Interactive Mesh Segmentation Based On Graph Laplacian

Enyang Gao[1, 2,a], Weijun Liu[1,b] and Tianran Wang[3,c]

[1]Shenyang Institute of Automation Chinese Academy of Sciences, Shenyang, P.R.China

[2]Graduate School of The Chinese Academy of Sciences, Beijing, P.R.China

[a]gaoenyang@sia.cn, [b]wjliu@sia.cn, [c]trwang@sia.cn

Abstract. This paper introduces a novel algorithm that decomposes a given shape into meaningful parts requiring only strokes to specify foreground and background regions. the user is asked to draws freehand sketches to provide some facets as belonging to the desired part of the surface, and then an energy function is constructed based on graph Laplacian. Finally, a solution of minimizing energy function is provided and then segmentation are finished. We have presented an effective interactive system with an easy-to-use UI for mesh segmentation. The experiment results show that our algorithm is robust, fast, and capable of producing satisfactory results with regard to the human intuition and geometric attributes.

Introduction

The problem of segmenting a 3D shape into meaningful parts has become an important and challenging problem in computer graphics, with applications in areas such as modeling [3], morphing[4] , mesh parameterizations [22,28] , 3D shape retrieval[1] , collision detection[15], texture mapping[26] and skeleton extraction[7].

Related work a lot of approaches have been discussed in the past for mesh segmentation. Previous mesh segmentation methods can be classified into two classes: fully automatic methods and interactive methods.

Automatic mesh segmentation: A variety of different methods for automatically partitioning a mesh into parts have been discussed in the past. These methods can be classified into part-type and patch-type [21]. In part-type segmentation, the goal is to segment the object represented by the mesh into meaningful, mostly volumetric, parts, and in surface-type segmentation the objective is to partition the surface mesh into patches under some criteria. Part-type segmentation is worth our concern because our goal is to develop intuitive and semantic decompositions to the object unlike patch type segmentations used for parameterizations, rendering, or mesh processing.

The majority of the methods found in mesh segmentation are derived from the idea of two-dimensional image segmentation, such as Watersheds methods [19][8], Spectral methods[17],mean shift method[25][18],and so on. There are also some

notable differences between them and 3D boundary mesh segmentation. Images are highly regular and are not embedded in higher dimensional space.

When geometry and topology information were used, there will be better results. [10] and [24] proposed the methods based on the curvature tensor analysis, criterion based on the curvature information were proposed. Producing results consistence both with intuition and semantics, some other authors [14] consider homogeneous curvature properties.

A data-driven approach to simultaneous segmentation and labeling of parts in 3D meshes was proposed by [6]. An objective function is formulated as a Conditional Random Field model, the objective function is learned from a collection of labeled training meshes. The algorithm uses geometric and contextual label features.

Interactive mesh segmentation: recently, interactive mesh segmentation has been popular in computer graphics while fully automatic segmentation still remains a hard problem especially because it concerns semantics of the mesh shape. In general, they can be classified into two categories: boundary based approaches and region based approaches.

In boundary based approaches [11][12][13][2], the user is asked to provide an initial area that is closed to the desired cut. To execute a cut, the user needs to sketch along the cut boundary to specify a sparse set of points. These methods originated from [16]. The drawback of the approaches is that drawing a contour is not an easy thing and is hard and tedious to control.

In region based approaches, regional information are used as input to provide an initial labeling of few surfaces. The user draws strokes to specify foreground and background regions for each cut.The easy mesh cutting [5] is the representative approaches. Users can cut meaningful components from meshes by simply drawing freehand sketches on the mesh, obtain the cutting results based on an improved region growing algorithm using a feature sensitive metric. The random walks approach [9] provides fast mesh segmentation according to the probability value computed by minimizing Dirichlet energy. Cross-Boundary Brushes approach [27] roughly draws one or more strokes across a desired cut and automatically returns a best cut running through all the strokes.

