

# A Scribble-Based Interface for Mesh Segmentation

Fuyan Yang  
Linyi University  
LinYi, China  
yangfuyan1@163.com

*Abstract—This paper presents a method for three-dimensional mesh surface models segmentation that combines interactive part selection with region growth. First, some mesh vertices are selected as object and background seeds using scribble-based selection methods, and the others are unknown region. Then, the mesh model can be automatically cut into two parts using region growth. The result is directly affected by the vertices on the part boundaries. As a result, the improved-methods set the vertices that are both adjacent to object and background as special vertices, and use cutting results as object and background seeds to do the region growth again. The experiments show that the result is more accurate because most vertices of the object and background are selected as seeds.*

*Key words—mesh segmentation; scribble-based; boundaries; the region growth*

## I. INTRODUCTION

In recent years, 3D mesh models is one of the most methods to represent digital geometry model and has widely used in many fields, such as internet, entertainment and manufacture industries. How to reuse existing mesh models and how to modify an existing mesh model, has become an important research topic.[1,18] Mesh segmentation, like image segmentation, is the process of partitioning a mesh into more visually simple and meaningful components. It has recently become of interest and a key ingredient in many mesh manipulation algorithms. [2,19,22]

Mesh segmentation techniques can be classified into patch-type and part-type[20]. Patch-type segmentation is often used for texture mapping [3], building charts [4] and geometry-image creation [5]. Part-type segmentation divides a mesh into meaningful parts without restricting the part topology[6]. However, human perception is extremely complicated. Both the component decomposition and the concept of “meaningful” are highly subjective and hard to qualify, not to mention computing the meaningful decomposition automatically. We aim to cut a mesh into components that are consistent with user intention and meaningful. Generally, we cut a meaningful component from its underlying mesh by specifying which parts of the mesh belong to the “foreground”(the part we want to cut out) and the rest to the background.

This paper presents a method for three-dimensional mesh surface models segmentation that combines interactive part selection. First, some mesh vertices are selected as object and background seeds using scribble-based selection methods, and the others are unknown region. Then, the mesh model can be automatically cut into two parts using region growth. The

result is directly affected by the vertices on the part boundaries[21]. As a result, the improved-methods set the vertices that are both adjacent to object and background as special vertices, and use cutting results as object and background seeds to do the region growth again.

## II. RELATED WORK

The method of 3D mesh models segmentation can be divided into automatic segmentation method and sketch-based user interface. There are many automatic segmentation algorithm[7], such as curvedness-based region growing approach[8], hierarchical face clustering[9], iterative clustering[10]. In the most of automatic segmentation algorithms are split using a threshold as a control parameter, the result is not intuitive and hard to quality, it takes several attempts to obtain satisfactory results. Automatic segmentation method can automatically mesh model segmentation, but it is a significant computational workload, low efficiency and hard to get “meaningful” segmentation. In recent years, sketch-based interfaces have emerged as an approach to enhance user interactions for various design activities ranging from image segmentation to geometric design.

Semi-automatic mesh segmentation need to user choose. There are several ways to achieve semi-automatic mesh segmentation. The easiest way is to allow users to select vertices, triangular faces directly on the mesh model, We can get the desired part through numerous selection. Although this method is very intuitive and very easy to implement, this approach will fail when the model is large or the desired part is not currently visible; Other interactive method of segmentation is to allow users to select the mesh boundary directly. The most important is get the boundary in mesh segmentation. Once get the object boundary, we can get the required model part. We can select the boundary manually[11], also we can first select some point in the boundary of the target, then get boundary according to the shortest path[12,17]. However, this method requires drawing a boundary line or boundary point, in order to obtain the boundary line or boundary point, we needs to rotate the mesh model in the selection process. This has brought great difficulties, we can not get the required part of the model when the boundary is not visible. In recent years, sketch-based interfaces have emerged as an approach to enhance user interactions for various design activities ranging from image segmentation to geometric design. They are expected to provide interaction between computers and users that do not hinder creative thinking. Sketch-based interaction has been successfully used in image segmentation[13] and mesh

segmentation[14]. This mesh segmentation result is directly affected by the vertices on the part boundaries, the vertices on the part boundaries easily split error. We present the improved-methods set the vertices that are both adjacent to object and background as special vertices, and use cutting results as object and background seeds to do the region growth again. The experiments show that the result is more accurate because most vertices of the object and background are selected as seeds.

