Volume Based Mesh Segmentation

TAN-CHI HO AND JUNG-HONG CHUANG
Department of Computer Science
National Chiao Tung University
Hsinchu, 300 Taiwan

Mesh segmentation has become a key ingredient in many mesh applications in computer graphics. In this paper, we propose a hierarchical segmentation that decomposes a polygonal object into meaningful parts in such a way that not only components on a higher level reveal higher degree of salience than their descendant parts but also the components on each level of hierarchy have similar degree of salience. Moreover, the number of boundaries on each level of the hierarchy is determined automatically. The proposed segmentation is based on the Minimum Slice Perimeter (MSP) function [1], which represents non-local shape features and has better interpretation for the object parts. The gradient of MSP function is used to locate the segmentation regions and a new measure of part salience is proposed to evaluate the significance of the segmentation regions. For each level of hierarchy, some most perceptually significant segmentation regions are selected based on their salience measures and boundaries are then computed from the selected segmentation regions by using a capacity that considers both the curvature and MSP gradient.

Keywords: shape analysis, mesh segmentation, minimum slice perimeter, computer graphics, geometric modeling

1. INTRODUCTION

Triangular meshes are fundamental modeling representation for computer graphics applications. The demand of techniques for analysis, processing, transmission, and rendering 3D meshes are increasing in response to the wide-spreading applications, leading to tremendous technical developments for 3D meshes in the last decade. As meshes are becoming larger and more complex, decomposing an object into smaller and simpler components is essential for many mesh techniques, including parameterization, texture mapping, morphing, editing, shape matching, compression, and more. Thus mesh segmentation has become a key ingredient in many mesh applications.

To perform geometric operations on the mesh surface, we often require the help of properties defined on the surface. Most of the existing mesh segmentation methods rely on surface properties that are either local detail geometric features or global topological structure. The use of too local or too global surface properties limits many segmentation algorithms to either decompose a model into several surface patches or be able to handle only models with some specific topological structure such as core-salient features. The object part is an intermediate level structure relative to the entire mesh and it can be analyzed using some intermediate level surface properties. Our segmentation scheme is based on the observation that neighboring regions having similar internal volume tend to be grouped into a part. To describe the internal volume of the object, we use a recently developed surface property called *Minimum Slice Perimeter* (MSP) [1]. The MSP is a

surface function that describes the local internal volume around the surface point. It is derived from the concept of short-cut [2] which tries to find the best representative slice plane for a surface point. And the internal volume around the surface point can then be approximated as the perimeter of the slice. Compared to the *Shape Diameter Function* (SDF) [3] derived from the concept of medial-axis, the short-cut slice in MSP has better interpretation for the local internal volume than the maximum inscribed balls in SDF. For object parts that are highly non-cylindrical, the SDF describes only part of the internal volume since it requires multiple inscribed balls to fill the volume.

Since complex models often contain features in different scales or salience, ranging from global structure to detail surface features, it is useful to decompose the models into components in several levels or hierarchically. Several approaches have been proposed in this direction. Hierarchical face clustering techniques construct the hierarchy in bottom up fashion [4]. Top down approaches start from the root representing the whole object and partition the component into two or more parts [5]. This process continues recursively until a certain condition is met. At each level of the top-down approach, the segmentation is usually derived implicitly by locating the best boundary between parts. Several hierarchical segmentation techniques have been proposed [5-8]. Most of them possess the property that components on a higher level reveal higher degree of salience than their descendant parts. But many of them cannot ensure that the components on each level of hierarchy have similar degree of salience.

Locating boundary between parts can be done by either boundary-based or region-based approach. Boundary-based approaches use local geometric properties, such as curvature, to locate boundary. Region-based approaches seek for regions with similar properties, such as the combination of geodesic and angular distances in [5], and from which boundaries are derived. Since parts have different levels of salience, to evaluate the significance of a boundary between parts we need to include the salience measures of the associated parts. However, the boundary computed by using boundary-based and region-based approach usually lacks for the salience measures of the parts associated with the boundary.

This paper presents a new hierarchical segmentation scheme that decomposes an object into meaningful parts in such a way that not only components on a higher level reveal higher degree of salience than their descendant parts but also the components on each level of hierarchy have similar degree of salience. Our segmentation is based on a recently developed surface function called *Minimum Slice Perimeter* (MSP) which represents the object's internal volume on the surface [1]. As neighboring regions having similar internal volume tend to be grouped together and form a part, the gradient of MSP can be used to locate the candidate segmentation regions that contain the boundaries. Moreover, the significance of segmentation regions is evaluated in the process of hierarchical segmentation. The evaluation takes into account the salience information of the parts associated with the segmentation region.

