Communications in Information and Systems
Volume 17, Number 2, 65–83, 2017

Interactive mesh cutting with Laplace coordinates and gradient

Bin Liu, Weiming Wang*, Junjie Cao†, and Xiuping Liu‡

Sketch-based mesh cutting has been attended by researchers in Computer Graphics, mainly due to its success in dealing with semantic information of mesh with little user interaction. Although most existing approaches have gained great improvement, they depend on the predefined geometric features sensitive to shape and complex mathematical theory, and most methods can not generate a global unique solution for the segmentation problem. In this paper, we propose a novel sketch-based mesh segmentation model with Laplace Coordinates and gradient, which is independent of the complex geometric features and could perceive the differences of mesh parts. Furthermore, our algorithm is easy to implement, mathematically simple, and a global unique solution can be guaranteed because of our convex quadratic model. Benefiting from the Laplace Coordinates and gradient, our method holds an anisotropic behavior and can better fit the cutting boundary. Namely, triangular faces sharing similar attributes are kept closer to each other while big jumps naturally happen on the boundary between mesh parts when our method is applied. A large number of experiments illustrate the enhanced efficacy of our method compared with the state-of-the-art techniques.

1. Introduction

The research of digital geometry processing is gradually switching to the structure analysis in recent years, aiming at acquiring the high-level semantic information. Because interactive mesh cutting technologies bring about an intuitive and flexible mode for semantic information, a plenty of mesh cutting methods have emerged [2, 6, 13, 16, 17, 20, 24, 25, 28, 31, 32]. The

*Research supported in part by NSFC (61702079), 2016M601308 and DUT16RC(3)061.
†Research supported in part by NSFC (61572099, 61772104.
‡Research supported in part by NSFC (61370143).

common procedure of interactive mesh cutting consists of two folds: first, users draw some rough sketches on the meaningful regions of mesh surface, then mesh cutting algorithms are used to propagate sketches for generating final semantic segmentation. Besides, interactive mesh cutting methods can serve as basic tools in many fields, such as geometric modeling [8], shape editing and deformation [14], shape retrieval [1] and texture mapping [18].

As described in [19], different theories are used to achieve interactive mesh cutting: Region Growing [13, 24], Random Walker [16, 28], Graph Cuts [2, 6], Hierarchical Aggregation [25], Harmonic Field [31], etc. Their differences are mathematical formulas, the definition of geometric features sensitive to shape, pairwise vertex/face distance and edge weight computation. Most methods need minimize a non-quadratic energy functional in order to effectively produce the cutting results. Therefore, sophisticated and costly optimization tools are demanded to be used. In addition to the random walker techniques, the uniqueness of other solutions are not be guaranteed. However, random walker does not have the anisotropic property [29]. As a result, the cutting boundaries cannot be well controlled with random walker. How to design a shape sensitivity metric is another issue for most methods in interactive mesh segmentation.

In this paper, we propose a novel sketch-based mesh cutting model with Laplace Coordinates and gradient defined on the dual graph of mesh. It is independent on the input shape metric and could perceive the differences of mesh parts. As the proposed model is a quadratic energy functional defined on an affinity graph, it can be effectively solved, and the uniqueness of the solution can be guaranteed because of the properties of convex function. Benefiting from the propagation ability of Laplace Coordinates, the triangular faces with the similar properties will be labelled as the same class. To further shrink the cutting boundaries, the gradient term is designed and integrated in our optimization formulation. Furthermore, the proposed method can well control the anisotropic propagation of labels during segmentation [3]. Consequently, triangular faces with similar attributes are automatically gathered together and they are naturally separated on the boundaries. Moreover, our model can also achieve multiple parts segmentation on one section, unlike other methods [6, 21] by iterative or progressive strategies.

The proposed mesh cutting technique is specially simple, which contains four main steps: 1) drawing meaningful sketches, 2) constructing affinity graph on triangle mesh, 3) building and solving optimization formulation, 4) Assigning labels to triangle faces. Since the proposed optimization framework is quadratic, it can be effectively solved through a linear system after

derivation, making the proposed methodology quite simple to be coded. Benefiting from the Laplace Coordinates and gradient, the proposed algorithm slightly outperforms the state-of-the-art interactive mesh cutting methods on effectiveness and performance.

2. Relate work

Although researchers could understand the meaningful object structure through sketches in different ways [7], such as foreground/background sketch-based, foreground sketch-based, cross-boundary sketch-based and along-boundary sketch-based, most mesh cutting approaches can be classified into 3 classes: Region Growing [13, 24], Random Walker [16, 28] and Graph Cuts [2, 6]. In this section, we only introduce some representative works about algorithm design.

