Mesh Editing in ROI with Dual Laplacian

Luo Qiong, Liu Bo, Ma Zhan-guo, Zhang Hong-bin
College of Computer Science, Beijing University of Technology, China
lqjingjing@sohu.com, liubo@bjut.edu.cn,mzgsy@163.com,zhb@public.bta.net.cn

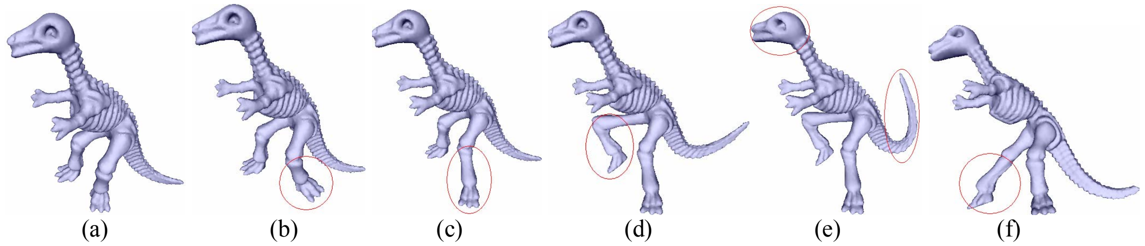

Figure 1. Deform the dinosaur model: (a) the original meshes; (b) (c) (d) (e) (f) are the deformed results and the ROIs are labeled with red circles.

Abstract

In this paper, we propose a new mesh editing method worked in the dual domain for ROI (region of interest). This method has intuitive user interface and efficient reconstruction process. The mesh detail is represented by dual Laplacian, and the reconstructed mesh is obtained by solving a quadratic minimization problem which preserves the mesh detail in least square sense. Preconditioned bi-conjugate gradient method (PBCG) is adopted to iteratively update the positions of the dual vertices and the dual Laplacian coordinates. The experiments demonstrate that our method can work effectively and give visually pleasing results.

Keywords--- mesh deformation, ROI, interactive interface, dual Laplacian.

1. Introduction

Mesh editing is an active research field in computer graphics and has received much attention recently. It is useful in a variety of applications in computer modeling and animation. Mesh editing often requires that the global shape deform in a user specified way while geometric details are well preserved.

In the past years, many mesh editing techniques were developed. Free-Form Deformation (FFD) [1] embeds the original model in the control lattices and indirectly controls the deformation of model by the control lattices. Multi-resolution technique [2, 3] encodes the mesh geometry as a coarsest mesh and several levels of refinement. The modified shape is computed by adding finer level details to the deformed coarsest mesh. Sheffer [4] represents the mesh by rotation-invariant pyramid coordinates. Kho [5] gives a method to implicitly control the deformation in the region of interest (ROI) with a reference curve that users sketch. Shi [6] proposes a fast multigrid algorithm for deforming meshes.

Laplacian techniques [7-8] cast mesh deformation as minimizing an energy function which contains terms for both detail preservation and position constraints. The Laplacian coordinate describes the mesh geometry detail which is expressed as the difference between a vertex and its one-ring neighbor vertices. Since the Laplacian coordinate is encoded in a global coordinate system, it faces the transformation problem: the Laplacian coordinates need to be appropriately transformed to fit the orientation of details in the deformed mesh. Kin-Chung [9] gives dual Laplacian deformation algorithm to edit whole model, which can avoid transformation problem to some extent. Our method is based on the dual Laplacian representation.

Generally speaking, deformation is usually carried out in some parts of a model, while the other parts (a majority of the whole model) are unchanged. If taking the whole model into consideration, the editing operation may be a complex process, especially for large mesh models. Therefore, it is necessary to develop effective editing methods for the deformation of ROI.