Our approach In this paper, we develop an interactive mesh segmentation method based on graph Laplacian. Our work is motivated by the semi-supervised spectral clustering. As shown in fig.1. The user simply and quickly draws freehand sketches on the mesh in our system. The freehand strokes roughly mark out the subpart of foreground and background. The foreground of interest is then cut out from the mesh based on an improved feature sensitive metric and a prior that the nearing facets are likely to be in the same class(foreground or background).

Our method is composed of three steps: First, the user is asked to draws freehand sketches toprovide some facets as belonging to the desired part of the surface. Second, an energy function is constructed based on graph Laplacian. Third, a solution of minimizing energy function is provided and then segmentation are finished. Take the case in 1 for instance, the user freely segments the model into several parts of any size.

Mesh metric and graph Laplacian

Mesh metric Pottmann et al. introduced the isophotic metric [20], a new metric on surfaces, in which the length of a surface curve is not just dependent on the curve itself, but also on the variation of the surface normal along it. A weak variation of the normal brings the isophotic length of a curve close to its Euclidean length, whereas a strong normal variation increases the isophotic length. They consider the field of unit normal vectors $n(x)$ attached to the surface points $x \in S$ as a vector-valued image defied on the surface. One can map each surface point x to a point $x_f = (x, wn)$ where w denotes a non-negative weight, whose magnitude regulates the amount of feature sensitivity and the scale on which one wants to respect features. By measuring distances of points and lengths of curves on the featured manifold instead of S a feature sensitive metric on the surface is introduced. This feature sensitive metric for feature sensitive morphology on surfaces were used in [20].

A distance metric measuring the change of geometric properties along the edges were proposed in [5], which considers both the isophotic metric and the minima rule. For two points i and j on a surface, the distance is defined by:

$$d_\Gamma(i, j) = \int_\Gamma ds + \omega \int_\Gamma ds^* + \omega^* \int_\Gamma g(k_{ij}) ds \tag{1}$$

Where the first two terms account for the isophotic metric, and the last term for the minima rule.

For a facet i and its neighbor facet j on the mesh, the improved isophotic metric distance between them are calculated as the following:

$$d(i, j) = |i - j| + \omega |n_i - n_j| + \omega^* \beta(k_{ij}) \tag{2}$$

Where n_i and n_j are respectively the normals of facets i and j, k_{ij} is the directional curvature along the line direction ij. In our implementation, the curvature function $\beta(k_{ij})$ augment the effect of negative curvature in the improved isophotic metric, and is defined as follows:

$$\beta(k_{ij}) = \begin{cases} k_{ij}, if\ k_{ij} \geq 0 \\ k^2_{ij}, otherwise \end{cases} \tag{3}$$

Graph Laplacian. Let $G = (V, E)$ be an undirected graph with facet set $F = \{f_1, \cdots f_n\}$. In the following we assume that the graph G is weighted by affinity of neighbor facets, that is each edge between two facets i and j carries a

nonnegative weight ω_{ij}.

Symmetric affinity matrix $W \in R^{n \times n}$ is constructed where for all i, j, W_{ij} encodes the affinity of face i and face j as the graph should represent the local neighborhood relationships, this construction is only useful if the similarity function itself models local neighborhoods. In our implementation, such a similarity function is the Gaussian similarity function

$$\omega_{ij} = \exp(d(i, j)^2) / 2\sigma^2 \tag{4}$$

where the parameter σ controls the width of the neighborhoods.

In Euclidean domain R^n, Laplace operator, denoted Δ by or ∇^2, named after Pierre-Simon de Laplace, is a second order differential operator.

Laplace operator is defined as the divergence ($\nabla \cdot$) of the gradient (∇). Thus if f is a twice-differentiable real valued scalar function and $f = [f_1, \cdots f_n]$, then the Laplacian of f is defined by $\Delta f = \nabla^2 f = \nabla \cdot \nabla f$.

Definition of the discrete Laplacian on weighted graph is:

$$(Lf)(v_i) = \sum_{v_i \square v_j} \omega_{ij}(f(v_i) - f(v_j)) \tag{5}$$

and a quadratic form of it is:

$$E = f^T L f = \sum_{v_i \square v_j} \omega_{ij}(f(v_i) - f(v_j))^2 \tag{6}$$

The operator L is equivalent to a matrix called Laplace matrix [23].

