Detail-preserving sculpting deformation

Wenbing Ge
Key Lab on Machine Perception and Intelligence of MOE
Peking University, Beijing, China
Email: dgewb@hotmail.com

Gang Xu
Hangzhou Dianzi University, Hangzhou, China
Project Galaad, INRIA Sophia-Antipolis, France
Email: xugangzju@gmail.com

Kin-chuen Hui
CAD Laboratory, Department of Mechanical and Automation
The Chinese University of Hong Kong, Shatin, Hong Kong

Guoping Wang
Key Lab on Machine Perception and Intelligence of MOE
Peking University, Beijing, China

Abstract

Sculpting deformation is a powerful tool to modify the shape of objects intuitively. However, the detail preserving problem has not been considered in sculpting deformation. In the deformation of a source object by pressing a primitive object against it, the source object is deformed while geometric details of the object should be maintained. In order to address this problem, we present a detail preserving sculpting deformation algorithm by using Laplacian coordinates. Based on the property of Laplacian coordinate, we propose two feature invariants to encode the Laplacian coordinate. Instead of mapping the source mesh to the primitive mesh, we map the smooth version of source mesh to the primitive mesh and use the Laplacian coordinates to encode the geometric details. When the smooth version of the source mesh is deformed, the Laplacian coordinates of the deformed mesh are computed for each vertex firstly and then the deformed mesh is reconstructed by solving a linear system that satisfies the reconstruction of the local details in least squares sense. Several examples are presented to show the effectiveness of the proposed approach.

1. Introduction

Surface editing is an important and time-consuming work in many fields such as computer-aided design (CAD), manufacturing, and computer animations. Besides the surface representation, editing methods and tools are also essential in an editing process. How to find a powerful and fast method to complete the editing tasks, is an key problem in surface editing.

NURBS has been a standard representation in CAD systems, but it is rather tedious when there are large number of patches of which continuity have to be maintained across their boundaries. Polygonal representations of 3D objects, and in particular, triangular meshes, have become prevalent in numerous application domains. There are some effective geometry processing and editing tools for triangular mesh. Sculpting deformation is a powerful tool to modify the shape of objects intuitively. When a source object is deformed by pressing a primitive object against it, the original object should be deformed into the shape of the primitive object while geometric details are maintained. However, there are few work on this problem from the view of detail-preserving.

In this paper, we propose a detail preserving sculpting deformation technique for surface editing. A tool surface, also called the primitive surface, is presented to modify the source surface in a way similar to a sculpting surface, the source surface is deformed into the shape of the primitive surface while maintaining the detail and topology of the source surface. See Figure 1 for an example of such an operation. The advantage of this technique is that you can keep the details when you modify the source surface by sculpting. The method is useful in computer animation and computer aided design.

2. Related work

In this section, we will present an overview of a number of sculpting deformation techniques and some related work on detail preserving deformations.

Sculpting deformation The approaches using mathematical representations that depend on the correlation with control meshes or the characteristic of the surface itself. It is not clear enough to the user how much the modification will affect the final shape. Angelidis et al. [1] present sweepers, a new class of space deformations suitable for interactive virtual sculpture. The artist describes a basic deformation as a path through which a tool is moved. The tool causes a deformation of the working shape along the path. Nevertheless, as a sculpting tool, it needs the power of pulling. Li et al. [2] introduce a method that deforms the surface directly with a virtual force along the path. The only difference between pull and push is the direction of force on the surface.

Detail preserving deformation Detail-preserving is a key problem in a deformation process. Differential coordi-

978-1-4244-3701-6/09/$25.00 ©2009 IEEE

Figure 1. Skull model hit by a fist. Left shows the undeformed skull model. Middle shows the skull model hit by a hand. Right shows the deformed skull model. Note the detail of the letters on the forehead is preserved in the deformation

nates based methods are good candidates for detail preserving deformation. Poisson meshes [3] manipulate gradients of the mesh's coordinate functions using local transforms and then reconstruct the surface from the Poisson equation. Laplacian coordinates represent surface details in another form, thus can be used in surface deformations[4], [5], [6], [7], [8]. Another surface representation is rotation-invariant coordinates[9], which consist of the tangential part and normal part invariant to rotation and translation, in the first and second discrete forms respectively. Multiresolution approaches[10], [11], [12], [13] enable detail-preserving deformations by decomposing the geometry into several levels of detail, represented as displacements in a local coordinate frame, and the coarse meshes act as control objects.