Computer Graphics, Imaging and Visualisation (CGIV 2007)
0-7695-2928-3/07 $25.00 © 2007 IEEE

We propose a new mesh editing framework with an intuitive interface and efficient reconstruction algorithm. Deformation is confined only in ROI which is determined by our Water-Filling technique. We represent the mesh surface information with dual Laplacian coordinates, and reconstruct the deformed mesh by solving a quadratic minimization problem. The resulted linear system is solved using the preconditioned bi-conjugate gradient method. Observing that direct interaction in 3D space is often a laborious task especially for non-expert users, we propose an intuitive user interface easy to manipulate. Experiments show that our system can give natural deformation with pleasing visual effect.

In this paper, we make the following contributions:

- A new technique named Water-Filling that determines the ROI effectively with easy manipulation and intuitive interface
- An interactive ROI editing mechanism based on the dual Laplacian representation, which satisfies the preservation of local geometry and position constraints

The rest of the paper is organized as follows. In Section 2, we introduce the dual Laplacian representation. Section 3 presents our editing framework and explains implementation details. In section 4, we show some experiment results, and conclude in Section 5.

2. Dual Laplacian

Our method is based on the dual Laplacian representation. Dual Laplacian follows the same principle as Laplacian, but presents the Laplacian coordinates in dual domain. There are many ways to build dual mesh of the primal mesh. In our method, we choose barycenter dual mesh and take triangular mesh for the input mesh, the dual vertices can be found by getting the centroid of the triangular faces in primal mesh. It is obvious that there is a one-to-one mapping between the triangular faces in primal mesh and the dual vertex in dual mesh.

As in [10], assume that triangle plane i in mesh consists of three vertices vij (j=1, 2, 3), then the dual vertex included by the triangular plane i is:

$$\tilde{\mathbf{v}}_{\mathrm{i}} = \frac{\mathbf{1}}{\mathbf{3}}\sum_{\mathrm{j=1}}^{\mathbf{3}} \mathbf{v}_{\mathrm{ij}} \tag{1}$$

in matrix form,

$$\tilde{\mathbf{V}}_{\mathbf{I}} = \mathbf{D}\mathbf{V}_{\mathbf{IJ}} \tag{2}$$

where $\mathbf{D}$ is the vertex-face incident matrix, with only the diagonal elements nonzero. Fig. 2 shows the fixed one-ring structure of the dual vertex $\tilde{\mathbf{v}}_i$. The unique decomposition of dual vertex is given by Equation (3) [9]:

$$\tilde{\mathbf{v}}_i = \tilde{\mathbf{q}}_i + h_i\tilde{\mathbf{n}}_i = \sum_{j\in(1,2,3)} \tilde{w}_{i,j}\tilde{\mathbf{v}}_{i,j} + h_i\tilde{\mathbf{n}}_i \tag{3}$$

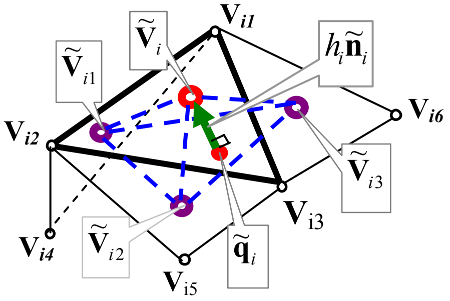

Figure 2. Illustration of the one-ring structure of dual vertices

where $\tilde{\mathbf{q}}_{\mathrm{i}}$ is the projection of the dual vertex $\tilde{\mathbf{v}}_{\mathrm{i}}$ in the plane i, and $\tilde{\mathbf{n}}_{\mathrm{i}}$ is the outward unit normal, the normalized weights $\tilde{w}_{i,j}$ are the barycentric coordinates of $\tilde{\mathbf{q}}_{\mathrm{i}}$ corresponding to $\Delta\tilde{\mathbf{v}}_{i,1}\tilde{\mathbf{v}}_{i,2}\tilde{\mathbf{v}}_{i,3}$, and $\sum_{j=1,2,3}\tilde{w}_{i,j} = 1$. h_i is the signed magnitude of the normal component. Let $\tilde{\mathbf{l}}_i$ denotes the Laplace operator of $\tilde{\mathbf{v}}_i$, Rearrange the expression (3):