The easy mesh cutting method [13] is the representative of region growing algorithm. The idea behind region growing algorithm is to implement the minima rule on a local height computation [15]. By defining a good mesh subpart feature, the algorithm can recognize the structure boundary through growing the corresponding region incrementally. Although region growing algorithm is very important in the field of computer vision and digital geometry processing, its ability in fitting segmentation boundaries is not quite effective under the situation of local irregular gradient [3]. In addition, the design of shape sensitivity feature need complex and comprehensive geometry knowledge, such as various curvatures. As a contrast, our method is insensitive to input shape metrics.

The random walker algorithm, presented in [16], provides mesh segmentation according to the probability value computed by minimizing a Dirichlet energy, which is similar in spirit to the corresponding method for image segmentation [10]. Its theoretical basis is standard graph Laplacian formulation $Lx = 0$, where L is a matrix built from an edge weight matrix W and a diagonal weighted valency matrix D. Lai et al. [16] use initial seeds to segment mesh, and then merge the regions found based on shape sensitivity feature. Each non-seed triangle is determined to belong to the foreground region if the triangle has a higher probability in reaching foreground seeds than others. Their solution also can be figured out by a linear system. However, it was shown in [29] that random walker prefer to produce the "flatter" solution because of its isotropy. Furthermore, it is not quite effective when capturing mesh segmentation boundaries. Our method inherits the advantages of random walker while compensates its defects.

Brown et al. [2] and Fan et al. [6] utilize the Graph Cuts techniques to achieve the mesh semantic parts segmentation. The rational is the graph-cut formulation, finding the minimum cut by the hard constrains provided by the user interactions and the soft constraints provided by the connection relationships between adjacent vertices/faces of the mesh model. A min-cut/max-flow algorithm usually deals the problem of unique solution and reaches a good segmentation. Brown et al. [2] introduce a hierarchical acceleration based on a high-level octree hierarchy for fast cutting. Fan et al. [6] achieve the mesh cutting tasks by Graph Cuts based on the Gaussian mixture models on the shape diameter function metric of the shape. However, these methods depend on complex mathematical foundations and solution algorithms.

Other interactive mesh cutting methods, such as Hierarchical Aggregation [25] and Harmonic Field [21, 31]. Xiao et al. [25] employ a hierarchical method for extracting features through local adaptive aggregation. It needs the statistics of curvature to define the multi-scale similarity measure in the processing of neighboring aggregation. Meng et al. [21] and Zheng et al. [31] use a harmonic field generated by Possion equation $\triangle\Phi = 0$ to complete the interactive mesh segmentation. Then either Graph Cuts technique need to be applied or best isoline need to be calculated for obtaining better segmentation boundaries. Nevertheless, the shape sensitivity feature has an impact on the segmentation boundary.

In recent years, automatic mesh segmentation obtains big improvements because of the thriving of machine learning. Sidi et al. [23] introduce a descriptor-space spectral clustering method for co-segmentation of shapes. Xie et al. [26] apply extreme learning machine for 3D shape segmentation and labeling. Guo et al. [11] use deep convolutional neural networks to acquire the 3D mesh labeling. However, these methods depend on plentiful meshes for training good segmentation models and they are inflexible in understanding the semantic information.

3. Laplace coordinates for interactive mesh cutting

In this section, we address our novel model for sketch-based interactive mesh cutting. First, an adjacency graph is constructed with a common geometric feature, which will be introduced in Sec. 3.1. Second, the details of Laplace Coordinates and gradient based on interactive mesh cutting optimization framework will be given in Sec. 3.2. Finally, the solution of our formulation will be derived and the uniqueness of the solution will be analyzed in Sec. 3.3

(a) SDF metric (b) Curvature metric (c) Normal metric

Figure 1: Comparison among SDF metric, curvature metric and common normal metric. Our results are almost similar with the same sketches as input semantic labels.

3.1. Graph construction

Considering the mesh M as a graph, we construct the graph $G = \{V, E\}$ on its dual graph, whose V is the set of face centriods of the mesh and E is the set of edges connecting with neighboring face centroids. As described in [19], features can be used to design shape metric on V only if they are sensitive to the parts of mesh. Therefore, some mesh cutting methods [6, 13, 24] gingerly predefine a good shape metric by integrating some simple geometrical features, which is quite complicated in designing. However, some features are difficult to be calculated while the others are simple.Through a large number of experiments, we find that the proposed method is insensitive to geometrical features.

Figure 1 shows the interactive cutting results with different geometrical features under the same sketches: (a) The shape diameter function (SDF), introduced in [22], (b) Curvature [9], (c) Face Normal. From this figure we can see that the cutting results are almost similar, and the cups are completely segmented into two meaningful parts. SDF is a pose-invariant

part-aware shape metric which measures the diameter of the object's volume in the neighborhood of each point on the surface [6]. In addition, it also has some robustness in noises. So, to better deal with these situations during mesh cutting, we use the SDF as our feature metric in this paper.