The proposed hierarchical segmentation scheme starts from the whole object and, for each level of the hierarchy, locates segmentation regions by applying a threshold to the gradient of MSP function, then evaluates the significance of segmentation regions, and finally selects a set of most significant segmentation regions for that hierarchy level. The boundaries for that level are then computed by using graph cut [5] with a capacity that considers both the curvature and MSP gradient.

The paper makes the following contributions:

1. Propose a segmentation scheme based on a surface function MSP that encodes local volume information. The proposed segmentation scheme has several advantages:

   - The MSP function is an intermediate-level surface property. It is more global than surface functions that represents local geometry detail, such as curvature, and less global than functions such as geodesic function.
   - Regions of similar volume are grouped into a part according to the intermediate-level surface function MSP.
   - A significance measure of a boundary that takes into account the local curvature and the changes in MSP value is presented.

2. Propose a hierarchical segmentation on which not only components on a higher level reveal higher degree of salience than their descendant parts but also the components on each level of hierarchy have similar degree of salience.

2. RELATED WORK

In the past decade, many mesh segmentation methods have been proposed. Based on the objective, mesh segmentation methods generally fall into two categories: patch-type and part-type [9]. Patch-type segmentation usually decomposes the mesh into several patches by analyzing the surface properties such as dihedral angles [10, 11], curvature [11-13], geodesic distance [14], and planarity [4]. Part-type segmentation tends to segment the complex object into several meaningful components, usually based on the concepts from cognition theory [15, 16]. For example, the minima rule states that human perception tends to break an object into parts along the region of minimum negative curvature [15]. Moreover, the salience of parts determined by relative volume, boundary strength, and degree of protrusion is important for human perception [16]. Our method is a part-type segmentation, and we will mainly review this type of segmentation and refer readers to the excellent survey paper by Shamir [9] for the patch-type segmentation.

As described in section 1, locating boundary between parts can be done by either boundary-based or region-based approach. Mangan and Whitaker used the watershed to segment the object into several parts according to the curvature on the surface [12]. Lee *et al.* [7] cut the object into parts by first finding the loop along the minimum negative curvature, and then test the salience of the divided parts based on the part salience theory [16]. However, the surface curvature is too local for describing the shape of object and locating cut boundaries based on the curvature cannot always result in a meaningful part segmentation. Moreover, the iterative segmentation process proposed leads to a binary hierarchical segmentation on which each level does not provide intuitive meaning for object parts.

The geodesic distance is another attribute widely used in mesh segmentation [5, 17-20]. The *averaged geodesic distance* (AGD) derived as the average of geodesic distance from a surface point to all other points can be used to represent the degree of protrusion of a part. However, such attributes are useful only for models that have obvious core part and feature parts.

The *Medial Axis Transform* (MAT) is a global shape descriptor of the object [20]. MAT or skeleton can be used as a guideline for segmentation. For example, Li *et al.* segmented the object by moving a sweep plane along the skeleton of object [22]. Since

the size of the cutting section of the sweeping plane can be regarded as local volumetric information of the object, the cut boundaries are usually at the regions where the size of cutting section varies rapidly. Oscar *et al.* segmented the skeletal mesh by measuring thickness corresponding to the skeleton nodes derived from the skeletonization process [23]. The most concave region for the cutting is searched for each skeleton branch by comparing the thickness of the skeleton nodes with their neighbors [23]. Reniers and Telea observed that the junction between two skeleton paths has high potential to be a good place for separating two parts [24, 25]. A surface attribute related to MAT is proposed in [3], called *Shape Diameter Function* (SDF). SDF measures the local diameter of the object at the surface points by sampling the rays fired from the surface point inward to the other side of the mesh and averaging the length of those rays sampled. They also proposed a hierarchical segmentation method by fitting k Gaussian functions to the histogram of SDF values, and clustering the mesh faces according to their corresponding Gaussian functions. However, the fitting of the global histogram can not reflect the difference between the object parts and some small parts with no salient feature may be segmented. The segmentations generated by using different number of Gaussian functions do not have consistent part correspondence and the part boundaries may not always lie on meaningful regions.