$$\begin{aligned}\tilde{\mathbf{l}}_i &= -h_i\tilde{\mathbf{n}}_i = \tilde{\mathbf{q}}_i - \tilde{\mathbf{v}}_i \\ &= \sum \tilde{w}_{i,j}\cdot\tilde{\mathbf{v}}_{i,j} - \tilde{\mathbf{v}}_i \\ &= \sum_{j\in(1,2,3)} \tilde{w}_{i,j}(\tilde{\mathbf{v}}_{i,j} - \tilde{\mathbf{v}}_i)\end{aligned} \tag{4}$$

in matrix form,

$$\tilde{\mathbf{l}} = \tilde{\mathbf{L}}\tilde{\mathbf{V}} = \tilde{\mathbf{L}}\mathbf{D}\mathbf{V} = \mathbf{A}\mathbf{V} \tag{5}$$

$$\phi(\mathbf{v}) = \mathbf{A}\mathbf{V} \tag{6}$$

where $\mathbf{A} = \tilde{\mathbf{L}}\mathbf{D}$ and $\phi(\mathbf{v})$ denotes the dual Laplacian coordinates of the primal vertices $\mathbf{v}$. Since $\mathbf{D}$ is of full rank matrix and $\mathbf{L}$ has rank $n_d - 1$, given the original dual Laplacian coordinates $\tilde{\mathbf{l}}$ of the ROI mesh and one fixed primal vertex position, we can get the rest of the vertex positions by solving the equation (5) without error.

The rationale of using the dual Laplacian to avoid the transformation problem and preserve geometry property is to decompose the global geometry into local parameterization information and local geometry information which are two sets of scalar data. The local parameterization information is defined by the coefficients of the Laplace operator, and the local geometry information is captured by the magnitudes of the Laplacian coordinates. Both the Laplacian coefficients and the magnitudes of the Laplacian coordinates are kept unchanged as possible before and after editing [9].

3. Mesh editing framework in ROI

The basic idea of our framework for dual Laplacian mesh editing in ROI is to describe the mesh surface detail information by dual Laplacian coordinates and reconstruct the deformed mesh by preserving them in the least square sense.

For a ROI mesh editing system, there are two key problems that must be solved carefully: 1. how to determine the ROI; and 2. how to deal with the boundary vertices between the ROI and the fixed component. Our solutions are described in detail as follows.

3.1. Water-filling technique to determine ROI

Our system supplies an intuitive way for users to determine the ROI. The user begins editing by directly clicking two points on the screen, the first point and the last point, which are then projected onto the corresponding vertices in the surface of 3D model, obtaining the first vertex and the last vertex. The system implicitly connects the two vertices with a line which we name it as region line, and computes a cutting plane perpendicular the region line at the last vertex. The intersection of the plane with the surface defines a boundary of the ROI. Then search the neighbors starting from the first vertex and extend the search operation to the neighbors, only when meeting the boundary the search process end. Label all the vertices of been searched and determine the region of interest. This selecting method is similar as the water filling phenomenon, so we called it Water-Filling technique. This method works efficiently only for the region that involves a closed mesh at the side of first vertex and our system only considers ROI which satisfies this requirement. In general, the regions that users usually want to deform mostly satisfy this condition, like the leg, arm, tail or head. So Water-Filling technique is a good choice for ROI selecting in most cases. If the region been selected is not satisfied, users can cancel it and select other regions. The interactive interface is flexible for users. After determining the ROI, The following step is to select the deform-controlling handles in the ROI and determine their new positions. Handles are usually a subset of vertices in ROI.

