Joint Labelling and Segmentation for 3D Scanned Human Body

Hanqing Wang, Changyang Li, Zikai Gao, Wei Liang*
School of Computer Science & Technology, Beijing Institute of Technology

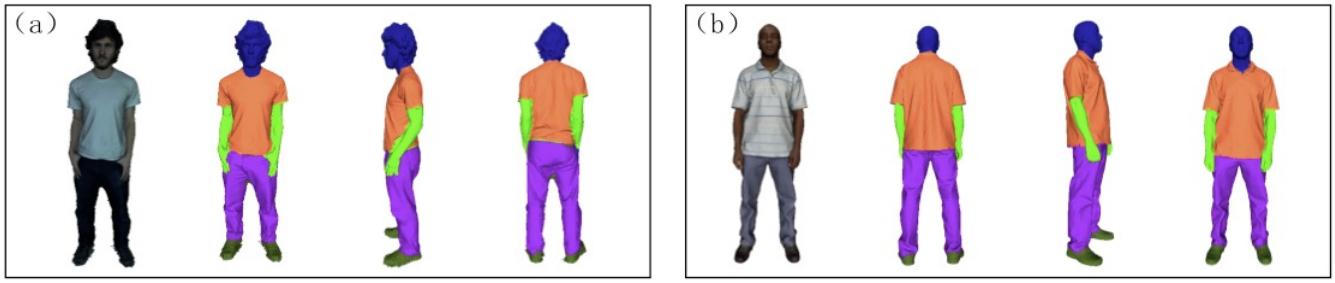

Figure 1: *Two examples of our approach results. The left in (a) and (b) is the input of our method. The right three in (a) and (b) are the labelling and segmentation results from three views, in which different colors depict different human parts.*

Abstract

In this paper, we present an approach to perform 3D human body labelling and segmentation jointly. Given a 3D mesh of scanned human body with texture, our approach segments it into 5 parts: head, torso, arms, legs and feet automatically. We assume that the faces on the same part of human body share similar color features and are constrained by geometry. According to this assumption, we formulate the labelling and segmentation of 3D Mesh as an energy function optimization problem. In this energy function, a data term models the color information and a smooth term models the geometry constraint. Then a GraphCut algorithm is applied to solve the optimization problem. The experiment results show good performance of our method.

Keywords: human body segmentation, 3D mesh segmentation, 3D labelling, GraphCut

Concepts: •**Theory of computation** → *Algorithmic mechanism design;* Computational geometry; •**Applied computing** → **Fine arts;**

1 Introduction and Motivations

Labelling and segmentation for 3D human body mesh into functional parts is a fundamental problem in computer vision, computer graphics and virtual reality community. Accurate segmentation and labelling will help a lot of tasks, such as pose estimation, events understanding, skeleton extraction, character rigging and so on. For human body, people have more identical knowledge about function parts than general objects. However, the large variation of appearance makes segmentation task challenging.

*liangwei@bit.edu.cn

SIGGRAPH 2016 Posters, July 24-28, 2016, Anaheim, CA
ISBN: 978-1-4503-ABCD-E/16/07
DOI: http://doi.acm.org/10.1145/9999997.9999999

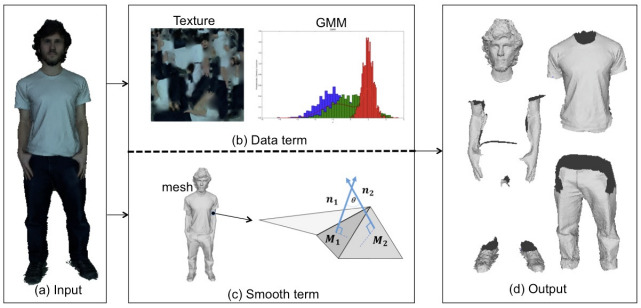

Figure 2: *The framework of our approach. Given a scanned 3D mesh human body (a), our approach interprets each vertex as one of human parts (d). We construct an energy function, in which the data term (b) and the smooth term (c) model color and geometry constraints respectively.*

In this paper, we propose a method to segment 3D scanned human body mesh. We define five functional parts for human body, which are head, torso, arms, legs and feet. Given a 3D mesh of human body with its texture, our approach aims to interpret each vertex as one of the five parts automatically. Fig. 1 demonstrates the results of our method. In Fig.1 (a) and (b), the first column is the 3D mesh input. The right three columns are the labelling and segmentation results from different views, in which different colors depict different parts of human body.

