Ryutarou Ohbuchi, Keisuke Banba, Supervised, geometry-aware segmentation of 3D mesh models, accepted, ICME2012 Workshop on Human-Focused Communications in the 3D Continuum, July 9, 2012, Melbourne Australia

SUPERVISED, GEOMETRY-AWARE SEGMENTATION OF 3D MESH MODELS

Keisuke Bamba
University of Yamanashi
Kofu, Yamanashi, Japan
g09mk029AT_yamanashi.ac.jp

Ryutarou Ohbuchi
University of Yamanashi
Kofu, Yamanashi, Japan
ohbuchiAT_yamanashi.ac.jp

Abstract—**Segmentation of 3D model models has applications, e.g., in mesh editing and 3D model retrieval. Unsupervised, automatic segmentation of 3D models can be useful. However, some applications require user-guided, interactive segmentation that captures user intention. This paper presents a supervised, local-geometry aware segmentation algorithm for 3D mesh models. The algorithm segments manifold meshes based on interactive guidance from users. The method casts user-guided mesh segmentation as a semi-supervised learning problem that propagates segmentation labels given to a subset of faces to the unlabeled faces of a 3D model. The proposed algorithm employs Zhou's Manifold Ranking [18] algorithm, which takes both local and global consistency in high-dimensional feature space for the label propagation. Evaluation using a 3D model segmentation benchmark dataset has shown that the method is effective, although achieving interactivity for a large and complex mesh requires some work.**

Index Terms— Geometric modeling, manifold ranking, diffusion distance, computer graphics.

I. INTRODUCTION

Segmentation of a (manifold) 3D mesh model partitions a polygonal mesh surface into different regions of coherent properties based on local geometrical feature (e.g., "smoothness"), semantics associated with local geometry (e.g., "arm", "leg"), and other properties (e.g., color). Such 3D model segmentation has applications in such areas as mesh transmission [7] and part-based 3D model retrieval [2]. Majority of existing 3D mesh segmentation algorithms are of off-line, fully-automatic kind [10, 8, 4, 15, 16, 3, 9, 12, 11]. Recently, interactive mesh segmentation algorithms have been gaining attention [15, 5, 17]. While fully automatic segmentation algorithms can be useful, interactive segmentation is quite important for some applications. For example, a user may need to segment a part of a 3D model to be a query for a 3D model retrieval system. In such a case, the segmentation may be quite intentional and specific. Automatic segmentation probably won't be enough for such an application.

This paper proposes an interactive, geometry aware segmentation algorithm. A set of faces, or a region, on a 3D mesh model a user considers to be similar are interactively labeled by a set of brush strokes. (See Figure 1.) Then, the label is propagated to unlabeled faces by using a semi-supervised learning algorithm based on Manifold Ranking (MR) [18], which takes into account both local and global changes of local geometrical features.

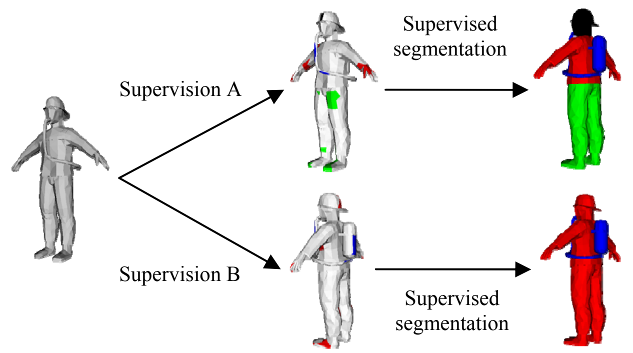


Figure 1. Supervised segmentation of 3D models is necessary as user intention to segment a 3D model may differ by user, by purpose of segmentation, etc.