3. Detail preserving sculpting deformation

Detail preserving sculpting deformation can be described as: a source surface is hit by a tool surface: the source surface is deformed into the shape of the tool surface while maintaining the detail and topology of the original models. See Figure 1 for an example of such an operation.

3.1. Laplacian coordinates on meshes and their properties

Let $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ be a triangular mesh.$\boldsymbol{V}$ denotes the set of vertices on the mesh and $\boldsymbol{E}$ denotes the set of edges. Let $\boldsymbol{V} = (\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_n)$ describes absolute Cartesian coordinates of the vertices in $\boldsymbol{R}^3$. We define the differential coordinate of vertex $\boldsymbol{v}_i$ to be the difference between the absolute coordinate of $\boldsymbol{v}_i$ and its neighbors:

$$\boldsymbol{\delta}_i = \boldsymbol{D}(\boldsymbol{v}_i) = \boldsymbol{v}_i - \frac{1}{d_i} \sum_{j \in N_i} \boldsymbol{v}_j, \qquad (1)$$

where d_i is the valency of vertex $\boldsymbol{v}_i$. Here, we mostly consider the neighborhood $N(i)$ to be a one neighborhood ring, i.e. $N(i) = \{j | (i,j) \in \boldsymbol{E}\}$

Laplacian coordinates are invariant under translation, but sensitive to other linear transformation such as rotation and scale. In fact, note that in Eq.(1) Laplacian coordinate $\boldsymbol{\delta}_i$ is the linear combination of $\boldsymbol{v}_i$ and its neighbors, so the following properties are straightforward.

Property 1 If the vertex $\boldsymbol{v}_i$ and its neighbors are rotated by a matrix $\boldsymbol{R}_i$, the Laplacian coordinate after rotation is $\boldsymbol{R}_i \boldsymbol{\delta}_i$.

***Proof*.**

$$\boldsymbol{\delta}_i^{'} = \boldsymbol{R}_i \boldsymbol{v}_i - \frac{1}{d_i} \sum_{j \in N_i} \boldsymbol{R}_i \boldsymbol{v}_j = \boldsymbol{R}_i (\boldsymbol{v}_i - \frac{1}{d_i} \sum_{j \in N_i} \boldsymbol{v}_j) = \boldsymbol{R}_i \boldsymbol{\delta}_i. \quad \square$$

Property 2 The scaling factor of Laplacian coordinate is proportional to the square root of the scaling factor of the mesh's area.

***Proof*.** If we enlarge the mesh by n times, the area of the mesh is enlarged by n^2 times. The Laplacian coordinate is changed into

$$\boldsymbol{\delta}_i^{'} = n\boldsymbol{v}_i - n\frac{1}{d_i} \sum_{j \in N_i} \boldsymbol{v}_j = n(\boldsymbol{v}_i - \frac{1}{d_i} \sum_{j \in N_i} \boldsymbol{v}_j) = n\boldsymbol{\delta}_i,$$

That is, it is enlarged by n times. Hence, the scaling factor of Laplacian coordinate is proportional to the square root of the scaling factor of the mesh's area. $\square$

3.2. Primitive-based detail preserving deformation

Feature invariants We encode the detail of a surface based on the Laplacian coordinates. Let δ_i be the Laplacian coordinates of vertex i in $\boldsymbol{G}$. From property 1 and property 2, we define two feature invariants—the relative magnitude and relative direction of Laplacian coordinates as follows:

Relative magnitude: we define rm_i to be the relative magnitude of Laplacian coordinates at vertex i as

$$rm_i = \frac{\|\delta_i\|}{\sqrt{s_i}},$$

where s_i is the total area of the triangles lying in 1-neighbor region of vertex i on mesh ***B***. Mesh ***B*** is a smooth version of ***G*** without detail while sharing the same connectivity with ***G***.