In Fig. 2, we show the framework of our method. Scanned 3D human body mesh is the input (Fig. 2 (a)). A prior of color distribution for each human body part is learned. In most segmentation methods, interactions with images are necessary to provide a priori, like scribbling to mark foreground and background. Different from those methods, our algorithm learns a priori for each part by aligning the input 3D mesh with a labelled T-pose human mesh by Chen's approach [Chen and Koltun 2015]. An energy function is constructed with data term and smooth term, which model color distribution (Fig. 2 (b)) and geometry constraints (Fig. 2 (c)) respectively. A GraphCut algorithm is applied to minimize the energy function to get the labelling and segmentation result for the 3D human body simultaneously, in Fig. 2 (d).

There are three main contributions in this paper:

i) We propose a method to label and segment 3D human body mesh jointly. Each vertex and face in the mesh are interpreted as one of human parts.

ii) We introduce a priori for 3D human body, which enables our method to perform labelling and segmentation automatically without human interactions.

iii) We construct an energy function to model color and geometry constraints for 3D human body. A GraphCut algorithm is used to solve the optimization problem.

2 Related Work

There are two streams in mesh segmentation study: automatic method and interactive method. A branch of mesh segmentation methods is to segment automatically [Attene et al. 2006; Golovinskiy and Funkhouser 2008; Katz and Tal 2003; Lavoué and Wolf 2008]. Most state-of-the-art methods of mesh segmentation is based on iterative clustering. Shlafman et al. [Shlafman et al. 2002] used k-means clustering to segment the models into meaningful pieces. Later, Katz [Katz and Tal 2003] improved the work of Shlafman et al. [Shlafman et al. 2002] by using fuzzy clustering and minimal boundary cuts to achieve smoother boundaries between clusters. Top-down hierarchical segmentation methods have also been used to segment objects with a natural hierarchy of features. Lai et al. [Lai et al. 2006] combined integral and statistical quantities derived from local face characteristics to produce more meaningful results on meshes with noise or repeated patterns. It's difficult for this method to handle large models directly, because it is time consuming to compute pairwise distances. Spectral clustering [Liu and Zhang 2004] could generate good results, but it also suffered from performance problems. To handle large models e.g. one model having more than 10,000 faces, mesh simplification [Katz and Tal 2003; Liu and Zhang 2004] or remeshing [Lai et al. 2006] strategy was used.

Another branch of mesh segmentation methods is to segment by interactive operations [Wong et al. 1998; Zöckler et al. 2000]. In these methods, the user specified some points on the cutting boundary and the cut is completed by finding the shortest path connecting the points. Some research used foreground/background snapping tools [Brown et al. 2009; Zhang et al. 2010; Ji et al. 2006; Wu et al. 2007] to get the priors for foreground and background. Initial seeds were specified by drawing free strokes on the mesh to specify the foreground/background regions, then a graph cut[Brown et al. 2009; Zhang et al. 2010] or region growing [Ji et al. 2006; Wu et al. 2007] algorithm is applied to partition the mesh into two regions.

In recent literature, there is a growing interest in the methods of Joint image segmentation and recognition, especially in computer vision community. Early works in this area include [He et al. 2004; Konishi and Yuille 2000; Tu et al. 2005]. Golovinskiy et al. [Golovinskiy and Funkhouser 2009] segmented urban range data using a graph cut method, and then applied a learned classifier based on geometric and contextual shape.

3 Method

3.1 Problem Definition and Formulation

We define a human body mesh as a set of faces $M = \{M_1, M_2, \ldots, M_N\}$, where N is the number of the faces. We define five functional parts for human body: head, torso, arms, legs and feet. Each part is depicted by a label L_i, and the label set is $L = \{L_1, L_2 \cdots, L_5\}$. Our goal is to interpret each face $M_m \in M$ in the mesh as one semantic label $L_i \in L$.