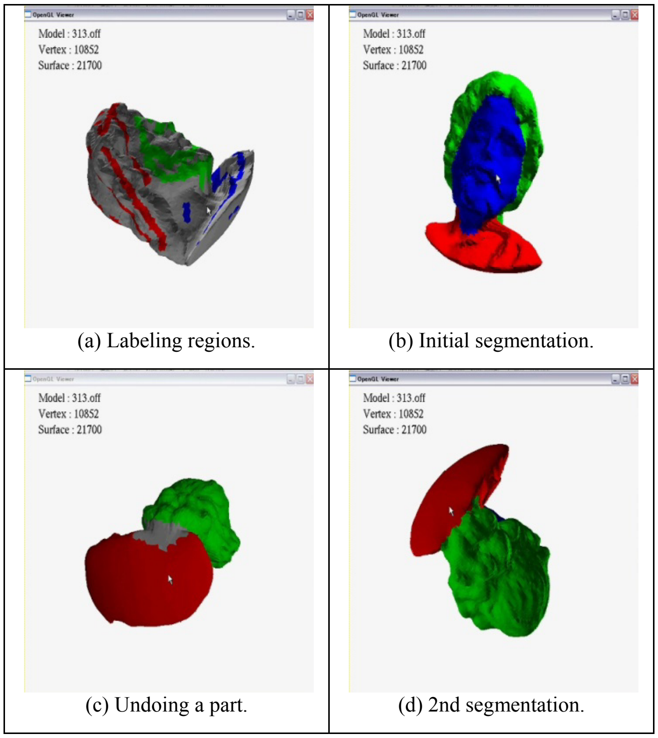
II. RELATED WORK

While there have been many proposal on mesh segmentation algorithms, majority of existing 3D mesh segmentation algorithms are of off-line, fully-automatic kind [10, 8, 4, 15, 16, 3, 9, 12, 11]. Automatic segmentation results, however, may not be what a user wants. While off-line learning from a corpus, e.g., [8], helps to incorporate human perception collectively, such an approach won't be able to reflect one's intention. Thus, depending on application scenarios, interactive segmentation algorithms such as [15, 5, 17] are quite important.

Figure 2 shows an example of supervised segmentation by using the proposed algorithm. The mesh is marked by three colors that are the three labels of segmentation regions (a). The system segments the model by propagating the labels (b). If initial segmentation is not what the user wants, part of the segmentation may be undone (c) (gray part in the image) to redo labeling for a new segmentation (d) with an improved result. (Color to region correspondence changed from labeling (a) to segmentation results (b~d) due to an implementation error.)

Interactive segmentation methods can be classified by various criteria [12]. One criterion is locality of the algorithm. Local approaches, e.g., [11], are fast, but results

Ryutarou Ohbuchi, Keisuke Banba, Supervised, geometry-aware segmentation of 3D mesh models, accepted, ICME2012 Workshop on Human-Focused Communications in the 3D Continuum, July 9, 2012, Melbourne Australia



(a) Labeling regions. (b) Initial segmentation. (c) Undoing a part. (d) 2nd segmentation.

Figure 2. Interactive and iterative segmentation using the proposed algorithm. Undesirable part may be undoed to be labeled and segmented again. (Color to region mapping changed from initial labeling (a) to segmentation results (b~d) due to an implementation error.)

may not be as good as desired, for they only consider local properties. Local search in feature space often get stuck in local minima/maxima.

Inspired by semi-supervised image labeling [13], we cast user-guided mesh segmentation as a semi-supervised learning problem that propagates segmentation labels given to a subset of faces to the unlabeled faces of a 3D model. As with [13], our algorithm tries to overcome the issue of local maxima/minima by employing a powerful diffusion-based learning algorithm called Manifold Ranking [18]. While our algorithm uses local geometrical feature, determination of segmentation regions takes both local and global changes of the geometrical features into account.

III. ALGORITHM

Proposed algorithm aims at supervised, geometry-aware segmentation of 3D mesh models. The algorithm assumes singly-connected manifold mesh 3D model as its input. The user labels a subset of the 3D mesh model marked by using an interface similar to 3D painting system. The algorithm first computes, for each face, *n*-dimensional local geometrical feature. The algorithm then propagates labels from marked faces to unlabeled faces over 2-manifold of the 3D mesh (actually, a dual graph of it) embedded in the *n*-dimensional feature space.