An iterative approach for decomposing the object into several parts is based on the k-means clustering [26]. Shlafman *et al.* used k-means clustering to segment the object into a user-specified number of components [17]. This work was later refined to achieve hierarchical segmentation [5]. However, the geodesic distance used in [5] describes only the protrusion of object parts and hence the resultant hierarchical segmentation tends to cut the object along the longest parts at the higher level of the hierarchy, which may not meet the concept of perception. Moreover, the method is usually suitable only for objects having obvious core-salient features. The challenge to the k-means clustering methods is that the value of k needs to be given a priori. Liu and Zhang overcame such problems by applying the spectral analysis on the affinity matrix constructed using the mesh faces [18].

Another segmentation approach was based on the fitting of primitives. Attene *et al.* extended the hierarchical face clustering [4] and replaced the clustering metric by other similarity measures for the predefined primitives, such as spheres, cylinders and planes [27]. Mortara *et al.* detected the parts with tubular shape from the whole object [28]. They extracted the core part by excluding all the tubular parts from the object to complete the segmentation.

Pose-invariant mesh segmentation has attracted more attention in recent years. Such works focused on finding the consistent segmentation over different poses of a model. Katz *et al.* transformed the original model into a pose-invariant representation using multi-dimensional scaling and then used spherical mirroring to extract the core of the object and feature points to segment the objects [6]. In character animation, the pose-invariant segmentation can be achieved by finding the rigid components during the animation [29-31]. However, such methods are usually suitable only for articulated models and the requirement of animation sequences also poses some restrictions on the usability.

Golovinskiy and Funkhouser [32] defined a surface function called partition function that indicates how likely each edge is to lie on the boundary of a random segmentation drawn from a set of segmentations. Based on the partition function, a cut is associated with a consistency measure as the length-weighted average of the partition function values of its edges. The most consistent cuts defined as the set of cuts with highest consistency are used for finding the part boundaries. Chen *et al.* proposed a benchmark for

quantitative evaluation of mesh segmentation algorithms [33]. The benchmark includes a data set with 4,300 manually generated segmentations for 380 object meshes in 19 categories. It also provides software for producing four quantitative metrics for the comparison of segmentation algorithms.

3. VOLUME BASED MESH SEGMENTATION

3.1 Overview

The goal of our segmentation scheme is to utilize the volume information for segmentation. The *Minimum Slice Perimeter* (MSP) function [1] is a surface function that encodes the internal volume information around the surface point. We observed that neighboring surface regions with similar volume tend to be grouped into a part and consequently the gradient of the MSP function directly implies the potential regions for deriving part boundaries. The proposed scheme begins by computing the MSP values for each face of the mesh and deriving the gradient of the MSP function. The segmentation process then finds a set of segmentation regions by applying a threshold value to the gradient of MSP function. Each segmentation region divides the object into two or more potential object parts. We then test the saliency of the potential parts and obtain some most significant segmentation regions. Finally, a cut is derived within each selected segmentation region which separates the object into two parts. Fig. 1 illustrates the segmentation process.

(a) MSP.   (b) MSP gradient.   (c) Segmentation regions.   (d) Segmented parts.
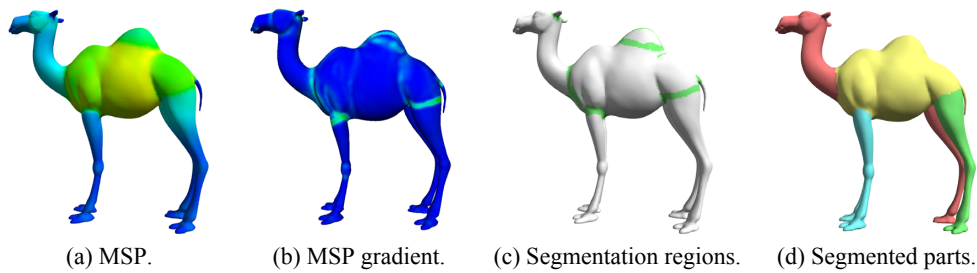Fig. 1. Steps of the mesh segmentation process.

The selection of the most significant parts directly implies that a hierarchical segmentation can be obtained easily by performing the segmentation process iteratively. Such a hierarchical segmentation possesses a property that parts at the same level will have similar saliency and parts at a higher level have higher saliency than parts at lower levels.

3.2 Minimum Slice Perimeter Function

The object's local volume is a valuable surface property for describing the shape of the object. To associate the internal volume information with the mesh surface, we use a recently developed MSP function that is derived from the concept of short-cut [2] and approximates the best possible short-cut for a surface point as the planar slice that passes

through the point and perpendicular to the surface and has the smallest perimeter of the intersection slice with the mesh surface.

