A Robust Segmentation Algorithm for 3D Complex Meshes

Yu Hou, Yong Zhao*
School of Mathematical Sciences, Ocean University of China, Qingdao, China

*Corresponding author: zhaoyong@ouc.edu.cn

Abstract. Segmenting 3D mesh is a difficult topic in computer graphics. This paper proposes a new mesh segmentation algorithm. First, we over-segment the input mesh to generate a series of superfacets. Second, we introduce a novel region fusion algorithm to obtain the segmentation result. Various experiments show that our algorithm is able to handle 3D meshes with complex shape and rich details. Even though the input mesh has noises or holes our algorithm can still achieve satisfactory results.

Keywords: Mesh segmentation, region fusion, robustness.

1. Introduction
With the development of computer technology, the segmentation of 3D mesh models is widely applied in the fields of computer graphics. The segmentation of 3D models helps to analyze and understand the model, which has great significance to the subsequent geometric processing. A large number of applications, such as shape retrieval and editing depend on the ideal segmentation result.

The complex topological structure and rich geometry features of 3D mesh model bring many difficulties to mesh segmentation. First, researchers proposed automatic segmentation algorithms [1, 2, 3]. Subsequently, interactive segmentation algorithms [4, 5] allow users to add constraints through different strokes. With the development of deep neural networks, researchers use deep learning techniques to further improve the segmentation quality [6, 7, 8].

This paper proposes a segmentation algorithm which makes full use of topological structure and geometric features. The dense sampling of 3D mesh model will lead to relatively low algorithm efficiency if the triangular facets are directly processed. This paper first divides the input mesh into a series of superfacets. Processing on the basis of superfacets can improve algorithm efficiency greatly.

Then, according to shape features of superfacets such as normal, Gaussian curvature(GC) [9], shape diameter function(SDF) [10], average geodesic distance(AGD) [11], conformal factor(CF) [12] and heat kernel signature(HKS) [13], we define intra-regional difference, inter-regional difference as well as fusion conditions, and merge similar superfacets to obtain the final segmentation result.

To verify the effectiveness and robustness of the proposed algorithm, we conduct segmentation experiments on mesh models with complex shapes or rich details. In particular, our algorithm can obtain satisfactory segmentation results even on models with noises or holes.

2. Related work
Early research belongs to automatic segmentation. Katz et al. [1] used fuzzy clustering and minimum cut techniques. Golovinskiy et al. [14] integrated segmentation boundaries of K-means, hierarchical...
clustering, and minimum cut. Chen et al. [2] proposed a 3D mesh segmentation benchmark, which includes mesh data set, manual segmentation results, and segmentation criteria. Au et al. [3] constructed scalar fields that can reflect shape concavity, and segmentation boundaries are isolines of scalar fields in concave region.

Interactive segmentation algorithms were then introduced to improve segmentation quality. Ji et al. [4] used a sketch-based interface to draw a few strokes on the mesh, specified foreground and background regions, and obtained segmentation results through region growing. Subsequently, different interaction schemes have been proposed. Zheng et al. [15] proposed cross-boundary stroke, Fan et al. [16] proposed foreground strokes, while Zheng et al. [5] only specified a boundary point.

In recent years, deep neural networks have been used to further improve segmentation quality. Guo et al. [6] reorganized the features of 3D models into 2D matrix as the network input, and learned the initial label information. Kalogerakis et al. [7] combined convolutional neural networks and conditional random fields to obtain segmentation result. To adapt to the irregularity of 3D mesh, Yi et al. [17] used graph convolutional neural networks in spectral space. Xu et al. [8] proposed rotation-invariant convolution and pooling operations, and then gave a two-stream framework.

3. Mesh segmentation algorithm
This part will detail our algorithm, including over-segmentation and region fusion.

3.1. Over-segmentation
Composed of a large number of triangular facets, the 3D mesh model has complex topological structure and rich geometric features. Therefore, huge computational cost is needed if these triangular facets are processed directly. Therefore, we over-segment the input mesh into a series of superfacets by binary space partition [18] to improve algorithm efficiency.