The segmentation follows the steps below;

1. **Extract local feature:** Extract *n*-dimensional local feature $\mathbf{h}_i$ (i=1…m) at every face f_i (i=1…m) of the 3D model to be segmented.
2. **Construct face connectivity graph G**: Given a mesh of the 3D model, construct a manifold mesh G representing the connectivity of *faces* f_i of the 3D mesh model. That is, G is a dual graph of the vertex connectivity of the original 3D mesh model. For example, a triangular face f_i of a 3D model has tree adjacent faces. Thus, f_i is a vertex having valence 3 in the mesh G. As mentioned above, f_i is associated with local geometrical feature $\mathbf{h}_i$.
3. **Compute edge weights of G:** For each edge connecting f_i and f_j in graph G, compute edge weight $\mathbf{G}_{i,j}$ based on the similarity of features $\mathbf{h}_i$ and $\mathbf{h}_j$. In the label propagation state (step 5) labels propagate smoothly and quickly through an edge having similar features, that is, a higher "diffusion coefficient". Conversely, if features on an edge is different, label propagation is impeded at the edge.
4. **Interactively seed regions:** Label a subset of faces interactively as a set of seed faces of a region having label l, (l=1…c). There may be c distinct region labels. Labeling is done by using an interface similar to a mesh painting tool.
5. **Propagate labels:** Propagates labels over the mesh G by using *Manifold Ranking* algorithm by Zhou, et al. [18]. If there are c distinct labels, each vertex has c relevance values, each of which indicates likelihood of the vertex having region label l.
6. **Select a label for vertex:** After label propagation step terminates, each vertex i of the graph G is associated with c relevance values corresponding to c region labels. The f_i assumes a label having the highest relevance values among the c labels. Present the segmentation result to user, for example, by visualizing labels by distinct colors of the faces.
7. **Repeat segmentation until satisfactory:** If resulting segmentation is not satisfactory, undo undesired part of the segmentation. Then go back to step 4. Repeat the interactive segmentation process while necessary.

Following subsections explain details of local geometrical feature and manifold ranking algorithm.

A. Per face Local Geometrical Feature

A feature vector $\mathbf{h}_i$ of a face f_i (i=1...m) is concatenation $\mathbf{h}_i$=($\mathbf{x}_i$, $\mathbf{q}_i$) of its 3D Euclidian coordinate $\mathbf{x}_i$ and local geometrical feature $\mathbf{q}_i$ of barycenter of the face f_i. The local geometrical feature, which we call *Local Statistical Feature* or LSF, is a 4D joint histogram of angles and distance between a pair of points [LSF_icme2012].

An LSF for a face is computed from a set of oriented points within a sphere of influence of radius r at centered at barycenter of the face (Figure 4). A surface based model is first sampled by thousands of oriented points placed at quasi-random locations on the faces. Number of samples per face is made so that area density is uniform over the 3D model. Orientation of a point is the surface normal of the face on which the point is generated at. An LSF for a face is computed by using points within a sphere of radius r of the barycenter of the face. Assume that the *vertex of interest* is $\mathbf{p}_1$ and its normal vector is $\mathbf{n}_1$. Assume also that point $\mathbf{p}_2$ and its normal vector $\mathbf{n}_2$ lie inside a sphere of radius r of the vertex $\mathbf{p}_1$. Using $\mathbf{p}_1$ and $\mathbf{p}_2$, a 4-tuple (α, β, γ, δ) of distance δ and angle-related values α, β, γ is computed as follows (See Figure 5);

$$\alpha = \arctan\left(\mathbf{w}\bullet\mathbf{n}_1, \mathbf{u}\bullet\mathbf{n}_2\right). \tag{1}$$

$$\beta = \mathbf{v}\bullet\mathbf{n}_2 \tag{2}$$

$$\gamma = \mathbf{u}\cdot\left(\mathbf{p}_2 - \mathbf{p}_1 / \left\|\mathbf{p}_2 - \mathbf{p}_1\right\|\right). \tag{3}$$

$$\delta = \left\|\mathbf{p}_2 - \mathbf{p}_1\right\|. \tag{4}$$

where $\mathbf{u} = \mathbf{n}_1$, $\mathbf{v} = (\mathbf{p}_2 - \mathbf{p}_1)\times\mathbf{u} / \left\|(\mathbf{p}_2 - \mathbf{p}_1)\times\mathbf{u}\right\|$, $\mathbf{w} = \mathbf{u}\times\mathbf{v}$.

The radius r of the sphere of influence defines locality of LSF; the smaller the radius, the local the feature is. If there are p oriented points within the sphere, (p-1) tuples are computed, and (p-1) each of values δ, α, β, and γ are collected into a 4-dimensional joint histogram to become an LSF feature.