### III. ALGORITHMS

#### A. Definition

In computer graphics, a mesh is a representation of 3D models. Mesh  $TM$  is defined as a pair of  $TM=(T, V)$ , where  $T$  is a simplicial complex representing the connectivity of the vertices, edges, and faces, thus determines the topological type of the mesh;  $V=(v_1, \dots, v_n)$  describes the geometric position of vertices in  $R^3$

- Definition1: Let  $D(v)$  is the adjacent vertices of  $v_i$ , where  $D(v)$  and  $v_i$  are representing the connectivity of the edges.
- Definition2: Let  $T(v)$  is the related triangle of  $v_i$ , where  $T(v)$  and  $v_i$  are representing the connectivity of the faces.

#### B. Similarity definitions

Algorithms can generated boundary automatically, when we specify which parts of the mesh belong to the “foreground”(the part we want to cut out) and the rest to the background. This requires a measure of similarity between the vertices. This paper proposes a new distance between adjacent vertices. The distance between the vertices in the same area is smaller than the distance between the vertices of the different regions. In this paper, the distance between the vertices is defined by the unit normal vectors and the area.

In Marr's vision theory, human vision is more sensitive to edge information. The minima rule [15]states that human perception usually divides a surface into parts along the concave discontinuity of the tangent plane. In order to achieve this goal, we consider both the unit normal vectors and the area. The larger multiplication of the two unit normals vectors, the more likely the two vertices are in the same area. In the mesh model, the area of the triangle in the smooth area of a large area is larger than that of the edge. So we take the vertex area as a factor to consider the distance between vertices. In this paper, the distance between adjacent vertices of the triangular mesh model is calculated as follows:

$$d = -n_p \cdot n_q \cdot s_q$$

Where  $n_p$  and  $n_q$  are respectively the normals of vertices  $p$  and  $q$ , the normals of vertices can be computed by various methods, we adopt the method that proposed in the

paper[16],  $s_q$  is respectively mean of triangle area associated with the vertex  $q$ .

#### C. Scribble-Based Interface

As shown in Figure 1(a), the user draws freehand sketches on the mesh simply and quickly in our system. The freehand strokes roughly mark out the subpart of interest (as foreground) and background (green sketch for foreground and red sketch for background). The subpart of interest is then cut out from the mesh (shown in pink in in Figure1(a) ) in real time via an efficient region growing segmentation algorithm based on an improved distant(see Section 2.2). The cutting boundary is almost correct and is smoothed by regrouping the vertices on the boundaries.

To specify an interesting part on the mesh, the user marks a few freehand sketches on the image plane by dragging the mouse cursor while holding a button (left button indicating the foreground and right button for the background). The sketches in the image plane are then projected onto the surface of the mesh in 3D world space by computing the point of intersection of rays from the view point through the points on the sketch curve. These high level sketches specified by users need not be very precise. As shown in Figure 1(a), most marking strokes are actually far from the component boundaries. The freehand strokes roughly mark out the subpart of interest (as foreground) and background, the subpart of interest is then cut out from the mesh. This method not only avoids the over-segmentation in the automatic algorithm, but also avoids the direct selection of the mesh boundaries. We can obtain accurate part-type segmentation.

#### D. Mesh segmentation approach

It is critical that the mesh segment approach should be fast enough to generate the cutting boundary with very little delay, when users specify which parts of the mesh belong to the “foreground” and the rest to the background. It is straightforward to think of using graph cut to do the segmentation task inspired from the work of Lazy Snapping [16]. We adopt an efficient region growing segmentation algorithm[4]. The main factors influencing the segmentation results are the selection of seeds, the rule of growth and the conditions for the termination of growth. We use the improved distance measurement between two adjacent points on the mesh. The region growing algorithm starts with different seed vertices from marker sketches simultaneously and grows several submeshes according to the improved region growing segmentation algorithm. First, some mesh vertices are selected as object and background seeds using scribble-based selection methods, and the others are unknow region. Then, the mesh model can be automatically cut into two parts using region growth. The result is directly affected by the vertices on the part boundaries. As a result, the improved-methods set the vertices that are both adjacent to object and background as special vertices, and use cutting results as object and background seeds to do the region growth again. The experiments show that the result is more accurate because most vertices of the object and background are selected as seeds.

Every mesh vertex on (or near) the input sketches is labelled as "F" (foreground part) or "B" (background part) and all other vertices are labelled "U" (unknown). For each unknown vertex  $v$ , we find a triplet  $(v, m, d)$  where  $d$  is minimum distance between  $v$  and the nearest marked vertex  $v^* \in N$  and  $m$  is the marking label of  $v^*$ . A queue  $Q$  is defined as a set of triplets and is initialized as empty.