We define the MSP function for a surface point p as the minimum perimeter of the planar slices passing through p and perpendicular to the surface at p as in Eq. (1).

$$\mathbf{MSP}(p) = \min_{n} \|\text{plane}(n, p) \cap M\| \tag{1}$$

where plane(n, p) represents a plane passing p with normal n. The minimum perimeter of slices aims to serve as a good approximation of the local volume around the surface point. Since the short cut is not necessarily perpendicular to the surface, requiring the planar slice perpendicular to the surface may be too strong for obtaining a good approximation of the short cut, especially around the tip of the protrusion parts. In the implementation, we relax the requirement and allow planar slice to be parallel to the direction within a normal cone of the surface point.

Fig. 2 demonstrates the MSP distribution on several models. The color ranging from blue to red represents the MSP value in ascending order. As we can see that the core parts have higher MSP value than the salient parts for the articulated models and, for models having complex topological structures, parts can be distinguished easily based on the MSP distribution. In Fig. 3, MSP and SDF are compared. Both MSP and SDF can reveal the relative volume size well for cylindrical parts, but the MSP performs better for the object parts that are highly non-cylindrical, such as the palm in Fig. 3. The SDF in general measures the thickness of the object part and hence represents only part of the volume information. Moreover, the MSP describes the changes in shape better than SDF since there exists large gaps in MSP value between the palm and the fingers while the SDF tends to blur the function value within such regions.
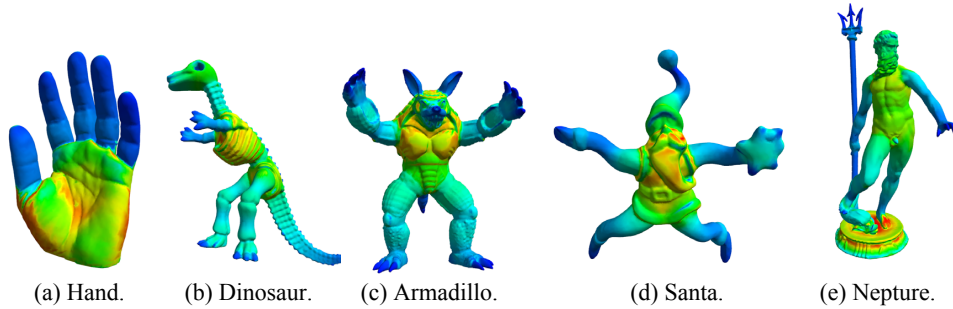

(a) Hand.   (b) Dinosaur.   (c) Armadillo.   (d) Santa.   (e) Nepture.

Fig. 2. The MSP on different models.

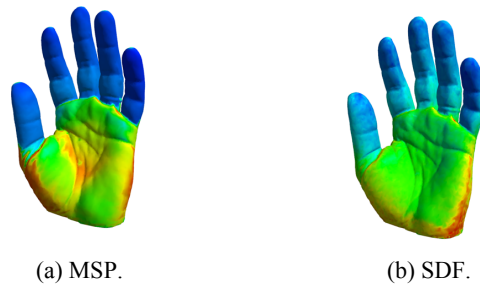
(a) MSP.   (b) SDF.

Fig. 3. Coloring of MSP and SDF on the olive hand model.

3.3 Segmentation Regions Finding

Computing the gradient of MSP function   The gradient of MSP function represents the rate of volume change in the neighborhood of the surface point. As shown in Fig. 4, there are large gaps in volume's size between the body and the neck and between the body and front limbs of the camel model, which are revealed in the distribution of MSP function (Fig. 4 (a)) and its gradient (Fig. 4 (b)). Computing the gradient of MSP function on a piecewise linear polygon mesh is not as trivial as that on the continuous surface. To compute the gradient of MSP function at a surface point x, we first derive a difference-vector for every edge in a neighborhood of x and then compute the MSP gradient vector at x as the average of all difference-vectors. The difference-vector for an edge is the vector in the direction of the edge vector and with the magnitude as the difference of the MSP values at its two endpoints.

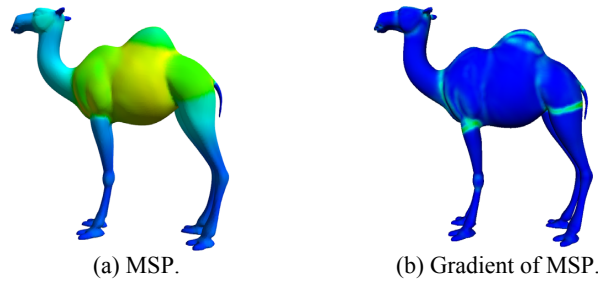(a) MSP.   (b) Gradient of MSP.