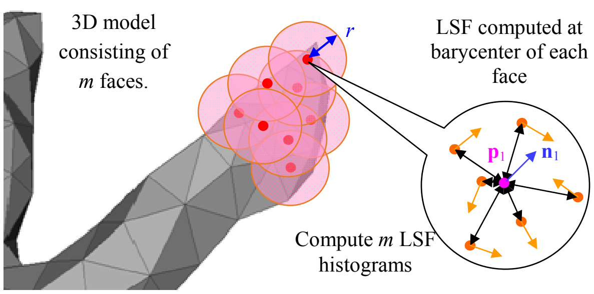


Figure 4. A local geometrical feature LSF is computed for each face from points in a sphere of radius r about the face.

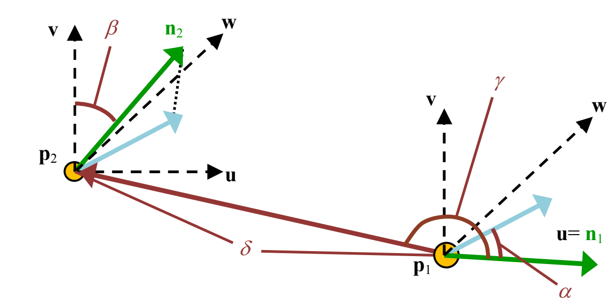


Figure 5. An LSF is a histogram of 4-tuples (α, β, γ, δ) from a pairs of oriented points within radius of influence r.

LSF feature is similar to Surflet Pair Relation Histograms (SPRH) [14] by Wahl, et al., but there are two differences. First, original SPRH is global, while LSF is local. Second, given a set of p oriented points, LSF computes a histogram by accumulating (p-1) tuples between a point of interest and points other than the point of interest that lie inside a sphere of influence. The SPRH, on the other hand, computes a histogram among all the pair of points.

In the experiments below, we compute LSF with 2,048 oriented points per 3D model. We use joint histogram having 3 bins for each value of the 4-tuple. Thus, if flattened to a 1D vector, LSF $\mathbf{q}_i$ is a $3^4 = 81$ dimensional vector. Combined with the coodinate of the face $\mathbf{x}_i$, the feature vector per face $\mathbf{h}_i$ has dimensionality n=84.

B. Label Propagation

Label propagation over G is performed by using *Manifold Ranking* (MR) algorithm by Zhou et al [18] [13] adapted to our purpose. MR is a learning algorithm that learns distribution of feature points in high-dimensional feature space by simulating diffusion of relevance value from labeled, i.e., source vertices to unlabeled vertices over G. In our segmentation algorithm, user-labeled faces in G are the "sources" of label propagation. Assuming there are c distinct region labels, c distinct diffusion processes would take place over the mesh G. Each face is thus associated with c distinct relevance values, each corresponding to likelihood of the face having one of c region labels. Propagation of a label relevance value occurs from a face f_i to the another face f_j across an edge shared by the faces on the original 3D mesh model. On dual graph G of face connectivity, this corresponds to propagation of label relevance value from a vertex f_i to f_j through an edge having weight $\mathbf{G}_{i,j}$.

Original MR [18] forms the mesh G based on similarity of vertices in an ambient high-dimensional feature space, e.g., by k-nearest neighbor connection. Each edge of G is associated with an affinity value that controls diffusion of relevance through the edge. The affinity is computed as the affinity among a pair of features on the edge. In the proposed segmentation algorithm, however, the connectivity of mesh G is *given a priori* as the connectivity of faces of a given 3D mesh.

Let $X = \{x_1, ..., x_l, x_{l+1}, ..., x_m\}$ be the set of vertices in G. Vertices from x_1 to x_l are (manually) labeled "source" vertices, and each one has one of the (initial) segmentation labels in $L = \{1, ..., c\}$. The rest of the vertices in X are the (initially unlabeled) vertices to be labeled by MR.

Define $\mathbf{F} = [F_1, ..., F_m]^T$ as an $m\times c$ matrix whose element $\mathbf{F}_i$ is a ranking score computed for a label at vertex i. Also define $\mathbf{Y} = [Y_1, ..., Y_m]^T$ as an $m\times c$ matrix whose element is $Y_{ij} = 1$ if label of the ith vertex $y_i = j$ $(j \in L)$, and $Y_{ij} = 0$ otherwise.