Binary space partition is performed recursively for each subspace until the number of triangular facets in the current space and the surface variation are less than user-specified thresholds. When the space partition is finished, the triangular facets in each subspace constitute a superfacet. The set of superfacets is denoted as \( F = \{F_i | 1 \leq i \leq |F|\} \), where \(|F|\) is the number of superfacets. Because the number of superfacets is far less than the number of triangular facets, segmentation on the basis of superfacets can greatly improve the efficiency and accuracy of segmentation.

3.2. Region fusion

3.2.1. Feature calculation. We measure mesh model using shape features such as normal, GC [9], SDF [10], AGD [11], CF [12], HKS [13]. For superfacet \( F_i \), its shape features can form a feature vector denoted as \( T(F_i) \). Then, the difference between adjacent superfacets \( F_i \) and \( F_j \) can be defined as:

\[
d(F_i, F_j) = \|T(F_i) - T(F_j)\|
\]

3.2.2. Fusion process. Each superfacet represents a local region, and the final segmentation result can be obtained by fusing them. Below we will introduce intra-regional difference, inter-regional difference, and fusion conditions.

Suppose \( R \) is a region, the intra-regional difference can be defined as:

\[
D(R) = \max_{i,j,F_i,F_j \in R} d(F_i, F_j)
\]

Where \( F_i \) and \( F_j \) are adjacent superfacets.

Suppose \( R_i \) and \( R_j \) are two adjacent regions, their inter-regional difference is defined as:
\[ \text{Dis}(R_i, R_j) = \min_{F_i \in R_i, F_j \in R_j} d(F_i, F_j), \] (3)

Where \( F_i \) and \( F_j \) are adjacent superfacets belonging to different regions.

If the inter-regional difference between two adjacent regions is less than the minimum value of the intra-regional differences of the two regions, then they are merged into a new region. The fusion condition can be expressed as:

\[ \text{Dis}(R_i, R_j) \leq \min(D(R_i) + t(R_i), D(R_j) + t(R_j)). \] (4)

\( t(\cdot) \) is a threshold function which is defined as follows:

\[ t(R) = \frac{m}{|R|}, \] (5)

Where \(|R|\) is the number of superfacets in the region \( R \), and \( m \) is a constant. \( t(\cdot) \) is to adjust fusion condition. Initially, each region contains only one superfacet, so \( D(R) = 0 \). The fusion process fails to start if there is no \( t(\cdot) \) function in equation (4).

We sort the inter-regional differences of all adjacent regions and perform the fusion with the order from small to large. After the fusion of two regions, their intra-regional differences and inter-regional differences with other adjacent regions need update. The fusion process can continue until all adjacent areas fail to meet fusion condition. Finally, each region constitutes as a segmentation part.

4. Experimental results and analysis

We use C++ to implement this algorithm. We have conducted segmentation experiments on various mesh models, some of which have problems such as noises or holes. Figure 1 shows the segmentation results of different types of mesh models, such as Airplane, Ant, and Glasses. The proposed algorithm can make full use of local shape features and merge regions belong to a same part to get perception-aware results.

![Figure 1. Segmentation results of different types of mesh models.](image-url)
Figure 2. Segmentation results of Octopus model with impulse noises. Impulse noises with the intensity of 0.3 and 0.4 are added in models of a and b along random direction.

Figure 3. Segmentation result of Cup model with holes.

In Figure 2, we added different impulse noises to Octopus model along random directions. Noises will affect the calculation of shape features and bring difficulties to segmentation process. Experiments show that our algorithm can obtain satisfactory segmentation results under noises of different intensities. Figure 3 shows the segmentation result of Cup model with holes. There are holes of different sizes. Similar regions still have similar shape features though existing holes, so the algorithm can still get ideal segmentation results.

5. Conclusions
This paper proposes a robust mesh segmentation algorithm. Our algorithm can make full use of shape features to handle mesh models of various types. Our algorithm can still obtain perception-aware segmentation results even if there are noises or holes in the model. In the future, we will extend the proposed algorithm to point clouds.

