

3D Model Editing from Contour Drawings on Orthographic Projection Views

Yuhui Hu^{1,3,4}, Xuliang Guo^{1,3,4}, Baoquan Zhao^{1,3,4},
Shujin Lin^{2,3,*}, and Xiaonan Luo^{1,3}

¹ School of Information Science & Technology, Sun Yat-sen University,
Guangzhou 510006, China

² School of Communication and Design, Sun Yat-sen University

³ National Engineering Research Center of Digital Life

⁴ 4 State-Province Joint Laboratory of Digital Home Interactive Applications

Abstract. 3D modeling and mesh editing for objects from orthographic projections of engineering drawing has been very common and convenient. People has been accustomed to using such simple and fast sketching way to express their visualized thought. In this paper we present a method for model designers by sketching on the orthographic projection to control the deformation of surface meshes. Our system lets the user easily choose the region of interest for constraining the deformation area, and then sketch the desired contour. For a given new sketch on the orthographic projection, the system automatically deform the 3D model by using Laplacian deformation, which makes the contours of 3D model fit the contour line of the users sketching. Various examples have validated the effectiveness of our proposed method, which can be regarded as a effective method in 3D model editing.

Keywords: Orthographic projection, Skech contour, Laplacian deformation, 3D model editing.

1 Introduction

Orthographic projection of engineering drawing is commonly seen in 3D objects reconstruction. In most of the CAD tools, traditional engineering objects are represented through three orthographic view: front, top and side views. A few contour lines suffice to sketch the main features of a shape. This is why model designs still prefer using sketching curves to create models, and explains the great success of sketch-based shape modeling and editing approaches. Given that, a method based on sketching the orthographic projection of 3D model has been proposed?in?the?thesis to get a new deformed model.

Many outstanding researches are dedicated to methods of model deformation by 2D sketches. In the work of Jie-Hui Gong et al. from reference[1], a new hybrid wireframe consisting of geometry, topology, vertexes and edges is proposed. Their system requires three views with good drawing for the handling of different

* Corresponding author.

objects such as lines, circles and other higher order curves. Higher order curves are approximated in polyline. But this is not suitable for the arbitrary sketches drawn by the user. In the work of Alec Rivers et al. from reference[2], the user specifies the silhouettes of a part from front, top or side views, then a 3D shape is automatically constructed in their system. Their system also just need solely 2D input and interaction to complete the complicated modeling work. They successfully reduce the required interface complexity for the user. In the work of Andrew Nealean et al. from reference[3], they successfully bridge the gap for interactively refining the design or re-use existing design by using curves as an interface for designing a surface. In their system, they also allow the user to define control curves by drawing them onto the shape in its current design stage, and then use the curves as handles for deformation right after their definition.

In this paper, we propose a sketching the contour of orthographic projection method combined with 3D mesh deformation, as shown in Figure 1. First of all, users choose the ROI (region of interest) for constraining the contour editing area and 3D model deformation area from the orthogonal view (front, top or side view). Then a new expected contour drawn by users in a 2D manner is needed before the system finds out the correspondence between the new sketch and the 3D model automatically. By doing so, users can change a given model into the shape they expect by Laplacian deformation.

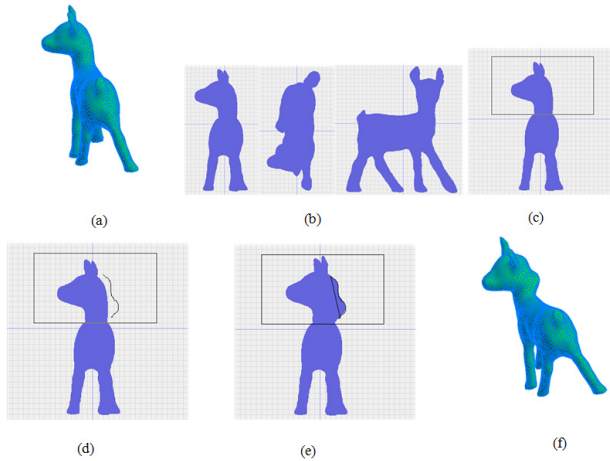


Fig. 1. (a) A template 3D model for a *Cervus nippon*; (b) The front, top and side views of the model; (c) ROI in front view chosen by user (d) A new drawing contour line sketched by the user; (e) The new orthographic projection after deformation; (f) Final model after the deformation

The organization of the rest of this paper is as follows. In the next section, the interface for the user to draw sketch is introduced. ROI (region of interest) in orthographic projection is chosen and uniform quadratic B-spline is applied to fit

the 2D strokes onto the orthographic projection (front, top or side view). In the third section, the correspondences between the new sketch and the 3D model mesh are determined. In the fourth section, Laplacian deformation method is applied to the 3D model deformation. The future work is concluded in the final section.