On each edge of G, diffusion of relevance value indicating the likelihood of a vertex assuming a label is controlled by the affinity $\mathbf{G}_{ij}$ of the edge given as below;

$$\mathbf{G}_{ij} = \begin{cases} \exp\left(-d(\mathbf{h}_i, \mathbf{h}_j)^2 / 2\sigma^2\right) & \textit{if} \quad i \neq j \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

The matrix $\mathbf{G}$ is positive symmetric. Note that $\mathbf{G}_{ii} = 0$ since there is no edge connecting a point with itself. As the distance $d(\mathbf{h}_i, \mathbf{h}_j)$, we chose, based on our preliminary experiments, $L_{0.5}$-norm [1] below;

$$d_{0.5}(\mathbf{h}_i, \mathbf{h}_j) = \sum_{a=1}^{n} \left(\left\| \mathbf{h}_{ia} - \mathbf{h}_{ja} \right\|^{0.5} \right)^{1/0.5} \quad (6)$$

From G, a normalized *graph Laplacian* $\mathbf{S}$ is formed;

$$\mathbf{S} = \mathbf{D}^{-\frac{1}{2}} (\mathbf{D} - \mathbf{G}) \mathbf{D}^{-\frac{1}{2}} \quad (7)$$

where $\mathbf{D}$ is a diagonal matrix in which $\mathbf{D}_{ij}$ equals to the sum of the i-th row of $\mathbf{G}$, that is, $\mathbf{D}_{ij} = \sum_j \mathbf{G}_{ij}$. The ranking vector $\mathbf{F} = \left[f_1, \cdots, f_n \right]^T$ that indicates the likelihood of vertices having a label can then be estimated by iterating the following equation until convergence;

$$\mathbf{F}^{(t+1)} = \alpha \mathbf{S} \mathbf{F}^{(t)} + (\mathbf{I} - a)\mathbf{Y} \quad (8)$$

The parameter $\alpha > 0$ affects convergence of the iteration above. Let $\mathbf{F}^*$ be the limit of the above iteration. This iteration has a closed form solution as follows;

$$\mathbf{F}^* = (1 - \alpha)(\mathbf{I} - \alpha \mathbf{S})^{-1} \mathbf{Y} \quad (9)$$

Given the converged solution $\mathbf{F}^*$, a vertex i would have the label y_i which has the highest relevance rank value.

$$y_i = \arg\max_{j \le c} F_{ij} \quad (10)$$

Computational cost is an issue to be considered if MR is to be used in an interactive segmentation loop. The cost of MR is dominated by the cost of meshing and computing $\mathbf{F}^*$. The cost increases with the size of matrix $\mathbf{S}$, that is, the number of faces m. For example, the matrix $\mathbf{S}$ takes $O(m^2)$ (if it is dense) to store and roughly $O(m^3)$ to perform label propagation via matrix inversion.

C. Patch-Based Segmentation

Due to the high computational cost of manifold ranking, the proposed segmentation algorithm can't handle a large mesh, e.g., a mesh having 10k or more faces, interactively. To alleviate this issue, we extend the segmentation algorithm described above so that the segmentation is *hierarchical*. In the following, this extension is called "hierarchical method", and the non-hierarchical version described in Section 3.2 is called "direct method".

Basic idea is to run the label propagation algorithm described in Section 2.2 on a mesh that consists of patches whose number is much smaller than the original number of faces N_f. Each patch is a set of faces that is topologically adjacent and have similar features.

Patches are formed in a pre-processing step. An input (complex) mesh having N_f faces is partition into a set of N_p patches, in which $N_p \ll N_f$. To partition the original mesh into set of patches, we employ, again, MR based label propagation guided by feature. Instead of manual labels, however, labels are propagated from *randomly placed* seed points on the (original) mesh surface. Feature of a created patch is a component-by-component average of the 4D joint histogram of the faces contained in the patch.

After forming patches, 3D model consisting of N_p patches is segmented in a manner identical to the process described in Section 3.2. Since $N_p \ll N_f$, the cost of manifold ranking in the label-segment interaction loop is much smaller, and the iteration could maintain an interactive response time, e.g., less than a few seconds.

Figure 6 illustrates the hierarchical method. A complex mesh (a) is pre-partitioned into "macro faces" (b by using randomly-placed seed points (red marks). Actually, macro faces in (c) is too large to obtain a reasonable segmentation; macro faces of the size shown in (c) is more reasonable.