Figure 3: *Two examples of probability in color space. From the left to the right, we show the probability of the faces belonging to head, torso, arms, legs and feet respectively. The more the faces are red, the higher the probability of belonging to this label is. It is worth noting that some parts of the mesh share the similar color distribution, like head and arms.*

Assume there is a set S_i, $S_i \subset M$,

$$S_i = \{M_j | M_j \text{ is labelled as } L_i\}, i \in [1,5], \quad (1)$$
$$M = \bigcup_{i \in [1,2,\cdots,5]} S_i.$$

We construct an energy function E to model the cost for each possible partition:

$$\mathrm{E} = \sum_{i=1}^{\|L\|} (\mathrm{E}_D(S_i) + \alpha \mathrm{E}_S(S_i)) \quad (2)$$
$$= \sum_{i=1}^{\|L\|} \sum_{j=1}^{\|S_i\|} \left[\mathrm{E}_D(S_{i,j}) + \alpha \sum_{M_k \in \delta_{S_{i,j}}} \mathrm{E}_S(S_{i,j}, M_k) \right],$$

where E_D is the data term to model the cost of assigning one face to one subset S_i in color space . E_S is the smooth term to model the geometry constraints between the adjacent faces. $\alpha \in [0,1]$ is the weight, which balances the data term and the smooth term. $S_{i,j}$ is the j^{th} element of the set S_i. $\delta_{S_{i,j}}$ defines the neighborhood of face $S_{i,j}$. $\| L \|$ and $\| S_i \|$ are the numbers of all labels L and the set $\{S_i\}$.

3.1.1 Data Term

The data term models the cost of assigning each face to one certain label. We decompose it as:

$$\mathrm{E}_D(S_{i,j}) = -ln(p(L_i|S_{i,j}) + \xi) \quad (3)$$

where $p(L_i|S_{i,j})$ indicates the probability of face $S_{i,j}$ labelled as L_i. ξ is a threshold to avoid zero value in the logarithm function. In our algorithm, $\xi = 10^{-30}$.

In the color space, we assume that each set of faces follows a Gaussian Mixture Model distribution, which is determined by its priori. The Eq. 3 is rewrittten as:

$$p(L_i|S_{i,j}) = \frac{1}{\sqrt{(2\pi)^n|C|}} e^{-\frac{1}{2}(x_{S_{i,j}}-\mu)^T C^{-1}(x_{S_{i,j}}-\mu)} \quad (4)$$

where $x_{S_{i,j}}$ is a three dimension vector denotes the RGB values of face $S_{i,j}$. μ is the mean vector. C is the covariance matrix.

3.1.2 Smooth Term

Smooth term models the geometry constraint of the adjacent faces and the difference between them. It is written as:

$$\mathrm{E}_S(S_{i,j}, M_k) = \begin{cases} 0 & S_{i,j} \text{ and } M_k \text{ have the same label} \\ -ln(\theta_{S_{i,j},M_k}/\pi) & \text{Otherwise} \end{cases} \quad (5)$$

where $\theta_{S_{i,j},M_k}$ is the dihedral angle between the faces $S_{i,j}$ and M_k.

3.2 Optimization

Algorithm 1 is the steps of our approach. Firstly, we preprocess the texture and mesh. Every triangle surface of the mesh has three vertices and they can be mapped to three points in texture. Each point in texture can be regarded as a pixel. We use an external rectangle to represent the information of this triangle. Each triangle represented as $T(x_1, y_1, x_2, y_2, x_3, y_3)$ can be replaced by a rectangle. The upper left point is $(min(x_1, x_2, x_3), min(y_1, y_2, y_3))$, the bottom right point is $(max(x_1, x_2, x_3), max(y_1, y_2, y_3))$.We use $R(x_1, y_1, x_2, y_2)$ to represent a rectangle whose upper left point is (x_1, y_1) and the bottom right point is (x_2, y_2). We use $X_{i,j}$ to store the sum value of $R(0, 0, x_i, y_j)$. Thus we get the sum of $R(x_1, y_1, x_2, y_2)$.

$$Sum(x_1, y_1, x_2, y_2) = X_{x_2,y_2} - X_{x_2,y_1} - X_{x_1,y_2} + X_{x_1,y_1} \quad (6)$$

To get the mean value, the sum needs to divide the quantity of pixels in this area.

Then we use the prior information to train a GMM model. This information is some segments of certain part like head, torso or other parts. We use Chens [Chen and Koltun 2015] approach as a priori. Non-rigid registration is for the unclothed human body, and the error of registration is about 10 centimeters, its not accurate enough for segmentation but its enough to give the approximate positions of body parts. This can replace manual scribbles.