Fig. 4. MSP and its gradient on the camel model.

The gradient of MSP function at x is finally the magnitude of the averaged MSP difference-vector at x. Its detail equation is as follows,

$$\nabla \text{MSP}(x) = \frac{1}{|B|} \left\| \sum_{e \cap B} (\text{MSP}(e_i) - \text{MSP}(e_j))\overline{e} \right\|,$$ (2)

where B is the neighborhood of x with a user-specified range, e_i and e_j are two end points of the edge e, possibly clamped to be within B, and $\overline{e}$ is the unit vector of the edge e. Such a formulation is similar to the method for computing the curvature tensor in Alliez *et al.* [34]. For efficiency consideration, we define B as the surface region within the sphere centering at x with a user-specified radius.

Deriving an appropriate threshold value   The segmentation regions are defined as the surface regions where the gradient of MSP is above a specific threshold value. An appropriate threshold value is hard to find in practice. A smaller threshold value will enlarge the segmentation regions, making the computation of a proper cut boundary more difficult. For a large threshold value, the segmentation region may not form in loops, even using extrapolation. To decide an appropriate threshold value automatically, we consider the cumulative function for the surface area with respective to the MSP gradient as follows,

$$A(V)\bigg| = \int_0^y a(v)dv,$$ (3)

where $a(v)$ denotes the area of the surface region having MSP gradient as v and $A(V)$ is the total area of the surface region that has the MSP gradient less than or equal to V. $A(v)$ is a monotonically increasing function, representing the changes of cumulative surface area with respect to the MSP gradient. A good threshold value will be the MSP gradient value that indicates a sudden drop on the value of $A(v)$ when the MSP gradient decreases. Hence, by considering $A(v)$ as a 2D curve segment, the desired threshold value will be at the position where the curvature is maximal. Figs. 5 (a) and (b) illustrate the graph of $A(v)$ and its curvature for the horse model, respectively. The segmentation regions extracted using the derived threshold value is shown in Fig. 5 (c).
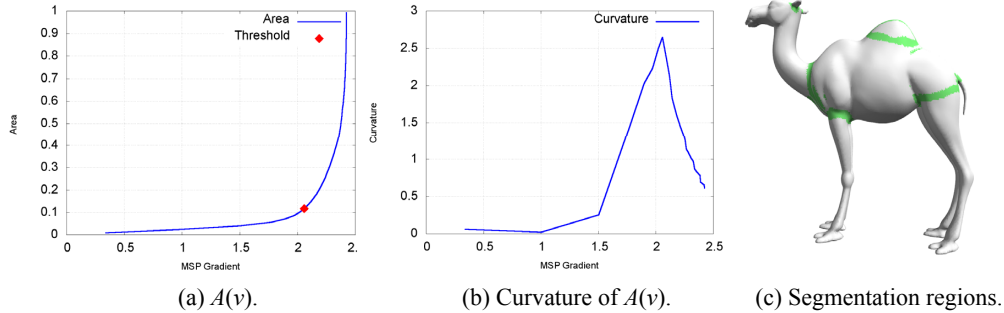

      (a) $A(v)$.   (b) Curvature of $A(v)$.   (c) Segmentation regions.
Fig. 5. $A(V)$, curvature of $A(V)$, and segmentation regions derived for the camel model.

Loop closing for segmentation regions   Since the cut boundary is extracted within the segmentation region, a segmentation region must form a loop. The segmentation region derived by applying threshold on MSP gradient in general does not guarantee to form a loop. We apply extrapolation to close those segmentation regions into loops. For a segmentation region that does not form a loop, a region growing process is performed starting from the region boundary, and in each iteration the face with the largest MSP gradient is added to the segmentation region until the newly added face connects to another region or close the loop. After closing up a loop, the faces within the newly grown region are peeled away iteratively in the decreasing order of their MSP gradients until a ribbon region with one face width left between the two newly connected segmentation regions. At this point, these two segmentation regions are merged. The merging process is executed iteratively until all segmentation regions form in loops. Fig. 6 illustrates the loop closing process.
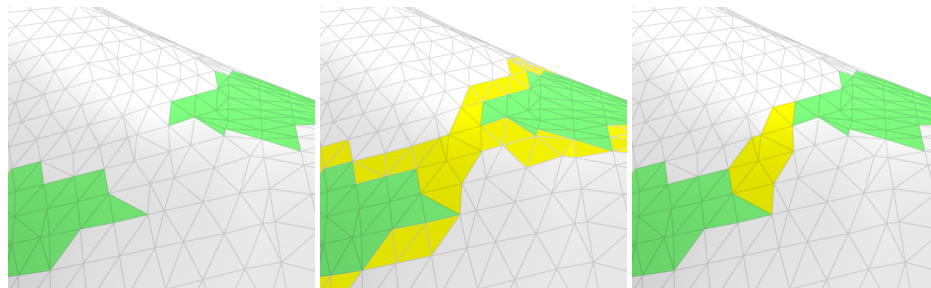
(a) Original segmentation regions.   (b) Region extrapolation.   (c) Grown region peering.
Fig. 6. The loop closing process.