For a triangular face f_i of the input M, the weight $w_{ij} = w(f_i, f_j)$ of neighboring faces is computed as follows:

$$w_{ij} = exp\left(-\frac{\beta(r_i - r_j)^2}{\sigma^2}\right), \tag{1}$$

where $\sigma = \max_E |r_i - r_j|$, r_i and r_j are the shape diameter function values of f_i and f_j, and β is a constant parameter which is set to 1 in our paper. Because $w_{ij} = w_{ji}$, the weight matrix is symmetric.

3.2. Quadratic energy functional

Given a set of sketches $\{C_1, ..., C_K\}$, $C_i = \{f_{i1}, f_{i2}, ..., \}$ denotes the set of triangular faces on the ith sketch, K is the number of sketches. Then we define a label vector $\mathbf{x_{C_k}}$ with length K for each set C_i as follows:

$$\mathbf{x_{C_k}} = \begin{cases} x_{C_k}(i) = 1, & if\ i = k \\ x_{C_k}(i) = 0, & otherwise. \end{cases} \tag{2}$$

where, $k = 1, ..., K$. Based on the above definition, we proposed the following quadratic energy functional for interactive mesh segmentation:

$$\begin{aligned} E(\mathbf{X}) = \mu_k \sum_{k=1}^{K} \sum_{f_i \in C_k} \|\mathbf{x_i} - \mathbf{x_{C_k}}\|_2^2 + \lambda \sum_{f_i \in F} \left\| d_i \mathbf{x_i} - \sum_{f_j \in N(f_i)} w_{ij} \mathbf{x_j} \right\|_2^2 \\ + \eta \sum_{f_i \in F} \sum_{f_j \in N(f_i)} \|\mathbf{x_i} - \mathbf{x_j}\|_2^2, \end{aligned} \tag{3}$$

where $\mathbf{X}$ denotes a $n \times K$ label index matrix, n is the number of faces, $\mathbf{x_i}$ represents the label index vector of f_i which is the ith row in $\mathbf{X}$, $\{\mu_k\}$, λ, η are trade-off parameters, F represents the triangular face set, $N(f_i)$ is the neighborhood faces of f_i, which is defined by sharing the same edge, and $d_i = \sum_{f_j \in N(f_i)} w_{ij}$ is the weighted valency of f_i.

Energy functional (3) is the weighted sum, which contains three items. The first item is a data term, which is used to approximate the label index vectors of faces in $\{C_1, C_2, ..., C_K\}$ as known label index vectors in Eq. (2),

the second item is a propagation term, which is applied to the propagation of the label index vectors in the neighboring faces, and the last one is a gradient term which is used to shrink the cutting boundaries.

Once Eq. (3) is minimized with respect to $\mathbf{X}$, the segmentation results can be obtained by specifying the label $y_i \in \{1, ..., K\}$ to face $f_i \in F$, as follows:

$$y_i = k, \;\; if \; x_i(k) = \|\mathbf{x_i}\|_\infty, \tag{4}$$

where $\|\cdot\|_\infty$ represents the maximum of absolute value in its elements.

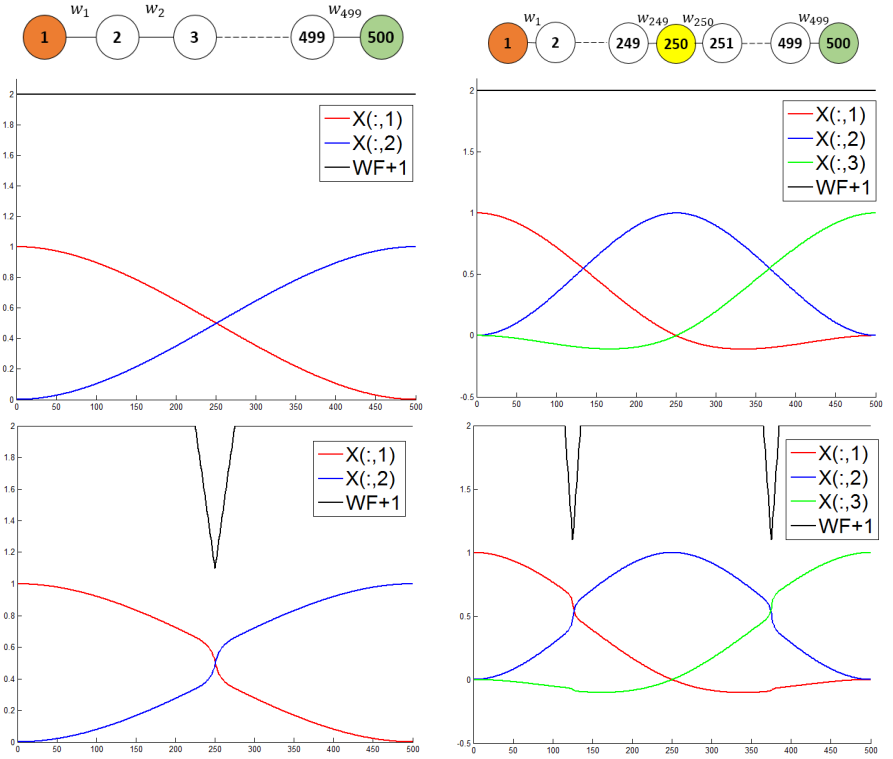
The propagation item is a simplified form of Laplace Coordinates. It is widely used to handle the geometrical problems in the field of geometry processing [27]. Let

$$\delta_i = \mathbf{x_i} - \frac{1}{d_i} \sum_{j \in N(i)} w_{ij} \mathbf{x_j}. \tag{5}$$

