2005 International Symposium on Nonlinear
Theory and its Applications (NOLTA2005)
Bruges, Belgium, October 18-21, 2005

A Visually Interactive Deformation of Surfaces Defined by Linear Lie Algebra with Extraction of Invariants

Mariko KANEKO† Naoya SAGARA† Mitsunori MAKINO† Jinhui CHAO†

†Department of Information and System Engineering, Chuo University
1-13-27 Kasuga, Bunkyo-ku, Tokyo 112-8551 Japan
Email: kaneko@makino.ise.chuo-u.ac.jp, nsagara@makino.ise.chuo-u.ac.jp,
makino@m.ieice.org, jchao@ise.chuo-u.ac.jp

Abstract—In this paper, we propose an interactive deformation and representation of 3D surfaces. The surfaces are supposed to be defined by linear Lie algebra, which can describe shape from their own invariants. Then the Free-form deformation (FFD) is used for easy manipulation. By extraction of modified invariants from the deformed surface, it can be easily represented by computer graphics. The proposed method assists designers in getting inspiration.

1. INTRODUCTION

Since virtual reality (VR) technology has been assisting designers more and more, they have begun to use the technology even at initial process of shape design, which is required not to disturb their inspiration. For example, the CAVE system [1], a major VR equipment, shows shapes stereoscopically in its virtual world so that they can manipulate the virtual shapes floating in space. Such stereoscopic representation and manipulation needs big computation power. Therefore efficient combination between representation and manipulation should be developed as well as each efficient technology.

Some of authors has proposed a surface model and its representation, which is defined based on the Lie algebra [2][3][4]. It is noted that characteristics of a shape of surface, defined by the Lie algebra model, are represented by a few parameters called as "invariants." As a result, a certain of complicated and smooth objects can be defined by a set of the invariants. Since both of extraction from real objects and representation by computer graphics have been being studied, it is expected to connect virtual world with real world.

However, the model has a disadvantage that exact representation of a surface may require huge computational cost because of complicated and large calculation of many integral calculus curves on the surface. In such case, meshes of triangular patches usually substitutes for the original surface. Adaptive and efficient generation of meshes has been proposed in order not to generate meshes redundantly [5][6][7]. Therefore, efficient manipulation should be developed for the previously proposed surface model.

In this paper, we propose a visually interactive deformation of surfaces, which are defined by the surface model based on the linear Lie algebra. In the proposed method, invariants and some initial parameters are extracted from meshes deformed by the free-form deformation (FFD) [8]. Since the proposed method works rapidly, it does not harm designers' interactivity. Furthermore, since the method connects "invariant-base" information on shape before/after the deformation, we can easily deform surfaces repeatedly. The advantage is also benefit of telecollaboration in design process.

2. EXTRACTION OF INVARIANTS OF DEFORMED SURFACE

Lie group is a C^∞ class differentiable manifold with nice properties, which is a one-to-one correspondence between Lie groups and their Lie algebra, or between the global structure and the local information of their normal/tangent vector fields. The vector field L is represented as follows:

$$L = \left(\sum_{i=1}^{n} a_{1i}x_i, \quad \cdots, \quad \sum_{i=1}^{n} a_{ni}x_i \right) \begin{pmatrix} \frac{\partial}{\partial x_1} \\ \vdots \\ \frac{\partial}{\partial x_n} \end{pmatrix} = v^T \nabla. \quad (1)$$

A normal vector v of a point x on a surface given by linear Lie algebra is defined as follows:

$$v = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x^1 \\ \vdots \\ x^n \end{pmatrix} = Ax, \quad (2)$$

where A is a representation matrix, which A is obtained from sets of position and normal vector. The proposed method calculates a normal vector of a vertex n as an average of normal vectors on adjoining patches (see Fig.1).

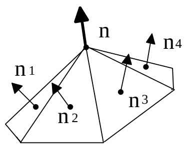

Figure 1: Normal vector of a vertex

In order to obtain the matrix A more quickly, it is efficient to limit vertices to inside a control area of FFD. Four sets of position and normal vector on a local area are necessary to calculation of A. More sets we can use, more exact result can be calculated. In the proposed method, more than four sets are used for the least-squares method. Additionally, a set of invariant λ_1, λ_2, λ_3 and the Euler angles θ, ϕ, ψ are obtained from the representation matrix A by the singular value decomposition.

3. REPRESENTATION OF SURFACE

An initial point of calculation on a surface plays an important role in determination of size of the surface, while a set of invariant determines its shape. Therefore a set of invariants and an initial point x_0 must be kept for representation. Then a series of points along an integral curve on the surface is generated by the followings:

$$x_{i+1} = x_i + \Delta t w_i, \tag{3}$$

where w_i is a tangent vector at x_i and Δt is assumed small enough (see Fig.2). A set of meshes is generated by a set of points, calculated by Eq.(2) repeatedly.

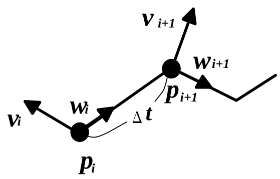

Figure 2: Generation of neighbor point

4. ALGORITHM

Here we show an outline of the porposed method:

Step1: Calculate the representation matrix from invariants.