### 3.4 Significant Segmentation Pairs Selection

The segmentation region derived using the gradient of MSP represents the strength of the part boundary, but does not reveal any salience information of parts it separated. However, the salience of segmented parts associated with the segmentation regions may range from detail feature to global structure.

There may have more than two parts associated with a segmentation region. Thus for a segmentation region, we define a segmentation pair to be two parts sharing the same segmentation region. The part saliency theory states that the salience of a 3D part depends on three factors: its size related to the whole object, the boundary strength, and the degree of protrusion [16]. Since we have derived the segmentation regions using MSP gradient, the local volume of an object part can be approximated by the average MSP value within the part region. The strength of boundary between a segmentation pair can be described as the difference of the averaged MSP values of the two parts. In order to measure the protrusion of the object part, we use the salience metric proposed by Gal and Cohen-Or [35] due to its computational efficiency and practicability for identifying surface features. Thus, we assign a salience-measure to each of segmentation pair (p_a, p_b) as follows,
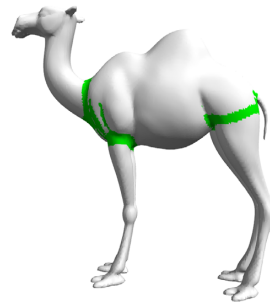
$$S(p_a, p_b) = \mathbf{min}(\mathbf{MSP}(p_a), \mathbf{MSP}(p_b))\mathbf{min}(S(p_a), S(p_b))\|\mathbf{MSP}(p_a) - \mathbf{MSP}(p_a)\|, \quad (4)$$

where p_a and p_b are the two parts sharing the segmentation region, $S(p) = \sum_{f \in p} Area(f)Curvature(f)^2$ denotes the saliency of the part p, and $\mathbf{MSP}(p)$ is the averaged MSP value of the part p. We use the Gaussian curvature in saliency computation since it has better description for the protrusion of object part.

To find the most significant segmentation pairs, we sort the value of S for all segmentation pairs in ascending order into a sorted sequence $\{S^*(i)\}$, and seek an index k such that $S^*(k) - S^*(k - 1)$ is the maximum; that is, look for the largest gap among the sorted $S^*(i)$. Those segmentation pairs that have S value higher than $S^*(k - 1)$ will be chosen as the segmentation regions for the current hierarchy level. As shown in Fig. 7, there is a large gap in the histogram of S between the segmentation pairs 17 and 18, and segmentation pairs 18 to 21 are selected.
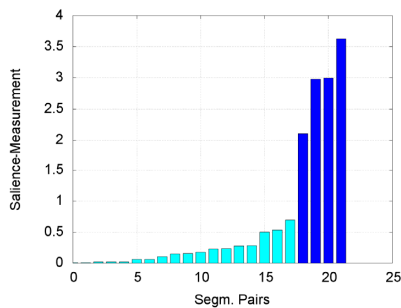
(a) Salience-measure function histogram.   (b) Selected segmentation pairs.
Fig. 7. Selection of the segmentation pairs.

3.5 Cut Boundary Extraction for the Selected Segmentation Pair

Once a segmentation pair is selected, we next compute a cut boundary between parts by performing a modified graph cut [5]. We propose a hybrid capacity that combines the gradient of MSP as well as angle difference as follows,

$$\text{cap}(i, j) = \begin{cases} 0 & e_{i,j} \in \cap_{k<l} \text{cut}_k \\ \dfrac{1}{1 + \left(\dfrac{\theta(e_{i,j})}{\theta_{\text{avg}}}\right)^2 + \dfrac{\partial \mathbf{MSP}(e_{i,j})}{\partial \mathbf{MSP}_{\text{avg}}} + 2 \dfrac{\|e_{i,j}\|}{\|e_{\text{avg}}\|}} & e_{i,j} \in E, i, j \neq S, T \\ \infty & \text{otherwise,} \end{cases} \qquad (5)$$