From the above definition we can see that δ_i evaluates the error between its current label and the weighted average of its neighbors. The propagation term can effectively spread labels to the neighboring faces with similar face normal and this propagation process stops when large normal jump is encountered.

Figure 2 illustrates some properties of Laplace Coordinates at the case of 1D graph with 500 nodes ordered from left to right. The colored nodes are seeds as the constraint of data term. The black curves describe the distribution of edge weights, unit weight or designed weight. The jump of weight implies that the graph has "cutting boundary". The red, blue and green curves are the each dimension distribution of the solution. For example, the distribution of $X(:, 1)$ is shown in the red curve. Note that, for composing, we plus 1 to the weight function (WF) when plotting its distribution. The solutions depict that the Laplace Coordinates have the remarkable ability in propagating the information of seeds. Moreover, the solution of designed weight emerges big jump at the boundary compared with the solution of unit weight. Namely, nodes sharing similar attributes will be kept closer to each other, showing the anisotropic behavior. In addition, the distributions of solution also explain the fitting boundary capacity of Laplace Coordinates. The results show the cutting boundaries, which can be obtained through comparing the values of each dimension of solution.

The propagation item has the ability to perceive the meaningful boundaries of mesh parts, see the top row in the Fig 3. For better fitting boundary, we design a regularization term composed of gradients, which is called



(a) Two seeds propagation. (b) Three seeds propagation.

Figure 2: Comparison between the solutions of different edge weights, unit weight or designed weight. Graphs are shown in the top row, whose colored nodes are seeds. The solutions (red, green and blue curves) are shown in the second and third rows while the corresponding edge weight functions are shown in the black curves.

boundary shrinkage. It estimates the similarity between its current label and its neighbors on boundary. Making it become small will shrink cutting boundary. Figure 3 shows the cutting boundary of Bear model without and with the gradient term. From this figure we can clearly see that the cutting boundary is much better after the gradient term is added. Our model possesses anisotropy benefiting from the Laplace Coordinates and boundary gradient.

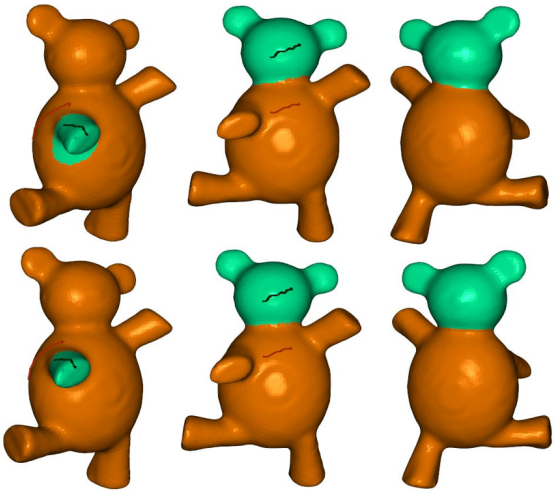


Figure 3: First row: the results without gradient item. Second row: the results with gradient item. The third column is the back view of the second column.

3.3. Solution and analysis

The label propagation term can be rewritten in matrix form as follows:

$$\sum_{i\in F}\left\| d_i\mathbf{x_i} - \sum_{j\in N(i)} w_{ij}\mathbf{x_j}\right\|_2^2 = \|\mathbf{LX}\|_2^2, \tag{6}$$

where, $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the graph Laplace matrix, $\mathbf{D}$ is the diagonal matrix with $D_{ii} = d_i$, and $\mathbf{W}$ denotes the weighted adjacency matrix of the graph,

$$W_{ij} = \begin{cases} w_{ij}, & \text{if } (f_i, f_j) \in E \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

The boundary shrinkage item can also be rewritten as:

$$\sum_{i\in F}\sum_{j\in N(i)} \|\mathbf{x_i} - \mathbf{x_j}\|_2^2 = \|\mathbf{SX}\|_2^2, \tag{8}$$

where, S subjects to

$$S_{(3i+k,j)} = \begin{cases} 1, & \text{if } f_i = f_j \\ -1, & \text{if } (f_i, f_j) \in E \\ 0, & \text{otherwise.} \end{cases} \qquad k = 0, 1, 2. \tag{9}$$

