j_k} p_k, \quad \sum_{k=1}^{d_i} \lambda_{i,j_k} = 1.$$

(3) When we denote the angle between two vectors  $a - b$  and  $c - b$  by  $\text{ang}(a, b, c)$ , we have to compute  $p, p_1, p_2, \dots, p_{d_i}$  satisfying the following conditions

$$\begin{aligned} \|p_k - p\| &= \|x_{j_k} - x_i\| \\ \text{ang}(p_k, p, p_{k+1}) &= 2\pi \text{ang}(x_{j_k}, x_i, x_{j_{k+1}}) / \theta_i \end{aligned} \quad (3)$$

where  $\theta_i = \sum_{k=1}^{d_i} \text{ang}(x_{j_k}, x_i, x_{j_{k+1}})$ ,  $x_{j_{d_i+1}} = x_{j_1}$  and  $p_{d_i+1} = p_1$ .

### 3. New shape-preserving parameterization

Considering the method of shape preservation parameterization, we must bear in mind the following assumptions that the method must satisfy. If the original graph is in the plane and the boundaries are equally given, then the internal points must also be matched. Of course, the method proposed by Floater satisfies this assumption, but as already mentioned, the result is not smoothly given for all vertices. The proposed shape conservation parameterization method is given smoothly for all vertices and is based on the basic fact that the average of all the points of a circle is the center of a circle. That is, if a unit circle is expressed as  $(\cos t, \sin t)$  for the parameter  $t$ , then we may derive the following fact that  $\int_0^{2\pi} (\cos t, \sin t) dt = (0, 0)$ .

The basic idea of this is to find the area of the arc that has the same effect as the influence of neighboring neighbors to each vertex to be parameterized. Consider a graph  $G = G(V, E, F)$  in which all vertices are on the  $xy$ -plane. Suppose that a parametrization of a partial graph  $G_i$  consisting of an arbitrary inner point  $x_i$  and its neighboring vertices is devised so that the corresponding inner point  $x$  also corresponds to the same position when the boundary is given as it is. When this method is applied to the entire graph, if the boundary of the graph is given as it is, the entire graph is parameterized without changing. Therefore, it is sufficient to deal with a graph consisting of one internal vertex  $x$  and neighboring vertices  $x_1, x_2, \dots, x_d$ . Let  $u_1, u_2, \dots, u_d$  be the boundary of the graph to be parameterized, and  $u$  be the parameterized coordinate of the interior vertex to be obtained using  $\lambda$ . Now the task is to find  $\lambda$  to make  $u = x$ .

Since  $\sum_{j=1}^d \lambda_{i,j} = 1$ , the parameterization method of the floater is independent of the coordinate system. Assume that  $x$  is at the origin without loss of generality, and define  $\theta_j = \text{ang}(x_j, x, x_{j+1})$  theta. Here,  $x_{d+1} = x_1$ ,  $x_d = x_0$ . Now we will set  $\lambda_{i,j}$  to:

$$\lambda_{i,j} \sim \left( \tan \frac{\theta_j}{2} + \tan \frac{\theta_{j-1}}{2} \right), \quad \sum_{j=1}^d \lambda_{i,j} = 1 \quad (4)$$

To confirm that this  $\lambda_{i,j}$  is a correct shape preservation parameterization, we can use this  $\lambda_{i,j}$  to show that  $u$  is set at the origin. First of all, we define a unit middle vector  $v_j = \frac{u_j + u_{j+1}}{\|u_j + u_{j+1}\|}$ , where  $u_{d+1} = u_1$ ,  $u_d = u_0$ .  $v_j$  is a unit vector that bisects the angle formed by two vectors  $x_j - x$  and  $x_{j+1} - x$ . By using this vector and Equation (4), we have the following relation:

$$\begin{aligned}
u &= \sum_{j=1}^d \lambda_{i,j} u_j = \sum_{j=1}^d \left( \tan \frac{\theta_j}{2} + \tan \frac{\theta_{j-1}}{2} \right) u_j = \sum_{j=1}^d \left( \frac{\sin \frac{\theta_j}{2}}{\cos \frac{\theta_j}{2}} + \frac{\sin \frac{\theta_{j-1}}{2}}{\cos \frac{\theta_{j-1}}{2}} \right) u_j \\
&= \sum_{j=1}^d \left( \frac{2 \sin \frac{\theta_j}{2}}{\|u_j + u_{j+1}\|} + \frac{2 \sin \frac{\theta_{j-1}}{2}}{\|u_{j-1} + u_j\|} \right) u_j = \sum_{j=1}^d \frac{(u_j + u_{j+1})}{\|u_j + u_{j+1}\|} \left( 2 \sin \frac{\theta_j}{2} \right) \\
&= \sum_{j=1}^d v_j \int_{-\frac{\theta_j}{2}}^{\frac{\theta_j}{2}} \cos t \, dt = \sum_{j=1}^d \int_{-\frac{\theta_j}{2}}^{\frac{\theta_j}{2}} (\cos(t - \arg(v_j)), \sin((t - \arg(v_j)))) \, dt \\
&= \sum_{j=1}^d \int_{\arg(v_j) - \frac{\theta_j}{2}}^{\arg(v_j) + \frac{\theta_j}{2}} (\cos t, \sin t) \, dt = \sum_{j=1}^d \int_{\arg(u_j)}^{\arg(u_{j+1})} (\cos t, \sin t) \, dt \\
&= \int_0^{2\pi} (\cos t, \sin t) \, dt = (0, 0) = x
\end{aligned}$$

This method is basically meaningful in that it can obtain smooth parameterization results for all vertices. Our method also greatly reduces the computational complexity of the linear system to obtain the coefficients used in the Floater method and the complexity of the memory space to store various variables accordingly.

We compare the execution times of the two algorithms for complex input data. The following table Table 1 shows the time required to perform two parameterizations of our method and Floater [9] for various data.

**Table 1. Comparison of execution time**

# of Data	Our method	Floater's method
60	40	50
120	120	140
180	251	321
240	511	601
300	912	1062
600	2323	2664

The parameterization algorithm proposed by Floater is 20 ~ 30% longer than our algorithm. This proves that our algorithm is more efficient than the algorithm used by Floater.

#### 4. Concluding Remarks

Virtual reality and augmented reality, which play a key role in the digital age, require three-dimensional computer graphics and, in particular, efficient space-mapping technology. In this paper, we have studied mesh parameterization method which is the core of editing technology of mesh data, which is the most basic part of 3D computer graphics. This technology is used in the fields of image processing, computer vision, broadcast video, and film. Our new method of shape preservation parameterization has dramatically reduced the execution speed while achieving almost the same effect as the currently used Floater's algorithm. In addition, we can confirm that space management is more important in terms of spatial and temporal aspects.

## References

- [1] J. F. Blinn, "Simulation of wrinkled surfaces," in *ACM SIGGRAPH '78*, pp. 286–292, 1978.
- [2] A. Sheffer, B. Lévy, M. Mogilnitsky, and A. Bogomyakov, "ABF++: Fast and robust angle based flattening," *ACM Transactions on Graphics*, vol. 24, no. 2, pp. 311–330, 2005.
- [3] Y. Lee, H. S. Kim, and S. Lee, "Mesh parameterization with a virtual boundary", *Computers and Graphics*, vol. 26, no. 5, pp. 677–686, 2002.
- [4] M. Alexa, "Recent advances in mesh morphing," *Computer Graphics Forum*, vol. 21, no. 2, pp. 173–196, 2002.
- [5] V. Kraevoy and A. Sheffer, "Cross-parameterization and compatible remeshing of 3D models," in *ACM SIGGRAPH 2004*, 2004.
- [6] A. Khodakovsky, N. Litke, and P. Schroder, "Globally smooth parameterizations with low distortion," in *ACM SIGGRAPH 2003*, pp. 350–357, 2003.
- [7] H. Biermann, I. Martin, F. Bernardini, and D. Zorin, "Cut-and-paste editing of multiresolution surfaces," in *ACM SIGGRAPH 2002*, pp. 312–321, 2002.
- [8] B. Levy, "Dual domain extrapolation," in *ACM SIGGRAPH 2003*, 2003.
- [9] M. Floater and K. Hormann, "Surface parameterization: A tutorial and survey," in *Advances in Multiresolution for Geometric Modelling*, pp. 157–186, 2005.