We summarize the steps of our approach as following:

- Step1. Label each vertex on foreground and background sketches as "F" and "B" respectively. Set  $Q = \Phi$ . Denote  $N$  the set of labelled "F" or "B".
- Step2. For each vertex in  $N$ , add the triplet  $(v, m, d)$  of its adjacent (by the distance defined in 2.2) vertex in  $N$ . Compared with the vertex in  $Q$ , if the vertex already exists in the  $Q$  and  $m$  value is different, the vertex  $v$  is set to a special vertex  $S$ , otherwise it will be added to  $Q$ .
- Step 3. Sort  $Q$  by value  $d$ .
- Step 4. Find minimum  $d_{min}$  of  $(v, m, d)$  in  $Q$ , label  $v$  as  $m$ , set  $Q = Q - \{(v_{min}, m_{min}, d_{min})\}$  and set  $N = N \cup \{v\}$ .
- Step 5. Repeat Step 2 to Step 4 until all the vertices are labelled as "F" or "B" except  $S$ .
- Step6. Set  $S$  and its adjacent vertex labelled "U"(unknown), repeat Step 1 to Step 5.

#### IV. EXPERIMENTAL RESULTS

The algorithm is implemented in Visual Studio.NET Microsoft 2010 environment. We use this algorithm to segment the nose and ears of the mesh (As shown in Figure 1 and Figure 2). As shown in Figure 1 and Figure 2, the user draws freehand sketches on the mesh simply and quickly in our system. The freehand strokes roughly mark out the subpart of interest (as foreground) and background (green sketch for foreground and red sketch for background). The subpart of interest is then cut out from the mesh (shown in pink in in Figure 1 and Figure 2) in real time via efficient region growing segmentation algorithm based on an improved distant (see Section 2.2). In Figure 1(a) and Figure 2(a), the cutting boundary is jaggy. In Figure 1(b) and Figure 2(b), the cutting boundary is smooth and the result is more "meaningful".

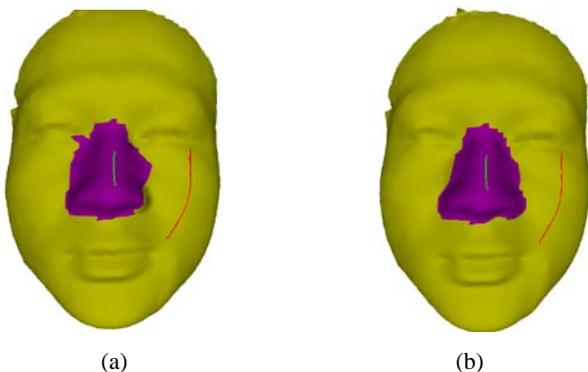


Figure1: (a) The cutting boundary is jaggy. (b) the modified boundary

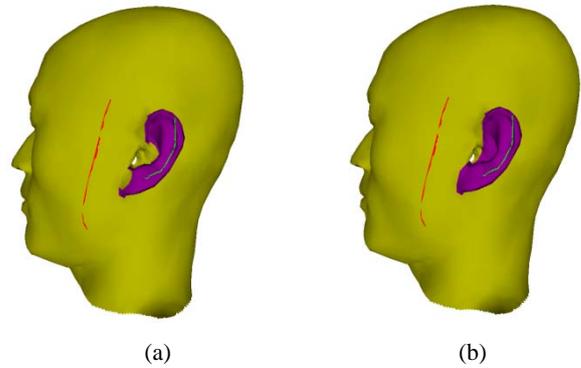


Figure2: (a) The cutting boundary is jaggy. (b) the modified boundary

#### V. CONCLUSIONS

This paper presents a method for three-dimensional mesh surface models segmentation by specifying which parts of the mesh belong to the "foreground" (the part we want to cut out) and the rest to the background. The subpart of interest is then cut out from the mesh in real time via an efficient region growing segmentation algorithm. This paper proposes a new distance between adjacent vertices. This paper presents the improved-methods set the vertices that are both adjacent to object and background as special vertices, and use cutting results as object and background seeds to do the region growth again. The experiments show that the result is more accurate because most vertices of the object and background are selected as seeds.