The hierarchical segmentation by using partitioning and macro faces do degrade segmentation quality. For example, boundaries of segmentation results are more uneven, for the unit of segmentation is now macro faces.

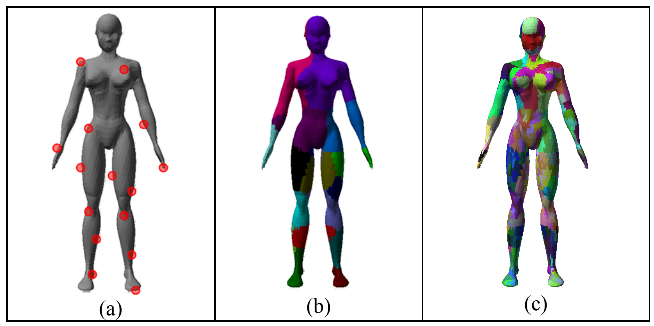


Figure 6. To reduce computational cost in the interactive label-segment loop, segmentation may be performed on a 3D model consisting of "macro faces".

IV. EXPERIMENTS AND RESULTS

We performed experiments to evaluate the proposed method in terms of segmentation quality and computational efficiency.

A. Quality of Direct Method

Quantitative evaluation of segmentation quality for 3D mesh segmentation algorithms is by itself is a research topic [4][6]. Furthermore, to author's knowledge, previously published quantitative evaluation methods are for fully automatic segmentation methods. We thus settled for a less than ideal option of comparing our interactive segmentation method with automatic segmentation methods by using the evaluation procedure and benchmark database by Chen et al [6]. Chen's benchmark contains 380 models, each of which is associated with multiple human-segmented ground truths. Chen et al [6] also defined multiple indices of segmentation quality.

We asked 8 volunteers to interactively segment the 95 models by proposed method with a simple graphical user interface. Given a model to be segmented, each volunteer selected one of ground truth segmentations, and tried to copy the segmentation using our system. Interactive label-segmentation loop might have iterated a few times before the user is satisfied. Following the protocol of the benchmark, quality of the segmentation is quantified by using the "cut discrepancy" measure proposed by Chen, et al [6].

Figure 7 shows the evaluation result for the direct method. Compared to seven methods, all of which are fully automatic, the proposed method has significantly lower cut discrepancy score. Supervised results using out method should be better than the results of fully automatic segmentations. Our proposed method at least achieved this as it produced the lowest cut discrepancy value.

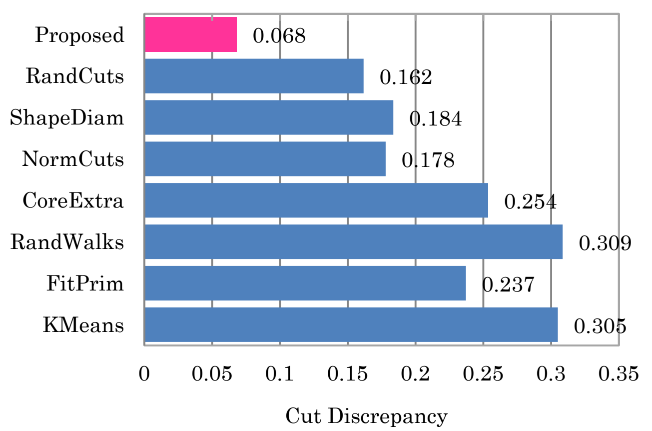


Figure 7. Comparison of segmentation quality with various automatic segmentation methods by using cut discrepancy.