Without loss of generality, by setting the trade-off parameters $\{\mu_k\}$ and λ to 1, and η to 2 in Eq. (3), the quadratic energy functional $E(\mathbf{X})$ can be rewritten in a general matrix form:

$$E(\mathbf{X}) = \mathbf{X^T}(\mathbf{I_p} + \mathbf{L^2} + \mathbf{2S^2})\mathbf{X} - 2\mathbf{X^T}\mathbf{B} + C, \tag{10}$$

where, $\mathbf{I_p}$ is a diagonal matrix such that $I_p(f_i, f_i) = 1$, $f_i \in \bigcup_{k=1}^{K}\{C_k\}$, and zero, otherwise, $\mathbf{B}$ is a matrix whose $\mathbf{B(i,:)} = \mathbf{x_{C_k}}$, $i \in C_k$, and zero, otherwise, and C is a constant matrix. Note that, $\mathbf{I_p} + \mathbf{L^2} + \mathbf{2S^2}$ is symmetric and positive definite. Therefore, minimizing $E(\mathbf{X})$ could obtain a unique solution, that is, solving the following linear system [12]

$$(\mathbf{I_p} + \mathbf{L^2} + \mathbf{2S^2})\mathbf{X} = \mathbf{B}. \tag{11}$$

Eq. (11) can be efficiently solved in many ways, such as the Cholmod [5] or MATLAB linear solvers. After we obtain the solution $\mathbf{X}$, Eq. (4) is applied to specify the labels of triangular faces.

4. Implementation and results

In this section, we will show some experimental results to demonstrate the applicability and flexibility of our algorithm. Most 3D shapes used in this paper are selected from the Princeton Segmentation Bencthmark (PSB) [4] data set and [7]. It includes 19 object categories, such as ant, cup, and glasses, with different poses or shapes. All experiments are implemented on a PC with an Intel Core $i7 - 4790K$. The average triangular face number of 3D shapes selected in this paper is 21583. For most meshes, the segmentation results of our algorithm are acceptable.

4.1. Experimental parameters

In addition to the parameter $\eta = 2$, other parameters, such as $\{\mu_k\}$ and λ, are all set to 1 in Eq. (3). After specifying the semantic sketches, our results can be obtained in less than two seconds for tens of thousands of triangles in the mesh.

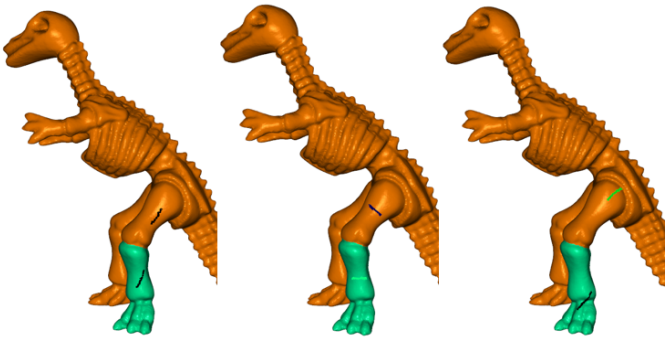


Figure 4: Cutting the shank of dinosaur model with different sketches. It shows that our method is independent of the positions or directions of sketches.

4.2. Foreground and background segmentation

This section illustrates the performance of our algorithm in the mesh foreground and background segmentation. Namely, the user draws two strokes on a mesh to specify the foreground and background, respectively. In this situation, the K in Eq. (3) is equal to 2.

From the Fig. 4, we can see that the shank of dinosaur model is completely cut out by different positions or angles of sketches. Thus users can obtain what they want to get by free paint on the mesh surface. Hence, our method are independent of the positions or directions of sketches.

Although the three Armadillo models in Fig. 5 have different poses, the cutting results are almost the same. Therefore, the proposed algorithm is irrelevant to pose because of the SDF metric and it can understand the semantic boundaries of models.

Figure 6 shows one example of using our method to cut a model with different levels of noise. The noisy models are generated by adding Gaussian noise to the vertices of meshes along the vertex normals [30]. As increasing the noise, our method still can well approximate the cutting boundary. However, we suggest using the guided filter method [30] to denoise the mesh when the noise reaches 70%.

To further evaluate our algorithm, we compare our cutting method with these methods on the data set [7]: the easy mesh cutting (EMC for short) [13], the cross-boundary brushes tool (CBB for short) [31], the iCutter tool (ICC for short) [20], and the paint mesh cutting (PMC for short) [6].