2 Sketching Two-dimensional Digitalized Contours

Our goal is to deform the 3D model to the shape we expected by changing its original contour, so the first thing we must do is to get the models orthographic projection(front, top and side views). For each vertex (x,y,z) on the 3D mesh, when we just focus on their coordinate in the xy -plane, we will get the front view of 3D model. The same applies to the top view in xz -plane and the side view in yz -plane. In this way, we can see the 3D model from different orthogonal view (front, top and side views), as shown in Figure 2 (b)-(d).

In order to constrain the contour editing area and 3D model deformation area, the user should choose a rectangular area as the ROI. By ROI we will focus on the interest area we want to deform, as shown in Figure 1(c). Once we have chosen the ROI, we can draw the new contour lines with free curve or a uniform quadratic B-spline. While using the B-spline to draw the new contour, we can move the B-spline control point to ensure smoothness and locality of the contour, because a quadratic B-spline is continuous, and each point on the B-spline is affected by three control points. The new contour drawing as a free cure is shown in Figure 1(d).

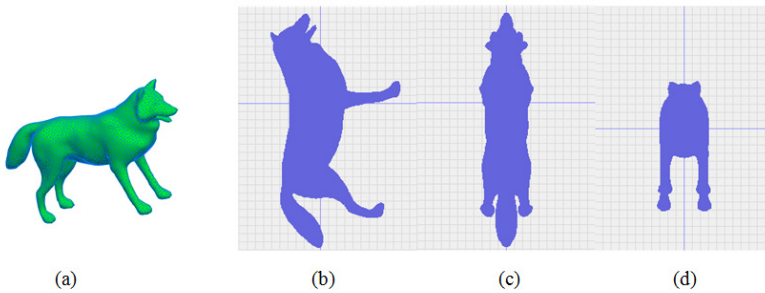


Fig. 2. (a) is a template 3D model, (b)-(d)the front, top and side views respectively

3 Determining 2D Sketches to 3D Model Mesh Correspondences

To deform the 3D model from the new contour drawn in Orthographic projection view by the user, we need to find out the correspondences between the new

contour lines and the 3D model. Once we obtain the correspondence, the Laplacian deformation can be preformed accordingly. The new contour lines drawn by the user was stored as a set of dense points, and the number of the vertexes on the 3D model is much more less. That is to say, most of points on the new contour are redundant, so we can not find out the correspondence between every points on the new contour and vertexes on the mesh. Thus we must sample these points to represent the new contour drawn by the user. From the start point of the users new contour, we sample a new point as its another feature points at a certain distance threshold d , the distance D_{ij} between feature points and the new contour points is calculated by

$$D_{ij} = \|S_{i-1} - C_j\| \quad (1)$$

Where $i(i \in [1, n])$ is the sequence number of feature points we sampled, $j(j \in [1, m])$ is the sequence number of new contour points. S is the set of sampling feature points and C is the set of new contour points draw by the user. If $D_{ij} \geq d$, we determine $S_i = C_j$, namely the contour point can be regarded as an another feature points.

According to the coordinates which 3D model projected on 2D Orthographic view, on the basis of sampling feature points we have found, its easy to find out the corresponding vertex V_i on 3D model, which has the nearest distance from S_i . Then the set of corresponding vertex V_i can be used as the additional constraint vertex.

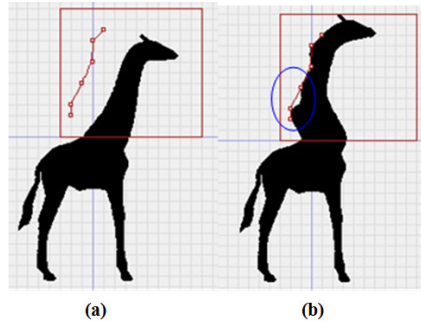


Fig. 3. (a) A new target contour with a big threshold distance d ; (b) Result after Laplacion deformation

Its very important to assign the certain distance threshold d , because there is no certain standard for the meshes intensity. For the elaborative 3D model, if we set the distance threshold too much bigger, that means the sampling process will add a feature point in a large distance, then the number of feature points will be small. In order to keep the shape of original model during the deformation as much as possible, if the threshold is large, then the grid points on the model in the range area of threshold will not fit completely to the target contour. As shown in Figure 3(a), all the hollow dot on the new drawing target contour is set

as the feature points , and 3(b) shows the deforming result with a big threshold distance. Thus, choosing an appropriate threshold directly affects the deforming result.