Relative direction: we define rd_i to be the relative direction of Laplacian coordinates at vertex i as

$$rd_i = F_i^{-1}\delta_i/\|\delta_i\|,$$

where F_i is the frame matrix related to vertex i on mesh ***B***. It is defined as the triplet (b_1^i, b_2^i, N^i), where N_i is the normal at vertex i, b_1^i is a unit vector orthogonal to some edge e_{ij}, b_2^i is determined by the righthand product of N_i and b_1^i.

The value of rm_i and rd_i encode the detail of ***G***, which should be preserved during sculpting deformation.

Instead of mapping the source mesh to the primitive mesh, we map the smooth version of source mesh to the primitive mesh. Let ***G*** and ***B*** be the source surface and the smooth version of ***G*** without detail, respectively. The surface ***B*** is a low-frequency surface associated with ***G***, which can be generated by mesh smoothing[14]. Note that by using this smoothing method, ***B*** and ***G*** have the same connectivity, and the amount of smoothing is a user-defined parameter, which depends on the material of the source object.

We edit the mesh ***B*** and modify it into another smooth mesh ***U***. The mesh ***B*** and mesh ***U*** share the same connectivity, but have different geometries. We can recover the mesh with original detail based on mesh ***U***, since each corresponding vertex on mesh ***B*** is encoded with invariant detail representation(relative magnitude and relative direction). We use the above steps to reconstruct the deformed mesh ***G'*** that preserves the detail of the original mesh.

Hence, the main steps of our surface-based detail-preserving deformation algorithm are:

1) Compute the relative magnitude rm_i and relative direction rd_i of source object ***G***;
2) Use the method in [14] to obtain the smooth version ***B*** of ***G***;
3) Edit the mesh ***B*** and modify it into another smooth mesh ***U***;
4) Compute the s_i^U and F_i^U for each vertex i on mesh ***U***. The magnitude of the Laplacian coordinate δ_i on the final deformed mesh is computed by multiplying rm_i with $\sqrt{s_i^U}$, the direction of the δ_i is computed by multiplying rd_i with F_i^U;
5) The final deformed mesh ***G'*** is reconstructed with the inverse Laplacian transform L^{-1}. That is,

$$\boldsymbol{G'} = L^{-1}((rm_i * \sqrt{s_i^U}) * (rd_i * F_i^U)).$$

Because our objective is to simulate the results of sculpting deformation, hence, we use OPCODE version 1.3[15] to detect the deformed region of source mesh and the corresponding region on tool mesh, and use the mean-value coordinate parameterization method[16] to mapping between the two region. Figure. 2 and Figure. 3 are another two examples of the detail-preserving sculpting deformation.

Acknowledgment

This work is partially supported by the Research Grants Council of the Hong Kong Special Administration Region (Project No. 412106), a Direct Grant from the Chinese University of Hong Kong (Project No. 2050413), the National Basic Research Program of China(Project No. 2004CB719403; No. 2004CB318000), National Natural Science Foundation of China(Project No. 60573151, 60773179) and the National High Technology Research and Development Program of China (Projected No. 2004AA115120).

References

[1] A. Angelidis, G. Wyvill and M. Cani. *Sweepers: swept user-defined tools for modeling by deformation*. In Proceedings of Shape Modeling and Applications, 2004, pp. 63-73.

[2] C. Li, W. Ge and G. Wang. *Dynamic surface deformation and modeling using rubber sweepers*. Edutainment 2006, LNCS 3942, 2006. pp. 951-961,

[3] Y. Yu, K. Zhou, D. Xu, X. Shi, H. Bao, B. Guo and H.-Y. Shum. *Mesh editing with Pois-son-based gradient field manipulation*. In: Computer Graphics Proceedings, ACM SIGGRAPH, 2004, pp. 641-648.

[4] A. Nealen, O. Sorkine, M. Alexa and D. Cohen-Or. *A sketch-based interface for detail- preserving mesh editing*. In: Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH, 2005, pp. 1142-1147.

[5] M. Alexa. *Differential coordinates for local mesh morphing and deformation*. The Visual Computer, Vol. 19(2), 2003, pp. 105-114.

[6] Y. Lipman, O. Sorkine, D. Cohen-Or, D. Levin, C. R?ssl and H.-P. Seidel. *Differential coordinates for interactive mesh editing*. In: Proceedings of Shape Modeling International, IEEE Computer Society Press, 2004, pp. 181-190.