Secondly, we use the topological relation to build the graph for formula 2. The weight of every edge between vertices is determined by the data term and smooth term. Every triangle surface in the mesh is a vertex in this graph. We use GMM to evaluate the possibility of a surface belong to a certain part. The angle between two surfaces reflects the difference between two surfaces. After this, we run max-flow-min-cost to solve the graph and get the result.

Thirdly, we use the result obtained in second step to retrain the GMM model, Actually, every time we get the result, we will retrain the GMM model, rebuild the graph and optimize it until the result is steady at last. The following is the pseudo code of this algorithm.

Algorithm 1 3D human body mesh segmentation

Input: M surfaces of huaman body mesh, $T_{n\times m}$ texture, L set of labels, P the initial sets of each part
Output: O Five sets of human body mesh

for $i = 1 : n$ **do**
  for $j = 1 : m$ **do**
    $X[i, j] = X[i, j-1] + X[i-1, j] - X[i-1, j-1] + T[i, j]$;
  end for
end for
for all $M_i \in M$ **do**
  Calculate RGB mean for M_i;
end for
repeat
  for all $L_i \in L$ **do**
    Train GMM model for label L_i using P_i ;
  end for
  for all $M_i \in M$ **do**
    Add weight to the edges connected to this node;
  end for
  Run GraphCut, Update P;
until The result is stable.
$O \leftarrow P$
Output O

4 Experiments and Results

4.1 Dataset

Many approaches of 3D mesh segmentation based on the geography features, like the Princeton Segmentation Benchmark [Chen et al. 2009], which provides a data set of meshes in 19 categories without textures. Whereas our approach takes color information into consideration. Therefore We collected 300 scanned human body meshes to evaluate our method. To speed up our method, we simplify these meshes and limit the number of vertices to 30,000. Then We segment the meshes into five parts manually like head, torso, arms, legs and feet. The manual segmentation is used to evaluate our method as ground truth.

4.2 Labelling and Segmentation

For each mesh, we firstly give an initial mark as the prior information to build the data term. The initial mark is obtained by the Robust Non-rigid Registration [Chen and Koltun 2015]. We use a labelled T-pose mesh and a raw scanned human body as the input of their approach to get the rough map of different parts from T-pose to raw mesh. Next step is iteration. Every time we iterate, we get a new data term. The iteration stops when the energy is stable or reaches the maximum iteration times.

As we test our algorithm, most iterations less than 12. Every iteration spends about 5 seconds where the optimization graph has 30 thousands nodes and 300 thousands edges and the whole expense of time is about 200 seconds. In Fig. 4 , this case iterates eight times. We can learn from Fig. 4 that the very first of iterations have some faces labelled incorrectly. After several iterations, the incorrect labels are corrected.

In experiments, we notice that global energy is not monotonously decreasing. In Fig. 5, we can find that the energy may flux at first. We record results of every iteration and find that it not just happens at first, when the faces are over marked, the energy may rise or even not converge.

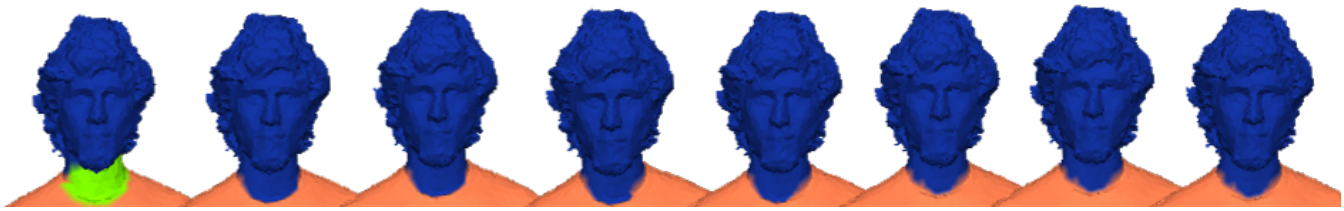

Figure 4: *An example of iteration. The process is shown from the left to the right. The first column is the first iteration and the last column is the eighth iteration. The zoom-in image on the top showing the details of the neck. The boundary becomes more reasonable with the iterations.*

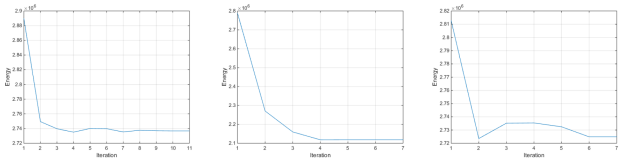

Figure 5: *Global energy in three iterations.*

4.3 Comparisons and Analysis

We take part of the metrics from the Princeton Segmentation benchmark [Chen et al. 2009] to evaluate our method. For each test, we compute the *Hamming Distance* to evaluate the difference between our results and the manual segmentation. We find that the average accuracy of our results is about 0.904.