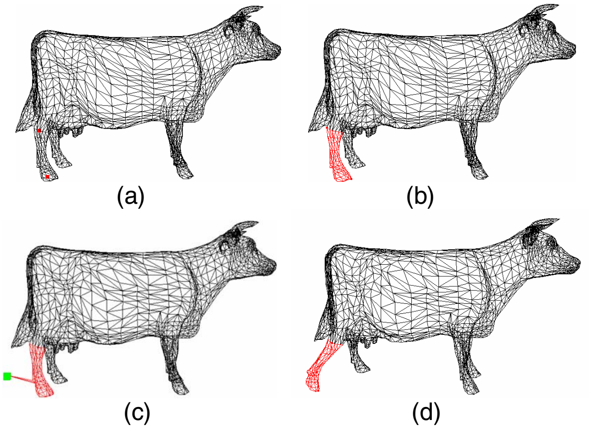

Figure 3. The cow model: (a) the original mesh, the red part on right back leg is the ROI, and the green point is the selected handle; (b) the green point is the new position of the handle that the user chooses by himself; (c) the deformed mesh that controlled by the handle, only the vertices in ROI are considered and other vertices out of the ROI are kept fixed; (d) the deformed model.

We have partitioned the model into two parts now: 1, the fixed part (black part); 2, the ROI (red part) which includes the deformation control handles (green points). Fig.3 illustrates this partition. The fixed part will be kept unchanged in the deformation. The green handles determine what the resulted model will be, and the red part should be moved following the green handles in a natural way and the new position of vertices in red part will be obtained in the editing process. The boundary vertices are regarded as handles.

3.2. Mesh deformation in ROI

Suppose **G (K, V)** denotes triangular mesh in ROI, where K denotes the connectivity and V denotes the geometry positions $\mathbf{V} = \{\mathbf{v}_i | 1 \le i < n\}$, the handles are a subset of the vertices in ROI $\mathbf{U} = \{\mathbf{u}_i | m \le i < n\}$.

It has been observed that the solution behaves better if the constraints coming from the handles are satisfied in the least squares sense rather than exactly [7]. We will process all handles in this way.

According to the dual Laplacian representation which described in part 2, we use the approach of energy minimization [7] to reconstruct the resulting mesh, the new positions **V'** are obtained by minimizing the following quadratic error function:

$$E(\mathbf{V}') = \sum_{i=1}^{n} \left\| \phi(\mathbf{v}_i{}') - \tilde{\mathbf{l}}_i \right\|^2 + \sum_{i=m}^{n} \left\| \mathbf{v}_i{}' - \mathbf{u}_i \right\|^2 \tag{7}$$

This quadratic error function corresponds to an over-determined linear sparse system of the form $\tilde{\mathbf{A}}\mathbf{V}' = b$, where $\tilde{\mathbf{A}}$ is the augmented system matrix which defined by the primal mesh and position constraints, and it keeps fixed throughout the whole process. Minimization of this error function is equal to solve the augmented linear system in the least square sense, leading to the normal equations $\tilde{\mathbf{A}}^T\tilde{\mathbf{A}}\mathbf{V}' = \tilde{\mathbf{A}}^T b$

in this heading, they should be Times 11-point boldface, initially capitalized, flush left, with one blank line before, and one after.

3.3. PBCG iteration

Noting that both the local parameterization information and local geometry information are nonlinearly dependent on the vertex position in primal mesh, iterative solver for the normal equations may be a good choice to minimize the distortion and get satisfactory results. We adopt the preconditioned bi-conjugate gradient method to solve the normal equations iteratively.

Since the matrix is a sparse matrix, it is inefficient to use general methods of linear algebra on such problem, because most of the arithmetic operations devoted to solving the set of the equations or inverting the matrix involve zero operands. The PBCG method will improve the efficiency of solving a sparse linear system.

We adopt the two steps introduced in [9] to update the positions of the dual vertices and the dual Laplacian coordinates:

Step 1. Update the dual vertex positions:

$$\tilde{\mathbf{A}}^T\tilde{\mathbf{A}}\mathbf{V}^{t+1} = \tilde{\mathbf{A}}^T\tilde{b}^t \tag{8}$$

$$\tilde{\mathbf{V}}^{t+1} = \mathbf{D}\mathbf{V}^{t+1} \tag{9}$$

where $\tilde{b}^t$ is derived from $\tilde{\mathbf{l}}^t$ and the position of current handles, $\mathbf{V}^0$ is the positions of the primal vertex and $\tilde{\mathbf{l}}^0$ is the dual Laplacian coordinates of the primal mesh.