Generally speaking, the density of local grid on a model is relatively uniform. The distance between two adjacent vertexes of the mesh, to some extent, reflects the density of local mesh. The first feature point we sampled S_0 is also the first contour point drawn by the user C_0 . We first find out the vertex V_0 on the mesh corresponding to S_0 , then find all the neighbors of V_0 to calculate the threshold distance d , and set d as following form:

$$d = \frac{1}{|N_i|} \sum_{j \in N(i)} \|V_0 - V_j\| \quad (2)$$

Where N_i is the set of all neighbors of V_0 , $|N_i|$ is the number of immediate neighbors of V_0 . The detail steps for the deformation are shown as the following:

- Step 1. The User draw a target contour L.
- Step 2. Get the first feature point on target contour L.
- Step 3. Find out the vertex V_0 on the mesh corresponding to S_0 .
- Step 4. Calculate the threshold distance d .
- Step 5. Sampling progress to get the set of feature points.
- Step 6. Find out the set of constrained vertexes V on model corresponding to S .
- Step 7. Calculate all the Cartesian coordinates for the set of constrained vertex V .
- Step 8. Add points V as mobile constraint points to Laplacian matrix L.
- Step 9. Laplacian deformation on the original mesh, and then update its Orthographic projection view.
- Step 10. Output the mesh model deformed.

The more detail information about step 7 to step 9 is introduced in the next section.

4 Laplacian Deformation based on Correspondences

The basic idea of the meshes deformation is to satisfy a linear deforming constraint (exactly in the least squares sense), while preserving differential properties of the original geometry. Such techniques have been introduced in various fashions [4][5][6]. In our system, in order to derive the linear constraints, we import the Laplacian coordinate to make the original geometry to be preserved in the least squares sense. For a mesh model, we present it as $M=(V, E, F)$, consisting of the set of vertexes V , the set of edges E and the set of facets F . Each vertex $V_i \in V$ is represented by Cartesian coordinates, denotes by $V_i = (x_i, y_i, z_i)$. To get the Laplacian coordinate of V_i , we should compute the center of mass of the neighbors of V_i as the follow form:

$$\delta_i = (\delta_i^x, \delta_i^y, \delta_i^z) = v_i - \frac{1}{d_i} \sum_{j \in N(i)} v_j \quad (3)$$

Where $N(i)=\{j|(i,j)\in E\}$ and means the number of neighbors of vertex.

Once we get the Laplacian coordinate of every vertex on mesh ,we introduce matrix L to present the mapping relation about the vector of absolute Cartesian coordinate transform to the vector of Laplacian coordinate.

$$LV = W \quad (4)$$

Where $V = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \dots & \dots & \dots \\ x_n & y_n & z_n \end{bmatrix}$ means the vector of Cartesian coordinate, and $W = \begin{bmatrix} \delta_1^x & \delta_1^y & \delta_1^z \\ \delta_2^x & \delta_2^y & \delta_2^z \\ \dots & \dots & \dots \\ \delta_n^x & \delta_n^y & \delta_n^z \end{bmatrix}$ means the vector of Laplacian coordinate. Before we construct the matrix L , we bring two matrix A and D . Matrix A is the adjacency matrix of the mesh

$$A_{ij} = \begin{cases} 1 & (i,j) \in E \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

And matrix D is a diagonal matrix where $D_{ii} = d_i$. Then Laplacian matrix L can compute as the following form. The example is shown in Figure 4.

$$L = I - D^{-1}A \quad (6)$$

As we can see, the sum of every row of L is zero, due to the nonsingular property of matrix L , which implies that we can not easily use the following formula to compute the new Cartesian coordinate of vertexes after the deformation.

$$V' = L^{-1}W \quad (7)$$

In order to uniquely restore the global Cartesian coordinates, we usually add some control point into the equation $LV=W$. Therefore the equation will be changed as:

$$\left(\frac{L}{I}\right)V = \left(\frac{W}{C}\right) \quad (8)$$

The same as $\tilde{L}V = \tilde{W}$. Where means the vector of control point's Cartesian coordinate, and I means such control points' unit vector (the j -th element is 1, the others is 0) (ex. $I_3 = (0, 0, 1, 0, \dots, 0)$).