Interactive mesh segmentation

In this paper, the interactive information is introduced as markers, which are input by the users to roughly indicate the position and main features of the foreground and background. When considering the variation of the labels on the neighbor facets and interactive information, the segmentation problem is turned into constrained optimal problem below:

$$\begin{aligned} &\min E \\ &s.t.\ f(v_i) = 0\ if\ i \in U \\ &s.t.\ f(v_i) = 1\ if\ i \in V \end{aligned} \tag{7}$$

Denote $U = \{p_1, \cdots p_{c1}\}$ and $V = \{q_1, \cdots q_{c2}\}$ as the corresponding sets of facet indices of the user markers, here c_1 and c_2 is the number of markers, and we let $c = c_1 + c_2$.

Using the Lagrange multiplier method, we can build a new target function:

$$F = E + \sum_{k=1}^{c} \gamma_k g_k \tag{8}$$

where $\gamma_{1 \dots \gamma_c}$ are the unknown scalars associated with each constraint. Minimizing F is equivalent to solving the following system:

$$\begin{cases} \partial F / \partial f_0 & = 0 \\ \dots \\ \partial F / \partial F_{n-c} & = 0 \\ \partial F / \partial \gamma_0 & = 0 \\ \dots \\ \partial F / \partial \gamma c & = 0 \end{cases} \tag{9}$$

Or in matrix mode:

$$\begin{pmatrix} L & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} f \\ \gamma \end{pmatrix} = \begin{pmatrix} 0 \\ f' \end{pmatrix} \tag{10}$$

where $\gamma = [\gamma_0, \cdots, \gamma_C]^T$ and f' is the vector composed by labels corresponding to sets of facet indices of the user markers. The coefficient matrix $B \subseteq R^{n \times c}$ with value 1 at B_{ij} if facet i is the constrained facet.

Experiment results

We have applied our system to segment a vast variety of models, including the recently introduced 3D Segmentation Benchmark [2]and other models with complex structures as well as CAD models. Here we provide some examples to demonstrate the applicability and flexibility of our mesh segmentation method based on graph Laplacian2.We implement our algorithm with Intel 2.93 GHz CPU and 2GB RAM. The program is developed using VS 2005. For solving the linear system in (10), we use TAUCS library. Our algorithm is quite efficient. Our current system is capable of dealing with meshes containing hundreds of thousands of vertices at interactive rates. We believe that our tools are very useful for cutting out components that are hard for existing automatic segmentation methods to detect and segment.

Conclusion

In this paper, we have presented an effective interactive system with an easy-to-use UI for mesh segmentation. Our results are robust to the user input and capable of reflecting geometric features and human shape perception, It can even segment details in on a mesh surface3. Our main contribution is we apply the graph Laplacian for interactive mesh cutting. The experiment results demonstrate that our algorithm is robust, fast, and capable of producing satisfactory results with regard to the user intention and geometric attributes.

In all examples shown in this paper we have used an energy definition that is corresponding to the geometric feature across the surface. Note that the energy we used did not reflect the intrinsic shape information of the surface, but only the simple geometry property. In the future, we will develop the segmentation method based on the special metrics reflecting the intrinsic shape of the surface with regard to human intuition.

References

[1] Attene, M., Falcidieno, B., Spagnuolo, M.: Hierarchical mesh segmentation based on fitting primitives. The Visual Computer 22(3), 181–193 (2006)

[2]. Chen, X., Golovinskiy, A., Funkhouser, T.: A benchmark for 3D mesh segmentation. In: ACM SIGGRAPH 2009 papers, pp. 1–12. ACM (2009)

[3]. Funkhouser, T., Kazhdan, M., Shilane, P., Min, P., Kiefer, W., Tal, A., Rusinkiewicz, S., Dobkin, D.: Modeling by example. In: ACM SIGGRAPH 2004 Papers, p.663. ACM (2004)