Step 2. Update the dual Laplacian coordinates:

$$\begin{aligned} \tilde{\mathbf{l}}_i^{t+1} &= -h_i^{t+1}\tilde{\mathbf{n}}_i^{t+1} \\ &= \sum_{j\in(1,2,3)} \tilde{w}_{i,j}(\tilde{\mathbf{v}}_{i,j}{}^{t+1} - \tilde{\mathbf{v}}_i{}^{t+1}) \end{aligned} \tag{10}$$

$\tilde{\mathbf{n}}_i{}^{\mathrm{t+1}}$ changes along with the updated dual vertex positions $\tilde{\mathbf{v}}_{i,1}{}^{\mathrm{t+1}}, \tilde{\mathbf{v}}_{i,2}{}^{\mathrm{t+1}}, \tilde{\mathbf{v}}_{i,3}{}^{\mathrm{t+1}}$, and the dual Laplacian coordinate $\tilde{\mathbf{l}}_i^{t+1}$ also changes accordingly.

The iteration process will terminate as long as the updating rate of primal vertex is below a threshold.

4. Experimental results

We have experimented with local mesh deformations in ROI on some models [11] (dinosaur, cow, camel, pithecanthrope and triceratops). Figures 4-7 show some editing results.

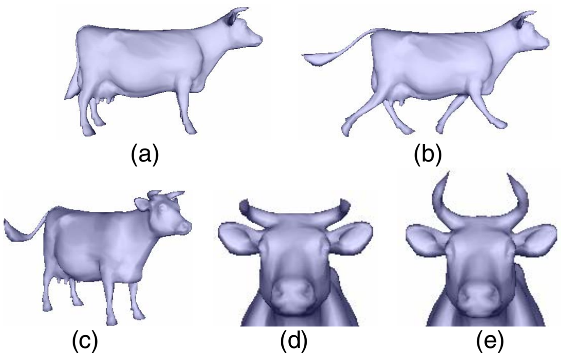

Figure 4. The cow model: (a) (d) are the original model; (b) the running cow with four legs and tail ROIs, we gain this model with five operations;(c) the shaking-tail cow with the tail ROI; (e) deform the horns.

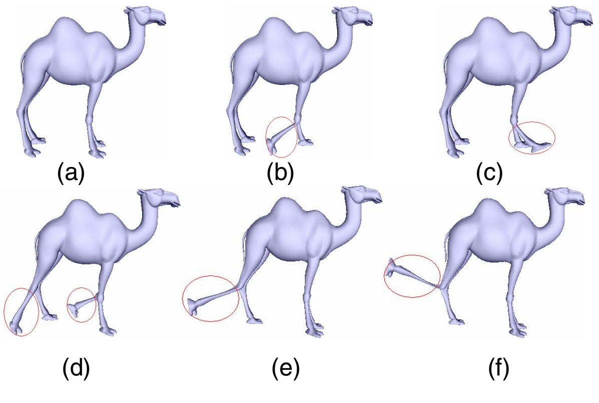

Figure 5. The camel model: (a) the original mesh; (b) (c) (d) (f) are the deformed camel model, the ROIs are labeled with red circles.

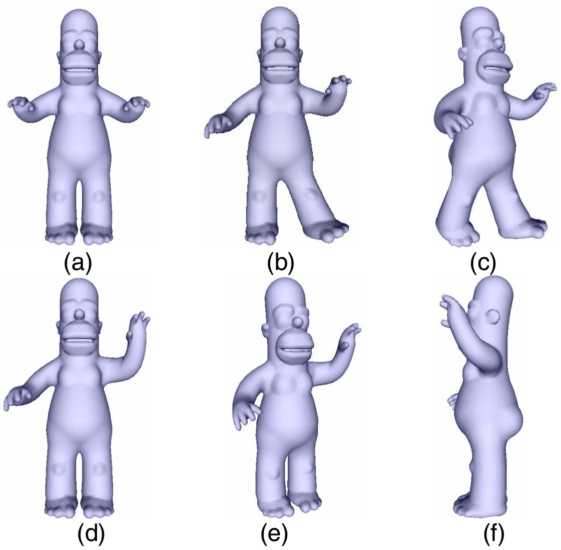

Figure 6. The pithecanthrope model: (a) the original mesh; (b) the left leg steps out; (d) left arm lifts up and right arm lifts down; (c) (e) (f) are the side view.