In our system, the set of Controlling points C is divided as two sets: moving constrained points and boundary constrained points. The the junction point of ROI and the 3D mesh is regarded as the boundary constrained points, where the coordinates will be keep unchanged after deformation. The new contour feature points S are regarded as the moving constrained points. The detail algorithm for Laplacian deformation in our system is following:

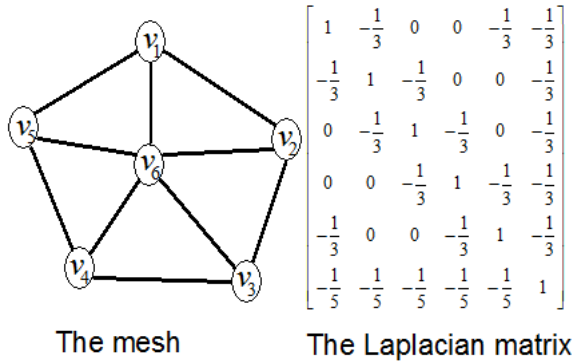


Fig. 4. A example of a triangular mesh and its associated Laplacian matrix

- Step 1. Input the original mesh M .
- Step 2. Compute the Laplacian matrix L for the vertexes in the ROI.
- Step 3. Initialize the matrix W according to form $LV = W$.
- Step 4. Initialize the set of Controlling points C : 1) Find out the junction point of ROI and the 3D mesh, and regard it as the boundary constrained points; 2) Compute the Cartesian coordinate of the new contour feature points S , and regard it as the moving constrained points.
- Step 5. Initialize the matrix \tilde{W} and the matrix \tilde{L} and by adding the set of Controlling points C and its corresponding unit vector I under the matrix W and matrix L
- Step 6. Take the least-squares solution to solve the formula $\tilde{L}V = \tilde{W}$,and then get the new Cartesian coordinate for all vertex in the ROI.
- Step 7. Update the position for all interest vertexes in ROI to get the new deformed mesh model.

5 Conclusion

In this paper, we propose a method by sketching the contour on orthographic projection views to control 3D mesh's deformations as we expected. User chooses a region of interest for deformation on front, top or side view, and then sketch the new arbitrary contour expected. After that, the system find out the correspondences between the new contour lines and the template 3D model automatically, and apply them to the Laplacian deformation. Laplacian coordinate are provided for the flexibility of deformation. Example shown in Figure 1 has proved that our method is an efficient approach in fast 3D model deformation.

From a user's point of view, many similar outstanding researches are dedicated to methods of contour drawing or sketching for modeling and deformation[7][8][9]. This is an strong evidence to show that contour drawing and sketching in our method to control the deformation is a valid approach. There are also a number of exciting avenues to explore in the future work. One direction is to extract more information from the user's new drawing contours, specially

relative depth and occlusion relationships form the drawing[10]. Developing techniques for deforming models more directly from sketching onto the 3D surface. Another future direction is to reduce the update time spent in the Laplacian deformation and updating process for the orthographic projection views.

Acknowledgments. This research is supported by the National Natural Science Foundation of China (61103162), the Project 985 of Innovation Base for Journalism & Communication in the All-media Era of Sun Yat-Sen University.

References

1. Gong, J.-H., et al.: Solid reconstruction using recognition of quadric surfaces from orthographic views. *Computer-Aided Design* 38(8), 821–835 (2006)
2. Rivers, A., Durand, F., Igarashi, T.: 3D modeling with silhouettes 29(4) (2010)
3. Nealen, A., Igarashi, T., Sorkine, O., et al.: FiberMesh: designing freeform surfaces with 3D curves. *ACM Transactions on Graphics (TOG)* 26(3), 41 (2007)
4. Sorkine, O.: Differential representations for mesh processing. *Computer Graphics Forum* 25(4), 789–807 (2006)
5. Alexa, M.: Differential coordinates for local mesh morphing and deformation. *The Visual Computer* 19(2), 105–114 (2003)
6. Lipman, Y., Sorkine, O., Cohen-Or, D., et al.: Differential coordinates for interactive mesh editing. In: *Proceedings of the IEEE Shape Modeling Applications*, pp. 181–190 (2004)
7. Kraevoy, V., Sheffer, A., van de Panne, M.: Modeling from contour drawings. In: *Proceedings of the 6th Eurographics Symposium on Sketch-Based interfaces and Modeling*, pp. 37–44. ACM (2009)
8. Nealen, A., Sorkine, O., Alexa, M., et al.: A sketch-based interface for detail-preserving mesh editing. In: *ACM SIGGRAPH 2007 Courses*, p. 42. ACM (2007)
9. Gingold, Y., Igarashi, T., Zorin, D.: Structured annotations for 2D-to-3D modeling. *ACM Transactions on Graphics (TOG)* 28(5), 148 (2009)
10. Karpenko, O.A., Hughes, J.F.: SmoothSketch: 3D free-form shapes from complex sketches. *ACM Transactions on Graphics (TOG)* 25(3), 589–598 (2006)