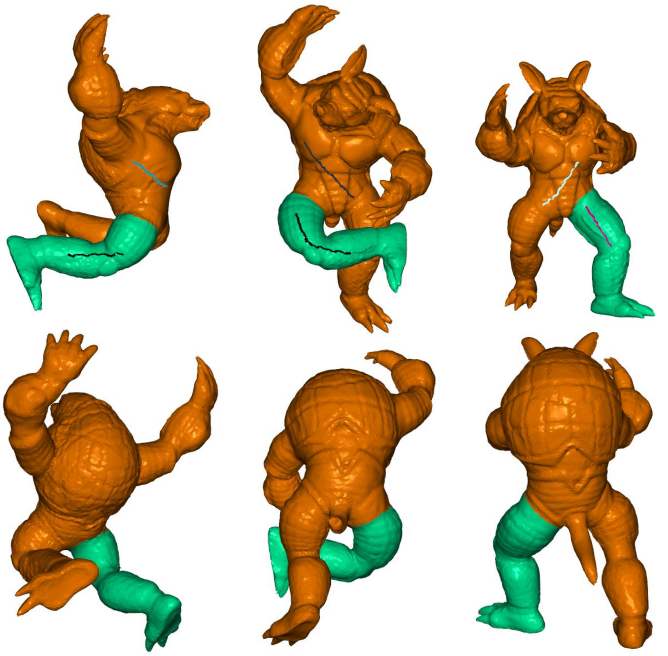
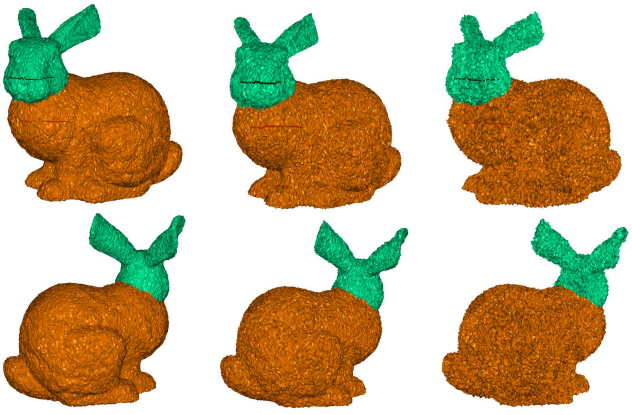


Figure 5: The user can easily cut out the same parts from objects with different poses. The second row is the corresponding back views of the first row with small angle change.

Method	EMC	CBB	ICC	PMC	Ours
BIJ	0.896	0.891	0.907	0.911	**0.915**

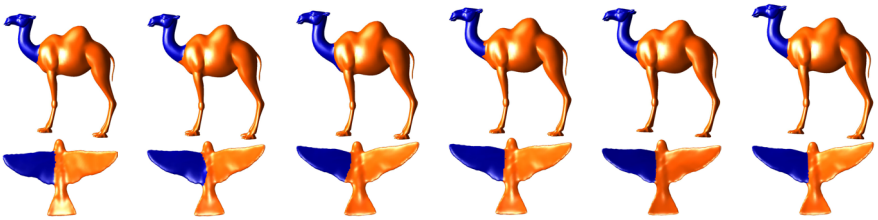
Table 1: Comparison of accuracy for five tools.

Since some methods may sensitive to the locations and shapes of sketches, we generate three cutting results under different positions and shapes of sketches for each algorithm and ask our colleagues to help us select the best one to conduct a fair comparison. A region-based measure is to used to compute the consistency degree between the cutting result produced by the algorithms and the ground truth. The measure is based on binary jaccard index (BJI), which is introduced in [6]. The averaged BJI value is displayed



(a) 10% (b) 30% (c) 50%

Figure 6: The cutting results of a model with different levels of noise. The numbers below the figures show the noisy intensity added the original mesh. The second row is the corresponding back views of the first row.



(a) EMC (b) CBB (c) ICC (d) PMC (e) Ours (f) GT

Figure 7: Comparisons with other methods on the camel and bird meshes. For a fair comparison, we conduct three experiments for each method with the best results exhibiting here. The GT (ground truth) is derived from [7].

in Tab. 4.2 on the dataset [7] and the best metric value is highlighted. Figure 7 shows some segmentation results of different sketch-based methods. The performance of other methods depends on the complex geometrical features and mathematical methodology. From this figure and table, we can see that the segmentation results generated with our algorithm are the most close to the ground truth.