[4] Gregory, A., State, A., Lin, M., Manocha, D., Livingston,M.: Interactive surface decomposition for polyhedral morphing.The Visual Computer 15(9), 453–470 (1999)

[5] Ji, Z., Liu, L., Chen, Z.,Wang, G.: Easy mesh cutting. In:Computer Graphics Forum, vol. 25, pp. 283–291. WileyOnline Library (2006)

[6] Kalogerakis, E., Hertzmann, A., Singh, K.: Learning 3Dmesh segmentation and labeling. ACM Transactions onGraphics (TOG) 29(4), 1–12 (2010)

[7] Katz, S., Tal, A.: Hierarchical mesh decomposition using fuzzy clustering and cuts. ACM Transactions on Graphics (TOG) 22(3), 954–961 (2003)

[8] Koschan, A.: Perception-based 3D triangle mesh segmentationusin g fast marching watersheds. In: Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, vol. 2. IEEE (2003)

[9] Lai, Y., Hu, S., Martin, R., Rosin, P.: Fast mesh segmentation using random walks. In: Proceedings of the 2008 ACM symposium on Solid and physical modeling,pp. 183–191. ACM (2008)

[10] Lavou é, G., Dupont, F., Baskurt, A.: A new CAD mesh segmentation method, based on curvature tensor analysis. Computer-Aided Design 37(10), 975–987 (2005)

[11] Lee, Y., Lee, S.: Geometric snakes for triangular meshes. In: Computer Graphics Forum, vol. 21, pp. 229–238. WileyOnline Library (2002)

[12] Lee, Y., Lee, S., Shamir, A., Cohen-Or, D., Seidel, H.: Intelligent mesh scissoring using 3d snakes (2004)

[13] Lee, Y., Lee, S., Shamir, A., Cohen-Or, D., Seidel, H.: Mesh scissoring with minima rule and part salience. Computer Aided Geometric Design 22(5), 444–465 (2005)

[14] L évy, B., Petitjean, S., Ray, N., Maillot, J.: Least squares conformal maps for automatic texture atlas generation. ACM Transactions on Graphics 21(3), 362–371 (2002)

[15] Li, X., Woon, T., Tan, T., Huang, Z.: Decomposing polygon meshes for interactive applications. In: Proceedings of the 2001 symposium on Interactive 3D graphics, pp. 35–42. ACM New York, NY, USA (2001)

[16] Li, Y., Sun, J., Tang, C., Shum, H.: Lazy snapping. ACM Transactions on Graphics (ToG) 23(3), 303–308 (2004)

[17] Liu, R., Zhang, H.: Segmentation of 3D meshes through spectral clustering. In: Proc. Pacific Graphics, vol. 4, pp. 298–305. Citeseer (2004)

[18]. Liu, R., Zhang, H.: Mesh segmentation via spectral embedding and contour analysis. In: Computer Graphics Forum, vol. 26, pp. 385–394. Wiley Online Library (2007)

[19] Mangan, A., Whitaker, R.: Partitioning 3D surface meshes using watershed segmentation. Visualization and Computer Graphics, IEEE Transactions on 5(4), 308–321 (2002)

[20] Pottmann, H., Steiner, T., Hofer, M., Haider, C., Hanbury, A.: The isophotic metric and its application to feature sensitive morphology on surfaces. Computer Vision- ECCV 2004 pp. 18–23 (2004)

[21] Shamir, A.: A survey on mesh segmentation techniques. In: Computer Graphics Forum, vol. 27, pp. 1539–1556. Wiley Online Library (2008)

[22] Sheffer, A., Praun, E., Rose, K.: Mesh parameterization methods and their applications. Foundations and Trends in Computer Graphics and Vision 2(2), 171 (2006)

[23] Von Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17(4), 395–416 (2007)

[24] Yamauchi, H., Gumhold, S., Zayer, R., Seidel, H.: Mesh segmentation driven by Gaussian curvature. The Visual Computer 21(8), 659–668 (2005)