Acknowledgments
This work was financially supported by Natural Science Foundation of Shandong Province (ZR2018MF006).

References
[1] S. Katz, A. Tal, Hierarchical mesh decomposition using fuzzy clustering and cuts, ACM Transactions on Graphics, 2003, 22(3), pp. 954-961.
[2] X.B. Chen, A. Golovinskiy, T. Funkhouser, A benchmark for 3D mesh segmentation, ACM Transactions on Graphics, 2009, 28(3), Article No.73.
[3] O.K.C. Au, Y.Y. Zheng, M.L. Chen, P.F. Xu, C.L. Tai, Mesh segmentation with concavity-aware fields, IEEE Transactions on Visualization and Computer Graphics, 2012, 18(7), pp. 1125-1134.
[4] Z.P. Ji, L.G. Liu, Z.G. Chen, G.J. Wang, Easy mesh cutting, Computer Graphics Forum, 2006, 25(3), pp. 283-291.

[5] Y.Y. Zheng, C.L. Tai, O.K.C. Au, Dot scissor: a single-click interface for mesh segmentation, IEEE Transactions on Visualization and Computer Graphics, 2012, 18(8), pp. 1304-1312.

[6] K. Guo, D.Q. Zou, X.W. Chen, 3D mesh labeling via deep convolutional neural networks, ACM Transactions on Graphics, 2015, 35(1), Article No.3.

[7] E. Kalogerakis, M. Averkiou, S. Maji, S. Chaudhuri, 3D shape segmentation with projected convolutional networks, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society Press, Los Alamitos, 2017, pp. 6630-6639.

[8] H.T. Xu, M. Dong, Z.C. Zhong, Directionally convolutional networks for 3D shape segmentation, in Proceedings of the IEEE International Conference on Computer Vision, IEEE Computer Society Press, Los Alamitos, 2017, pp. 2717-2726.

[9] M. Meyer, M. Desbrun, P. Schroder, A.H. Barr, Discrete differential-geometry operators for triangulated 2-manifolds, in Proceedings of Third International Workshop "Visualization and Mathematics", Springer, Heidelberg, 2002, pp. 35-57.

[10] L. Shapira, A. Shamir, D. Cohen-Or, Consistent mesh partitioning and skeletonisation using the shape diameter function, The Visual Computer, 2008, 24(4), pp. 249-259.

[11] M. Hilaga, Y. Shinagawa, T. Komura, T.L. Kunii, Topology matching for fully automatic similarity estimation of 3D shapes, in Proceedings of ACM SIGGRAPH, ACM, New York, 2001, pp. 203-212.

[12] M. Ben-Chen, C. Gotsman, Characterizing shape using conformal factors, in Proceedings of 1st Eurographics Workshop on 3D Object Retrieval, Eurographics Association Press, Aire-la-Ville, 2008, pp. 1-8.

[13] J. Sun, M. Ovsjanikov, L.J. Guibas, A concise and provably informative multi-scale signature based on heat diffusion, Computer Graphics Forum, 2009, 28(5), pp. 1383-1392.

[14] A. Golovinskiy, T. Funkhouser, Randomized cuts for 3D mesh analysis, ACM Transactions on Graphics, 2008, 27(5), pp. Article No.145.

[15] Y.Y. Zheng, C.L. Tai, Mesh decomposition with cross-boundary brushes, Computer Graphics Forum, 2010, 29(2), pp. 527-535.

[16] L.B. Fan, L.G. Liu, K. Liu, Paint mesh cutting, Computer Graphics Forum, 2011, 30(2), pp. 603-612.

[17] L. Yi, H. Su, X.W. Guo, L.J. Guibas, SyncSpecCNN: synchronized spectral CNN for 3D shape segmentation, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society Press, Los Alamitos, 2017, pp. 6584-6592.

[18] M. Pauly, M. Gross, L.P. Kobbelt, Efficient simplification of point-sampled surfaces, in Proceedings of the IEEE Visualization 2002, IEEE Computer Society Press, Los Alamitos, 2002, pp. 163-170.