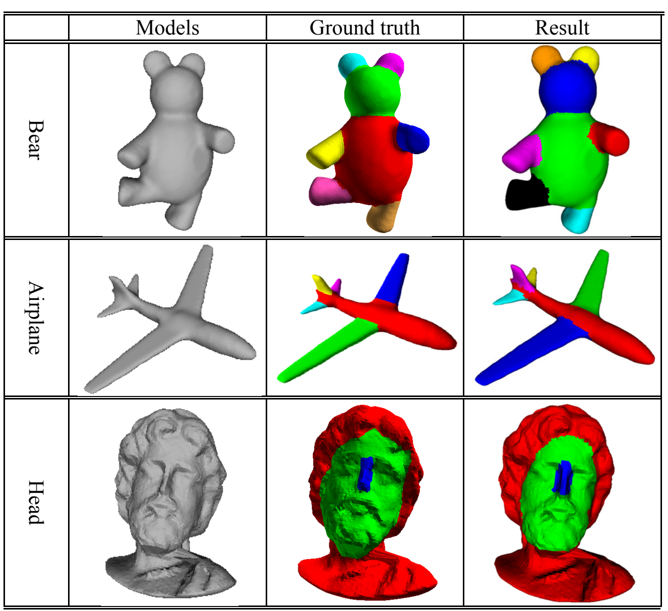


Figure 8. Three models (left), their ground truth (middle), and segmentation results obtained after less than a few iterations (right).

Figure 8 shows several 3D models, their ground truth segmentations, and the segmentation results based on the ground truth segmentation.

B. Quality and Efficiency of Hierarchical Method

We evaluated the hierarchical segmentation method described in Section 6.2 for its quality (qualitatively) and computational efficiency. The evaluation used the same 95 models as in the previous section. We first partitioned the model into parts having N_m=3,000 patches prior to the segmentation. A user then segmented the model so the result resembles one of the ground truths the user selected.

Examples of segmentation results and their execution timings using the hierarchical method are shown in Figure 9, and in Table 1. The "Chair" model has 31,456 faces, while the "Armadillo" model has 50,542 faces. In Table 1, for the direct method, "pre-processing" consists mostly of feature extraction. For hierarchical method, "pre-processing" also include patch formation. For both methods, the "propagate labels" step consisting mostly of manifold ranking is done in the interactive label-segment loop.

As the examples show, segmentations using the hierarchical method have lower quality at jagged the segmentation boundary. However, we observe that the quality of boundary is reasonable considering the model consists of only 3,000 patches. Reasonable quality of segmentation boundary is achieved since quality of patch

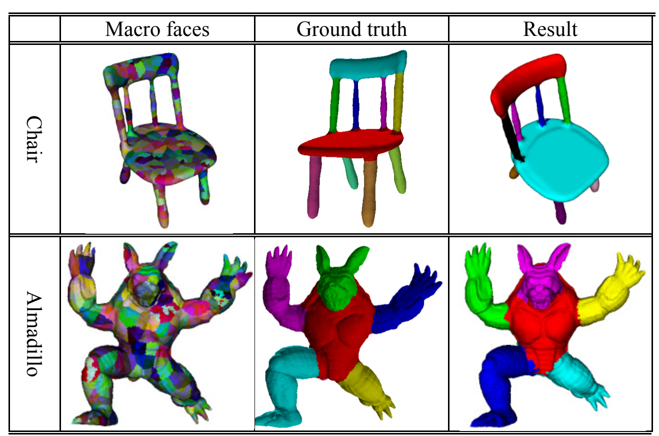


Figure 9. Examples of hierarchical segmentation. The chair model has 31,456 faces, while the armadillo model has 50,542 faces. Number of patches N_p=3,000.

Table 1. Execution time for patch-based hierarchical segmentation method for two example models.

Model	Direct		Hierarchical	
	Pre-process	Propagate label	Pre-process	Propagate label
Chair	183.3s	39.2 s	221.0 s	1.5 s
Almadillo	384.3s	51.9 s	464.8 s	1.3 s

boundaries, which are created the same MR-based label propagation, were good to start with

As for computational cost, time spent on label propagation during an interactive segmentation loop has been reduced from 40~50s down to just over a second. The hierarchical segmentation, on the other hand, spends more time in the pre-processing stage forming patches.

V. DISCUSSION

We have proposed an interactive, supervised, geometry-aware segmentation algorithm for 3D (manifold) mesh models. A user labels a small subset of faces of each segmentation region by using an interface similar to 3D model painting software. Then, the labels are propagated to non-labeled faces over the manifold of the mesh guided by local geometrical features computed for each face. The propagation is performed by using Manifold Ranking algorithm by Zhou et al. [18]. The proposed method does not scale to a complex 3D model, as the label propagation during interactive loop is computationally costly. To reduce the computational cost, we proposed a hierarchical segmentation algorithm.

We evaluated the segmentation quality and computational cost by using a subset of mesh segmentation benchmark [6]. Our algorithm has produced better segmentation quality than all the fully automatic algorithms. However, further study is needed for a better benchmark for interactive mesh segmentation algorithms.