[7] O. Sorkine, Y. Lipman, D. Cohen-Or, M. Alexa, C. R?ssl and H.-P. Seidel. *Laplacian surface editing*. In Symposium on Geometry Processing, ACM SIGGRAPH/Eurographics, 2004, pp. 179-188.

[8] K. Zhou, J. Huang, J. Snyder, X. Liu, H. Bao, B. Guo and H.-Y. Shum. *Large Mesh Deformation Using the Volumetric Graph Laplacian*. In: Computer Graphics Proceedings, ACM SIGGRAPH, 2005, pp. 496-503.

[9] Y. Lipman, O. Sorkine, D. Levin and D. Cohen-Or. *Linear Rotation-Invariant Coordinates for Meshes*. In: Computer Graphics Proceedings, ACM SIGGRAPH, 2005, pp. 479-487.

[10] D. Zorin, P. Schroder and W. Sweldens. *Interactive multiresolution mesh editing*. In: Com-puter Graphics Proceedings, ACM SIGGRAPH, 1997, pp. 259-268.

Figure 2. Bunny model hit by a fist. Left shows the undeformed bunny model. Middle shows the bunny model hit by a fist. Right shows the deformed bunny model. Note the detail of the bunny'skin is preserved in the deformation

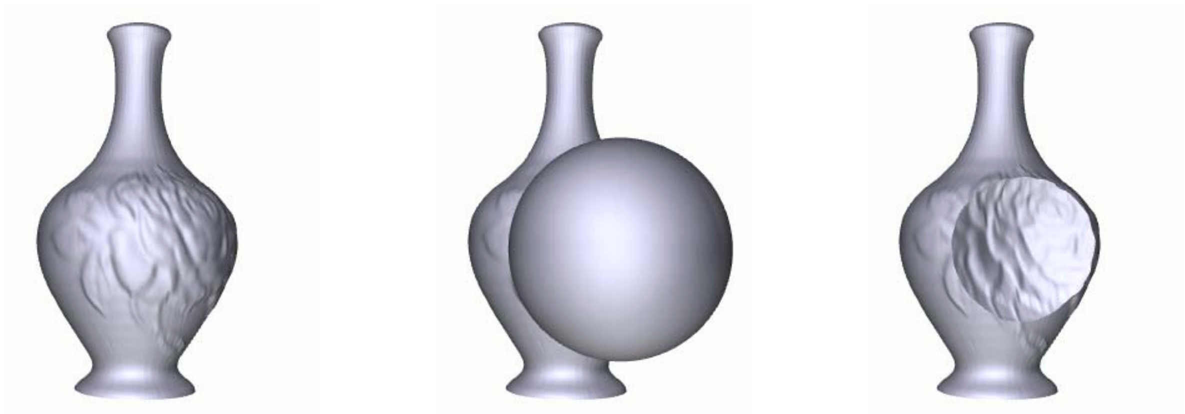

Figure 3. Vase model hit by a ball. Left shows the undeformed vase model. Middle shows the vase model hit by a ball. Right shows the deformed vase model. Note the detail of the carving on the vase is preserved in the deformation

[11] L. Kobbelt, S. Campagna, J. Vorsatz and H.-P. Seidei. *Interactive multi-resolution modeling on arbitrary meshes*. In: Computer Graphics Proceedings, ACM SIGGRAPH, 1998, pp. 105-114.

[12] I. Guskov, W. Sweldens and P. Schröder. *Multiresolution signal processing for meshes*. In: Computer Graphics Proceedings, ACM SIGGRAPH, 1999, pp. 325-334.

[13] I. M. Boier-Martin, R. Ronfard and F. Bernardini. *Detail-preserving variational surface design with multiresolution constraints*. In: Proceedings of Shape Modeling International, 2004, pp. 119-128.

[14] M. Desbrun, M. Meyer, P. Schroder and A. H. Barr. *Implicit fairing of irregular meshes using diffusion and curvature flow*. In: Proceedings of ACM SIGGRAPH, 1999, pp. 317-324.

[15] http://www.codercorner.com/Opcode.htm

[16] M. Floater. *Mean value coordinates*. Computer aided geometric design. Vol. 20(1), pp. 19-27.