Table 1: *Accuracy of labelling.*

	Wrong labelled	Total faces	Correct rate
1	2298	20092	0.896
2	2711	20018	0.865
3	1412	28056	0.950

Table 2 is *Confusion Matrix*. The element in *row* i and *column* j indicate the rate of a face that belongs to part i was labelled incorrectly by j. We find that the accuracy of the parts are almost above 90%. The mislabelling happens in the boundary of two adjacent parts. For example, arms and torso, feet and legs.

Table 2: *Confusion Matrix*

	Head	Torso	Arms	Legs	Feet
Head	98.8%	1.2%	0	0	0
Torso	0.3%	92.3%	6.7%	0.4%	0.3%
Arms	0	7.6%	89.7%	2.7%	0
Legs	0	0.2%	0.7%	96.9%	2.2%
Feet	0.5%	0.3%	0.8%	6.1%	92.4%

In our experiments, we find that not all of the meshes converge after iterations. For example, Fig. 6 shows a failure case. In the first iteration Fig. 6 (a), the GMM trained by the initial information segments the mesh correctly. In the second iteration Fig. 6 (b), the result of last iteration formed a new GMM. However, the model estimates the mesh inaccurately. When the GMM is retrained, the chair has bad influence because it covers about 30 percent faces of this mesh and they are useless.

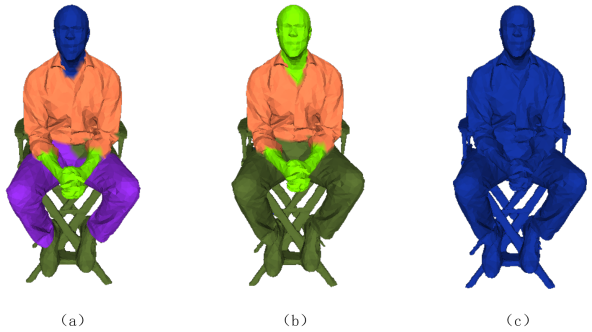

Figure 6: *The iterations of a mesh. (a) The 1st iteration. (b) The 2nd iteration. (c) The 3rd iteration. The mesh is labelled by 'head'.*

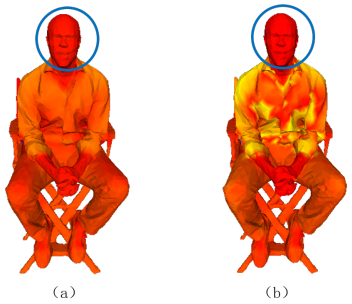

Figure 7: *The isothermal diagram of a case. (a) is the probability of a face belong to head. (b) is the probability of a face belong to hand. We find that hand and head share a similar probability.*

In Fig. 7, the probability of the faces belong to "head" (a) and "hands" (b) is shown. Geography information of head is not such complex like hands, so it has fewer faces than hands. That means head gets less weight in GMM that trained in the second iteration Fig. 6 (b). So the faces of the head were mislabeled by "hand".

If the initial mark is not good enough or it has some other objects, the iteration doesn't converge and the whole mesh is labelled as one part. The data term influences the result a lot. We think this happened because the color feature weighs too much.

5 Conclusion

In this paper, we label and segment 3D scanned human body mesh jointly. Our approach segments the mesh into 5 functional parts: head, torso, arms, legs, and feet automatically. We model color features and geometry constraints in an energy function. Then a GraphCut algorithm is applied to solve the optimization problem of

this function. Our segmentation method bases on color features and geometry features. It can be generalized to other objects easily.

We observe some failure cases. They happened when the prior information is poor or the color feature of different parts is similar. We may introduce more constraints, like the inherent features of human body, to limit the influence of color in the future.