Step2: Obtain the normal vector of a vertex by subsututting initial point to Eq.(2).

Step3: Generate a set of point by Eq.(3), and generate a set of meshe.

Step4: Deform the surface.

Step5: Calculate the average of normal vectors from adjoining meshes.

Step6: Take more than four positions and normal vectors, and normlize them.

Step7: Calculate the representation matrix from the sets by the least squeres method.

Step8: Obtain the invariant from representation matrix by the singular value decomposition.

From the above, invariants of deformed surfaces defined by Lie algebra can be extracted. Additionaly, the surface can be deformed repeatedly.

5. SIMULATION

We here show results by the proposed method. All simulation were done under the Table 1. Additionally, we used OpenGL for visualization. Table 2 shows the data (invariants and initial point).

Figure 3 is original shape. Figure 4 is represented from invariants of original shape. Figures 5 and 6 shows expansion and reduction of the original shape, respectively. Figures 7, 9 and 11 shows expansion to x, y, z axis direction by using FFD method. Figures 8, 10 and 12 shows represented shapes from invariants. The shape in Fig.3 consists of 10200 vertices. In this paper, a size of FFD control area covers target shape, thus invariants are obtained from all vertex of the surface. Since deformation of Figs 7, 9 and 11 takes 0.3 second, while interactive deformation is available.

Although invariants in Fig.4 differs from ones in Fig.3, it seems no difference between shapes in Figures 3 and 4. Figures 5 and 6 show that when the size of shape changes and also the value of initial point changes. Additionally, in Figures 8, 10 and 12, invariants and initial point are not same. In particular, we can see relationship of x, y, z axis with λ_1, λ_2, λ_3. Thus shape expands to direction small value of λ.

Invariants changes according to target shape, but to predict quantity of the change is difficult. In other words, we can deform the shape by changing the invariants, but it is difficult to guess a shape, from numerical value. Therefore when a user obtains arbitrary shape by deformation, it is effective to obtain invariants of the surface after deformation it.

Table 1: spec

OS	Microsoft Windows XP
CPU	Pentium4 CPU 2.60GHz
Memory	512MB RAM
Video Bord	NVIDIA GeForece FX5200

Table 2: Invariants and Initial Point

	Invariants ($\lambda_1, \lambda_2, \lambda_3, \theta, \phi, \psi$)	Initial Point (x, y, z)
Fig.3	1.0,1.0,1.0,0.0,0.0,0.0	300.0,600.0,100.0
Fig.4	1.0,1.0,1.0,0.0,0.0,0.1	300.0,600.0,100.0
Fig.5	1.0,1.0,1.0,0.0,0.8,-0.8	395.7,779.8,132.5
Fig.6	1.0,1.0,1.0,0.0,-1.1,1.1	204.3,420.2,67.5
Fig.8	1.0,2.0,2.0,0.0,-1.6,1.6	443.6,600.0,100.0
Fig.10	1.0,0.5,1.0,0.0,-1.0,1.0	300.0,869.7,100.0
Fig.12	1.0,1.0,0.5,0.0,0.2,-0.2	300.0,600.0,148.7

6. CONCLUSION

In this paper, we have proposed a visual interactive deformation of surfaces defined by linear Lie algebra and extraction of invariants.

The proposed method enables to obtain parameters, that determine uniformally deformed shape, easily. While our simulation focused on expansion in axis direction, the method is applicable to more complicated deformation. Subdivision technique may be necessary for such case. Additionaly, in order to reduce computational cost, we must select less and appropriate vertex at the calculation of representation matrix. Additionaly, implementation to a VR system will be discussed.

Acknowledgments

This paper is partially supported by Chuo University 21COE Program and Grant for Special Research, and Collaborative Research Grant of the Institute of Science and Engineering, Chuo University.

References

[1] C. Cruz-Neira, D.J. Sandin, T.A. DeFanti, R.V. Kenyon and J.C. Hart, "The CAVE: Audio Visual Experience Automatic Virtual Environment," *Communications of the ACM*, vol. 35, no. 6, pp. 65-72, June, 1992.

[2] Jinhui Chao, Akira Karasudani, Toshihiro Shimada, and Kenji Minowa, "Invariant Object Representation and Recognition Using Lie Algebras of Perceptual Vector Fields," *IEICE Trans.(D2)*, vol.J83-D2 no.9, pp. 1870-1878, Sep. 2000.

[3] Shin-ichi Kawano, and Mitsunori Makino, "A High Quality Triangular Meshing for Surface Defined by Linear Lie Algebra," *IEICE Trans.(D2)*, vol.J83-D2 no.12, pp. 2707-2715, Dec. 2000.

[4] Masaki Suzuki and Jinhui Chao, "Invariant Extraction and Segmentation of 3D Objects Using Linear Lie Algebra Models," *IEICE Trans. Inf. &Syst.*, vol.E85-D, no.8, pp.1306-1313, 2002.