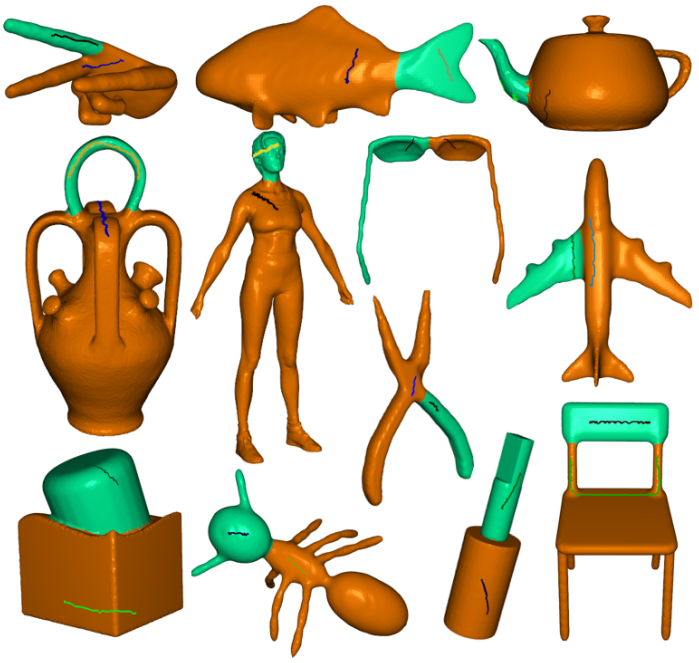
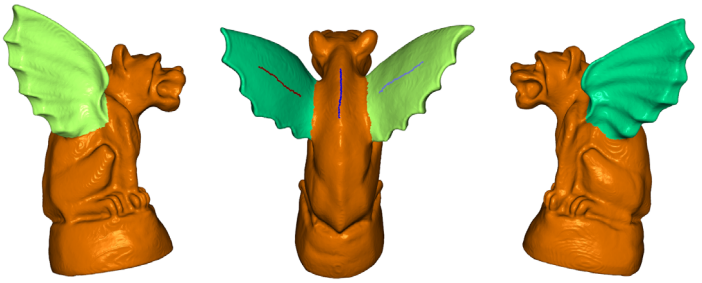


Figure 8: More foreground and background segmentation results on the data set [7] with interactive sketches.



(a) (b) (c)

Figure 9: (b) Results with interactive sketches. (a) and (c) are side views. The wings are completely cut out.

Some other foreground and background segmentation results are shown in Fig. 8. They further demonstrate the performance of our algorithm in foreground and background segmentation.

4.3. Multi-parts segmentation

We conclude this section showing that our model can easily cut a mesh into several parts in one section. Other methods, like PMC [21] and [6], apply iteratively specify the foreground and background sketches on the model to produce multi-parts segmentation results. However, these methods may depend on the order of the sketches. Eq. (3) unifies our segmentation model no matter how many sketches. Multi-parts segmentation only depends on the the situation of $K \geq 3$ and can be solved in a similar linear system.

Figure 9 illustrates the result of cutting the gargoyle model with three interactive sketches. Even though the user draw sketches only from one view, the wings are completely cut out, see the Fig. 9(a,c).

Figure 10 shows the collections of multiple parts segmentation results generated with our algorithm. Each sketch represents a semantic part of a mesh, then using Laplace Coordinates and boundary shrinkage to generate the cutting results where different parts are shown in different colors.

Figure 10: Our model is effectively applied to multiple-part segmentation task with interactive sketches.

5. Conclusions

In this paper, we propose a novel model for cutting out semantic parts from meshes with several simple sketches. It can produce the segmentation results in real time for most test models. Compared with the state-of-the-art methods, our model has the following advantages. It has simple mathematical formulation based on the Laplace Coordinates and gradient. Our model is easy to implement and it can guarantee to have a unique solution. Furthermore, it holds an anisotropic behavior which has a good fitting capability for cutting out the mesh boundaries. Our approach also has some limitations. For example, although the gradient item has some ability about shrinking boundary, it still appears little jaggy on the cutting boundaries. One potential solution is to apply a boundary refinement algorithm like [13, 24] after producing cutting results.

References

[1] G. Antini, S. Berretti, A. Del Bimbo, and P. Pala, *3D mesh partitioning for retrieval by parts applications*, in: Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on, 1210–1213, IEEE (2005).

[2] S. Brown, B. Morse, and W. Barrett, *Interactive part selection for mesh and point models using hierarchical graph-cut partitioning*, Canadian Information Processing Society (2009).

[3] W. Casaca, L. Gustavo Nonato, and G. Taubin, *Laplacian coordinates for seeded image segmentation*, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 384–391 (2014).

[4] X. Chen, A. Golovinskiy, and T. Funkhouser, *A benchmark for 3D mesh segmentation*, in: Acm transactions on graphics (tog), Vol. 28, 73, ACM (2009).

[5] T. A. Davis and W. W. Hager, *Dynamic supernodes in sparse Cholesky update/downdate and triangular solves*, ACM Transactions on Mathematical Software (TOMS) **35** (2009), no. 4, 27.

[6] L. Fan, K. Liu, et al., *Paint mesh cutting*, in Computer graphics forum, Vol. 30, 603–612, Wiley Online Library (2011).

[7] L. Fan, M. Meng, and L. Liu, *Sketch-based mesh cutting: A comparative study*, Graphical Models **74** (2012), no. 6, 292–301.

[8] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin, *Modeling by example*, in: ACM Transactions on Graphics (TOG), Vol. 23, 652–663, ACM (2004).

[9] R. Gal and D. Cohen-Or, *Salient geometric features for partial shape matching and similarity*, ACM Transactions on Graphics (TOG) **25** (2006), no. 1, 130–150.

[10] L. Grady, *Random walks for image segmentation*, IEEE transactions on pattern analysis and machine intelligence **28** (2006), no. 11, 1768–1783.