In the experiments, we also observed the algorithm may not converge if the input mesh is over simplified. The data term of the energy function needs enough faces to learn color features. For example, in Fig. 6, the "foot" has 211 faces, which are not sufficient for training. The probability model will be adapted for this situation in the future.

References

ATTENE, M., FALCIDIENO, B., AND SPAGNUOLO, M. 2006. Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer 22*, 3, 181–193.

BROWN, S., MORSE, B., AND BARRETT, W. 2009. *Interactive part selection for mesh and point models using hierarchical graph-cut partitioning*. Canadian Information Processing Society.

CHEN, Q., AND KOLTUN, V. 2015. Robust nonrigid registration by convex optimization. In *Proceedings of the IEEE International Conference on Computer Vision*, 2039–2047.

CHEN, X., GOLOVINSKIY, A., AND FUNKHOUSER, T. 2009. A benchmark for 3d mesh segmentation. In *ACM Transactions on Graphics (TOG)*, vol. 28, ACM, 73.

GOLOVINSKIY, A., AND FUNKHOUSER, T. 2008. Randomized cuts for 3d mesh analysis. *ACM transactions on graphics (TOG) 27*, 5, 145.

GOLOVINSKIY, A., AND FUNKHOUSER, T. 2009. Consistent segmentation of 3d models. *Computers & Graphics 33*, 3, 262–269.

HE, X., ZEMEL, R. S., AND CARREIRA-PERPIÑÁN, M. Á. 2004. Multiscale conditional random fields for image labeling. In *Computer vision and pattern recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE computer society conference on*, vol. 2, IEEE, II–695.

HERDTWECK, C., AND CURIO, C. 2013. Monocular car viewpoint estimation with circular regression forests. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, IEEE, 403–410.

JI, Z., LIU, L., CHEN, Z., AND WANG, G. 2006. Easy mesh cutting. In *Computer Graphics Forum*, vol. 25, Wiley Online Library, 283–291.

KATZ, S., AND TAL, A. 2003. *Hierarchical mesh decomposition using fuzzy clustering and cuts*, vol. 22. ACM.

KONISHI, S., AND YUILLE, A. L. 2000. Statistical cues for domain specific image segmentation with performance analysis. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 1, IEEE, 125–132.

LAI, Y.-K., ZHOU, Q.-Y., HU, S.-M., AND MARTIN, R. R. 2006. Feature sensitive mesh segmentation. In *Proceedings of the 2006 ACM symposium on Solid and physical modeling*, ACM, 17–25.

LAI, Y.-K., HU, S.-M., MARTIN, R. R., AND ROSIN, P. L. 2008. Fast mesh segmentation using random walks. In *Proceedings of the 2008 ACM symposium on Solid and physical modeling*, ACM, 183–191.

LAVOUÉ, G., AND WOLF, C. 2008. Markov random fields for improving 3d mesh analysis and segmentation. In *3DOR*, 25–32.

LIU, R., AND ZHANG, H. 2004. Segmentation of 3d meshes through spectral clustering. In *Computer Graphics and Applications, 2004. PG 2004. Proceedings. 12th Pacific Conference on*, IEEE, 298–305.

SHLAFMAN, S., TAL, A., AND KATZ, S. 2002. Metamorphosis of polyhedral surfaces using decomposition. In *Computer Graphics Forum*, vol. 21, Wiley Online Library, 219–228.

TU, Z., CHEN, X., YUILLE, A. L., AND ZHU, S.-C. 2005. Image parsing: Unifying segmentation, detection, and recognition. *International Journal of computer vision 63*, 2, 113–140.

WONG, K. C.-H., SIU, T. Y.-H., HENG, P.-A., AND SUN, H. 1998. Interactive volume cutting. In *Graphics Interface*, vol. 98, Citeseer, 107–113.

WU, H.-Y., PAN, C., PAN, J., YANG, Q., AND MA, S. 2007. A sketch-based interactive framework for real-time mesh segmentation. In *Computer graphics international*.

ZHANG, J., WU, C., CAI, J., ZHENG, J., AND TAI, X.-C. 2010. Mesh snapping: Robust interactive mesh cutting using fast geodesic curvature flow. In *Computer Graphics Forum*, vol. 29, Wiley Online Library, 517–526.

ZÖCKLER, M., STALLING, D., AND HEGE, H.-C. 2000. Fast and intuitive generation of geometric shape transitions. *The Visual Computer 16*, 5, 241–253.