[5] Hiroyasu Sano, Mitsunori Makino, and Jinhui Chao, "An Adaptive High Quality Mesh Generation for Surface Defined by Linear Lie Algebra," *Japan Society for Simulation Technology*, vol.20 no.3, pp. 251-258, 2001.

[6] Yu-ichi Yamamoto, Hiroyasu Sano and Mitsunori Makino, "A High Quality Mesh Generation with Automatic Differentiation for Surface Defined by Lie Algebra," *Proc. of NICOGRAPH/MULTIMEDIA '01*, pp. 129-134, 2001.

[7] Naoya Sagara, Meeko Kuwahara, Mitsunori Makino and Jinhui Chao, "An adaptive Mesh Generation of Surfaces Defined by Lie Algebra and its Visualization toward Intelligent Communication System," *Proc. of TENCON2004, CD-ROM*, 2004.

[8] Thomas W. Sederberg, Scott R. Parry, "Free-Form Deformation of Solid Geometric Models," *SIGGRAPH '86 Conf. Proc.*, vol.20, no.4, pp.151-160, 1986.

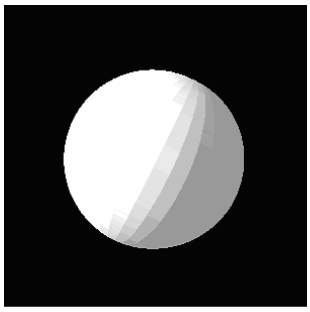

Figure 3: original shape

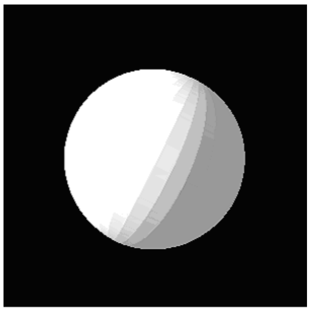

Figure 4: represented original shape

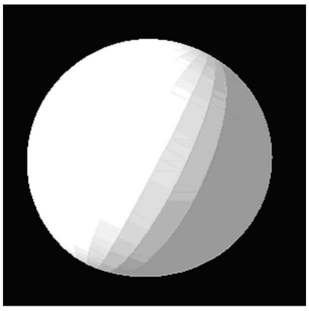

Figure 5: represented shape (expansion)

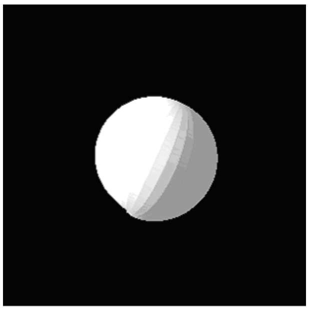

Figure 6: represented shape (reduction)

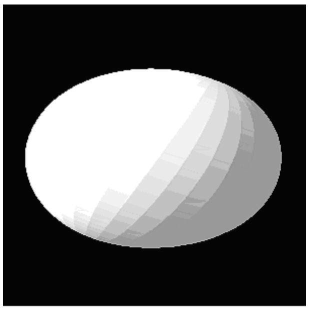

Figure 7: original shape (x_axis)

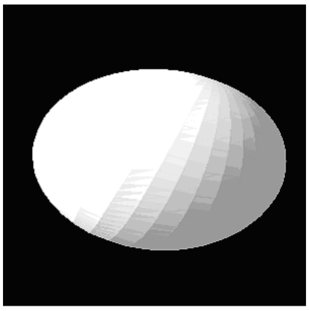

Figure 8: represented shape (x_axis)

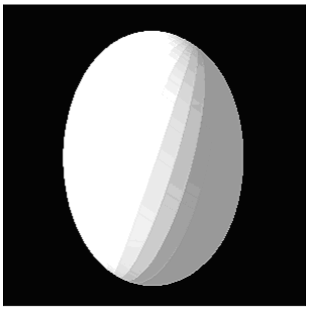

Figure 9: original shape (y_axis)

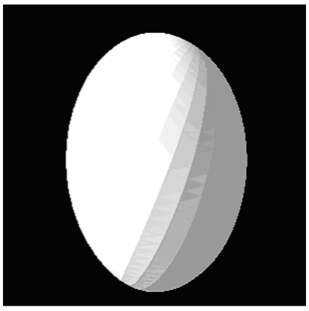

Figure 10: represented shape (y_axis)

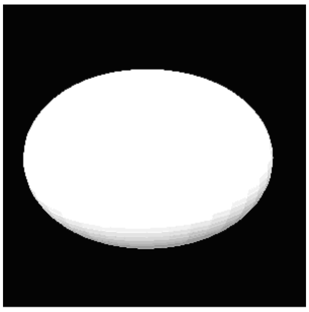

Figure 11: original shape (z_axis)

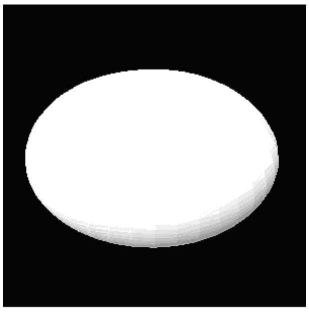

Figure 12: represented shape (z_axis)