[11] K. Guo, D. Zou, and X. Chen, *3d mesh labeling via deep convolutional neural networks*, ACM Transactions on Graphics (TOG) **35** (2015), no. 1, 3.

[12] V. Jankovic, *Quadratic functions in several variables*, The Teaching of Mathematics **8** (2005), no. 53-60, 4.

[13] Z. Ji, L. Liu, Z. Chen, and G. Wang, *Easy mesh cutting*, in: Computer Graphics Forum, Vol. 25, 283–291, Wiley Online Library (2006).

[14] S. Katz and A. Tal, *Hierarchical mesh decomposition using fuzzy clustering and cuts*, Vol. 22, ACM (2003).

[15] A. Koschan, *Perception-based 3D triangle mesh segmentation using fast marching watersheds*, in: Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, Vol. 2, II–II, IEEE (2003).

[16] Y.-K. Lai, S.-M. Hu, R. R. Martin, and P. L. Rosin, *Fast mesh segmentation using random walks*, in: Proceedings of the 2008 ACM symposium on Solid and physical modeling, 183–191, ACM (2008).

[17] Y. Lee, S. Lee, A. Shamir, D. Cohen-Or, and H.-P. Seidel, *Mesh scissoring with minima rule and part salience*, Computer Aided Geometric Design **22** (2005), no. 5, 444–465.

[18] B. Lévy, S. Petitjean, N. Ray, and J. Maillot, *Least squares conformal maps for automatic texture atlas generation*, in: Acm transactions on graphics (tog), Vol. 21, 362–371, ACM (2002).

[19] M. Meng, L. Fan, and L. Liu, *A comparative evaluation of foreground/background sketch-based mesh segmentation algorithms*, Computers & Graphics **35** (2011), no. 3, 650–660.

[20] M. Meng, L. Fan, and L. Liu, *iCutter: a direct cut-out tool for 3D shapes*, Computer Animation and Virtual Worlds **22** (2011), no. 4, 335–342.

[21] M. Meng, Z. Ji, and L. Liu, *Interactive mesh segmentation based on feature preserving harmonic field*, (2008).

[22] L. Shapira, A. Shamir, and D. Cohen-Or, *Consistent mesh partitioning and skeletonisation using the shape diameter function*, The Visual Computer **24** (2008), no. 4, 249.

[23] O. Sidi, O. van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or, *Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering*, Vol. 30, ACM (2011).

[24] H.-Y. Wu, C. Pan, J. Pan, Q. Yang, and S. Ma, *A sketch-based interactive framework for real-time mesh segmentation*, in: Computer graphics international (2007).

[25] C. Xiao, H. Fu, and C.-L. Tai, *Hierarchical aggregation for efficient shape extraction*, The Visual Computer **25** (2009), no. 3, 267–278.

[26] Z. Xie, K. Xu, L. Liu, and Y. Xiong, *3d shape segmentation and labeling via extreme learning machine*, in: Computer graphics forum, Vol. 33, 85–95, Wiley Online Library (2014).

[27] K. Xu, H. Zhang, D. Cohen-Or, and Y. Xiong, *Dynamic harmonic fields for surface processing*, Computers & Graphics **33** (2009), no. 3, 391–398.

[28] J. Zhang, C. Wu, J. Cai, J. Zheng, and X.-c. Tai, *Mesh snapping: Robust interactive mesh cutting using fast geodesic curvature flow*, in: Computer graphics forum, Vol. 29, 517–526, Wiley Online Library (2010).

[29] J. Zhang, J. Zheng, and J. Cai, *A diffusion approach to seeded image segmentation*, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2125–2132, IEEE (2010).

[30] W. Zhang, B. Deng, J. Zhang, S. Bouaziz, and L. Liu, *Guided mesh normal filtering*, in: Computer Graphics Forum, Vol. 34, 23–34, Wiley Online Library (2015).

[31] Y. Zheng and C.-L. Tai, *Mesh decomposition with cross-boundary brushes*, in: Computer Graphics Forum, Vol. 29, 527–535, Wiley Online Library (2010).

[32] Y. Zheng, C.-L. Tai, and O. K.-C. Au, *Dot scissor: a single-click interface for mesh segmentation*, IEEE transactions on visualization and computer graphics **18** (2012), no. 8, 1304–1312.

School of Mathematical Sciences, Dalian University of Technology
Dalian City, Liaoning Province, China
E-mail address: lg_liubin@mail.dlut.edu.cn

School of Mathematical Sciences, Dalian University of Technology
Dalian City, Liaoning Province, China
E-mail address: wwmdlut@gmail.com

School of Mathematical Sciences, Dalian University of Technology
Dalian City, Liaoning Province, China
E-mail address: jjcao@dlut.edu.cn

School of Mathematical Sciences
Dalian University of Technology
Dalian City, Liaoning Province, China
E-mail address: xpliu@dlut.edu.cn