where i and j are two faces on the object that are not in both the source region S and the target region T, e_{ij} denotes the edge between faces i and j, $\theta(e_{ij})$, $\partial \mathbf{MSP}(e_{ij})$, and $\|e_{ij}\|$ are the dihedral angle of e_{ij}, the MSP difference across e_{ij}, and the edge length of e_{ij}, normalized by the averaged dihedral angle (θ_{avg}), the averaged MSP difference ($\partial \mathbf{MSP}_{\text{avg}}$), and the averaged edge length $\|e_{\text{avg}}\|$) over the object, respectively. The first equation in Eq. (5) allow the current cut path cut_l to lie along the cuts cut_k, $k < l$, that have been generated previously. The $\dfrac{\|e_{i,j}\|}{\|e_{\text{avg}}\|}$ in Eq. (5) is a compensated term aiming to straighten the cut boundary since the path generated by using the tip of MSP gradient may not always be smooth.

3.6 Hierarchical Segmentation

The proposed segmentation can be adapted easily to a hierarchical scheme by iteratively performing the segmentation process described in previous sections. For each level of hierarchy, the selection of the most significant segmentation pairs ensures that the selected parts will have similar salience significance while having large differences compared to the remaining parts. Thus we obtain a top-down hierarchical segmentation that decomposes the object into parts into levels that reveal the salience significance down from the global structure to local features. Our segmentation regions are found by applying a threshold value to the cumulative function shown in Eq. (3). In order to capture more boundaries in finer levels, for each level we ignore surface area nearby the part boundaries derived in previous levels in computing Eq. (3). As a result, a lower threshold value on the histogram can be obtained to enlarge the segmentation regions. Fig. 8 illustrates the segmentation regions found in three levels for the camel model. We observe that the threshold value applied decreases as the segmentation level gets more deep (see the numbers inside the parentheses) and the segmentation regions for smaller features, such as the mouth and the fingers, are found in finer levels. Fig. 9 reveals the segmentation results of the camel model corresponding to the segmentation regions in Fig. 8.
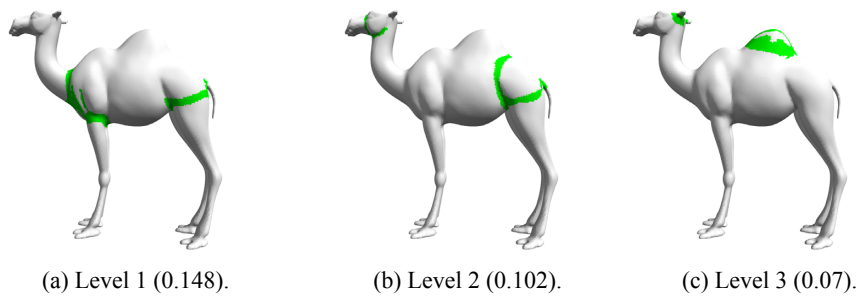
(a) Level 1 (0.148). (b) Level 2 (0.102). (c) Level 3 (0.07).

Fig. 8. Segmentation regions at three levels on the camel models.
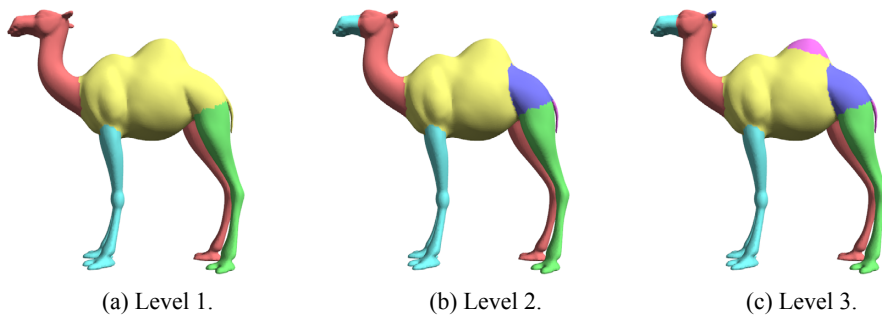

(a) Level 1. (b) Level 2. (c) Level 3.

Fig. 9. Hierarchical segmentation of the camel model in three levels.