In the future, improvement in patch-based hierarchical segmentation algorithm is necessary so that the segmentation quality would be closer to non-hierarchical ones. An approach similar to the one in Katz et al [9] may be used to obtain smoother boundaries. We are also considering methods to accelerate manifold ranking itself.

ACKNOWLEDGEMENTS

This research has been funded in part by the Ministry of Education, Culture, Sports, Sciences, and Technology of Japan (No. 18300068).

REFERENCES

[1] C.C. Aggarwal, A. Hinneburg, D.A. Keim, "On the surprising behavior of distance metrics in high dimensional space", *Proc. Int'l Conf. on Database Theory*, 420-434, 2001.

[2] G. Antini, G., S. Berretti, P. Pala, "3d mesh partitioning for retrieval by parts application", *Proc. IEEE International Conference on Multimedia & Expo* (*ICME*) 2005, 1210-1213, 2005.

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

[4] H. Benhabiles, J-P. Vandeborne, G. Lavoue, M. Daoudi, "A comparative study of existing metrics for 3D-mesh segmentation evaluation", *Visual Computer*, **26**(12), 2010.

[5] S. Brown, B. Morse, W. Barrett, "Interactive part selection for mesh and point models using hierarchical graph-cut partitioning", In *Proc. GI* '09, 23–30, 2009.

[6] X. Chen, A. Golovinskiy, T. Funkhouser, "A Benchmark for 3D Mesh Segmentation", *ACM Trans. on Graphics*, (Proc. SIGGRAPH 2009), **28**(3), 2009.

[7] W. Guan, J. Cai, J. Zheng, C.W. Chen, View-Based 3D Model Transmission via Mesh Segmentation, *Proc. IEEE Int'l Conf. on Multimedia and Expo* (*ICME*) 2007, 1311-1314, 2007.

[8] E. Kalogerakis, A. Hertzman, K. Singh, "Learning 3D Mesh Segmentation and Labeling", *ACM Trans. on Graphics*, (Proc. SIGGRAPH 2010), **29**(3), (2010).

[9] S. Katz, A. Tal, "Hierarchical Mesh Decomposition Using Fuzzy Clustering and Cuts", *ACM Trans. on Graphics*, (Proc. SIGGRAPH 2003), **22**(3), 2003.

[10] A.P. Mangan, R.T. Whitaker, Partitioning 3D Surface Meshes Using Watershed Segmentation, *IEEE Trans. Visualization and Computer Graphics*, **5**(4), 1999.

[11] D.L. Page, A. Koschan, M.A. Abidi, "Perception-based 3D Triangle Mesh Segmentation using Fast Marching Watersheds", *Proc. Int'l Conf. on Computer Vision and Pattern Recognition*, vol. 2, 27-32, 2003.

[12] A. Shamir, "A formulation of boundary mesh segmentation", *Proc. Int'l. Symp. 3D Data Processing* Visualization and Transmission, 82-89, 2004.

[13] H. Tong, J. He, M. Li, W-Y Ma, H-J Zhang, C. Zhang, "Manifold-Ranking-Based Keyword Propagation for Image Retrieval", *EURASIP J. Adv. Sig. Proc.* 2006.

[14] E. Wahl, U. Hillenbrand, G. Hirzinger, "Surflet-Pair-Relation Histograms - A Statistical 3D-Shape Representation for Rapid Classification", *Proc. 3DIM* 2003, 474-481, 2003.

[15] H-Y. Wu., C. Pan, J. Pan, Q. Yang, S, Ma, A sketch-based interactive framework for real-time mesh segmentation, *Proc. CGI* '07 (2007).

[16] H. Yamauchi, S. Lee, Y. Lee, Y. Ohtake,. A. Belyaev, H.-P. Seidel, "Feature sensitive mesh segmentation with mean shift", *Proc. IEEE Int'l Conf. on Shape Modeling and Applications* 2005, 236-243, 2005.

[17] Y. Zheng, C-L. Tai, "Mesh Decomposition with Cross-Boundary Brushes", *Computer Graphics Forum*, **29**(2), 527-535, (2010).

[18] D. Zhou, O. Bousquet, T. Navin L.J. Weston, B. Schölkopf, "Learning with Local and Global Consistency" *Advances in Neural Information Processing Systems* 16, 321-328, 2004.