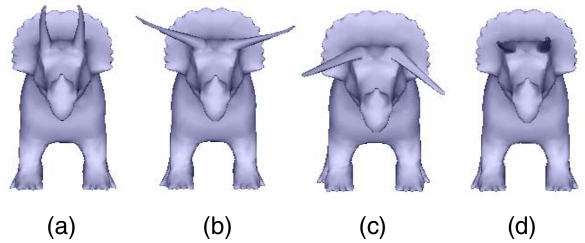

Figure 7. The triceratops model, select the two horns as ROI: (a) the original mesh; (b) (c) the horns rotate with different angles; (d) the horns rotate in perpendicular direction.

Experiments demonstrate that our system can obtain natural deformation effect in ROI, satisfying not only precise handle position constraints but also preserving local mesh details well.

5. Conclusion

We have presented an efficient method to edit surface mesh in ROI by the dual Laplacian representation. The mesh is reconstructed by minimizing a quadratic error function which involves both the preservation of local geometry features and handle position constraints. We use preconditioned bi-conjugate gradient method to iteratively update the dual vertices position and dual Laplacian coordinates to get the deformed mesh. The experiment results demonstrate that our editing system for mesh ROI can generate pleasing visual effect.

6. Acknowledgements

This work was supported by National Natural Science Foundation of China (No. 60443005, 60575008) and China Ministry of Education (No.20040005016).

7. References

[1] Sederber, T. W., Parry, S. R. Free-form deformation of solid geometric models. In Proceedings of the 13th annual conference on computer graphics and interactive techniques, ACM Press, 1986, 151-160..

[2] Zorin, D., Schroder, P., Sweldens, W. Interactive multiresolution mesh editing. In Proceedings of ACM SIGGRAPH 97, ACM Press, 1997, 259-268.

[3] Guskov, I., Sweldens, W., Schroder, P. Mutiresolution signal processing for meshes. In Proceedings of ACM SIGGRAPH 99, ACM Press, 1999, 325-334.

[4] Sheffer, A., Kraevov, V. Pyramid coordinates for morphing and deformation. In Proceedings of 3DPVT'2004. 68-75.

[5] Kho, Y., Garland, M. Sketching Mesh Deformations. In Proceedings of ACM SIGGRAPH 2005. ACM Press, 2005, 147-154.

[6] Shi, L., Yu, Y., Bell, N., Feng, W. A Fast Multigrid Algorithm for Mesh Deformation. In Proceedings of ACM SIGGRAPH 2006. ACM Press, 2006, 1126-1134.

[7] Sorking, O., Lipman, Y., Cohen-OR, D., Alexa, M., Rossl, C., Seidel, H.-P. Laplcian surface editing. In Processings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, Eurographics Association, ACM Press, 2004, 179-188.

[8] Nealen, A., Sorking, O., Alexa, M., Cohen-Or, D. A Sketch-Based Interface for Detail-Preserving Mesh Editing. In Proceedings of ACM SIGGRAPH2005. ACM Press, 2005, 1142-1147.

[9] Kin-Chung, O., Chiew-Lan, T., Ligang, Liu., Hongbo, F., Dual Laplacain Editing for Meshes. IEEE Transactions on Visualization and Computer Graphics. 2006, 386-395.

[10] Taubin, G. Dual MesH Resampling. In Processings of Pacific Graphics 2001. 94-113.

[11] http://shapes.aim-at-shape.net/index.php.