#### REFERENCES

- [1] Yi, B., Liu, Z., Tan, J., Cheng, F., Duan, G., and Liu, L., "Shape Recognition of Cad Models Via Iterative Slippage Analysis," *Comput-Aided Des.*, 2014,55(10), pp. 13-25.
- [2] Z.Ji,L.Liu,Z.Chen,and G.Wang.Easy mesh cutting[C].In *Computer Graphics Forum (Eurographics 2006)*, 25(3): 283-291.
- [3] P. Sander, J. Snyder, S. Gortler, and H. Hoppe, "Texture mapping progressive meshes," in *Proceedings of ACM SIGGRAPH*, pp. 409-416, 2001.
- [4] B. Levy, S. Petitjean, N. Ray, and J. Maillot, "Least squares conformal maps for automatic texture atlas generation," in *ACM Computer Graphics, Proc. SIGGRAPH 2002*, pp. 362-371, 2002.
- [5] P. Sander, Z. Wood, S. Gortler, J. Snyder, and H. Hoppe, Multi-chart geometry images, in *Proceedings of the Eurographics Symposium on Geometry Processing*, pp. 146-155, 2003.
- [6] D. Page, M. Abidi, A. Koschan, and Y. Zhang, "Object representation using the minima rule and superquadrics for under vehicle inspection," in *Proceedings of the 1st IEEE Latin American Conference on Robotics and Automation*, pp. 91-97, 2003.
- [7] Shamir A.A formulation of boundary mesh segmentation[C]. *Proceeding of the International Symposium on 3D Data Processing, Visualization and Transmission*, 2004.
- [8] Jagannathan A, Eric L.Miller.Three Dimensional Surface Mesh Segmentation Using Curvedness-Based Region Growing Approach[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007,29(12):2195-2004.
- [9] Garland M,Willmott A,Heckbert.Hierarchical face clustering on Polygonal surface[C].*Proceedings of ACM Symposium on Interactive 3D Graphics*,2001:49-58.

- [10] Shlafman S, Tal A, Katz S. Metamorphosis of Polyhedral surfaces using decomposition[C].Proceeding of Annual Conference of the European Association for Computer Graphics,2002:219-228.
- [11] T.Funkhouser,M.Kazhdan,P.Shilane,P.Min,W.Kiefer,A.Tal,S.Rusinkiewicz and D.Dobkin. Modeling by example[J].ACM Transactions on Graphics, 2004,23(3):652–663.
- [12] M.Zockler,D.Stalling,and H.-C.Hege. Fast and intuitive generation of geometric shape transitions[J]. The Visual Computer, 2000,16(5):241–253.
- [13] C.J.Armstrong,B.L.Price,W.A.Barrett.Interactive segmentation of image volumes with live surface[J].Computers&Graphics, ,2007,31(2):212–229.
- [14] Z.Ji,L.Liu,Z.Chen,and G.Wang. Easy mesh cutting[C].In Computer Graphics Forum (Eurographics 2006), 25(3): 283-291.
- [15] Y.Lee,S.Lee,A.Shamir,D.Cohen-Or and H.-P.Seidel.Mesh scissoring with minima rule and part salience[J].Computer Aided Geometric Design, 2005,22(5):444-465.
- [16] Brown S, Morse B and Barrett W.Interactive Part Selection for Mesh and Point Models using Hierarchical Graph-Cut Partitioning[C]. Proceedings of Graphics Interface 2009,23-30.
- [17] Bing Yi, Zhenyu Liu, Guifang Duan and Jianrong Tan.Coarse-to-Fine Extraction of Free-Form Surface Features(J). Journal of Computing and Information Science in Engineering,2015.
- [18] Shlafman S, Tal A and Katz S. Metamorphosis of Polyhedral Surfaces using Decomposition[C]. Computer Graphics Forum, 2002,21(3).
- [19] Chen, X., Golovinskiy, A., and Funkhouser, T. “A Benchmark for 3D Mesh Segmentation.” ACM Trans. Gr., 2009, 28(3): 1–73.
- [20] [20] A. Shamir. A formulation of boundary mesh segmentation. 3D Data Processing. Visualization and Transmission,2004,82-89
- [21] C Yang, H Suzuki, Y Ohtake, T Michikawa. Boundary smoothing for mesh segmentation. IEEE International Conference on Computer-aided Design & Computer Graphics,2009, 10(1):241-248
- [22] M Meng, L Fan, L Liu. A comparative evaluation of foreground/background sketch-based mesh segmentation algorithms. Computers & Graphics, 2011, 35(3):650-660