4. EXPERIMENTAL RESULTS

By using the volume information encoded in the MSP function, the proposed segmentation scheme can handle models of different topological types. Fig. 10 demonstrates the segmentation result for models shown in Fig. 2. Observed that the cut boundaries are located along the regions where there exists a large gap in MSP value and in the meantime follow the object's local features and, moreover, the salience-measure ensures that the most visually significant parts are segmented. The dinosaur (Fig. 10 (b)) and armadillo (Fig. 10 (c)) models have complex surface details and hence it is difficult to locate correct cut boundaries based on minima-rule alone. Since the MSP function is less sensitive to the surface detail noise, we can generate meaningful parts and reasonably good boundaries using the proposed hybrid capacity on these two models. The nepture (Fig. 10 (e)) model has genus higher than 1 and does not have obvious core-salient structure. Such kind of model can hardly be handled well using segmentation methods based on the global shape properties such as averaged geodesic distance [5, 6]. On the other hand, the proposed method finds no difficulty on such models since the volume information encoded by MSP function is less global and provides enough cues for identifying the parts from the models. Some smooth artifacts on cut boundaries may still be observed on some segmentation results such as the cut boundaries between the hind legs and the body of the camel (Fig. 9) and between the thighs and the body of the armadillo (Fig. 10 (c)) and of nepture (Fig. 10 (e)). In such regions, the tip of MSP gradient has large amount of disturbance and twice of compensation term in Eq. (5) may be not enough for yielding smooth boundaries. Moreover, we assume that the internal volume is uniform within the

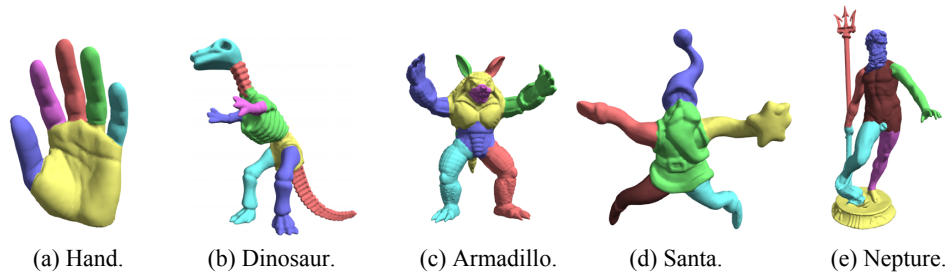| (a) Hand. | (b) Dinosaur. | (c) Armadillo. | (d) Santa. | (e) Nepture. |

Fig. 10. The segmentation result of different models.

object part while noticeable volume change exists between parts. For models with some adjacent parts that have no noticeable volume change between them, our segmentation scheme may fail to separate them into different parts, such as the case in Fig. 10 (e) where the hand and trident are not separated.

In Fig. 11, we compare the proposed method to other methods provided in the segmentation benchmark [33]. Cup and chair models do not have obvious core-salient structure on which the core extraction [6] and K-means [17] algorithms fail to produce good segmentations. The segmentation using SDF is based on a global fitting of the histogram function and is unable to reflect the local changes in object volume, leading to biased segmentation boundaries. Moreover, the SDF cannot correctly describe the non-cylindrical part and generates improper segments on the cup model. The randomized cuts method [32] generates good segmentation results for most of the models. However, its performance strongly depends on the existing segmentation methods it employs. For example, in [32] the randomized cuts method employs methods based on the local curvature and the geodesic distance, and in consequence, it may not generate good segmentations for models with complex local features. Fig. 12 shows the segmentation result of dinosaur model using the proposed scheme and the randomized cuts [32]. The proposed segmentation can tolerate the complex local features of the dinosaur model and segments the models into parts with similar salience significance. On the other hand, randomize cuts algorithm produces improper segmentation boundaries at the neck, the body, and the tail. We also perform a benchmark study of the proposed method against others using the segmentation benchmark [33]. The benchmark study is obtained by performing the comparison based on 20 models selected from the object database of the segmentation benchmark (two models from each object category). Fig. 13 reveals that the proposed segmentation scheme always yields lower error than others for the four metrics proposed in [33].

Since the interior volume of a model is almost constant during animation, the proposed segmentation scheme is inherently pose invariant. We list the segmentation result of the animated centaur model in four poses in the top of Fig. 14. The bottom of Fig. 14 illustrates the average error rate of MSP function for each of the four poses. For each pose, the MSP's average error rate is computed by averaging the differences in MSP value between the pose and all other poses. The MSP is almost invariant to the change of pose, except in some joint regions where very small deviation of MSP value may exist.

We decompose the dinosaur and armadillo models into a hierarchy of four levels, as shown in Fig. 15. The columns from left to right indicate the levels in ascending order. The most significant parts, such as the body of armadillo and the four limbs, are decomposed at the first level. As going down in the hierarchy, we observe that parts at the same level have similar salience significance and parts in lower levels have less salience sig-
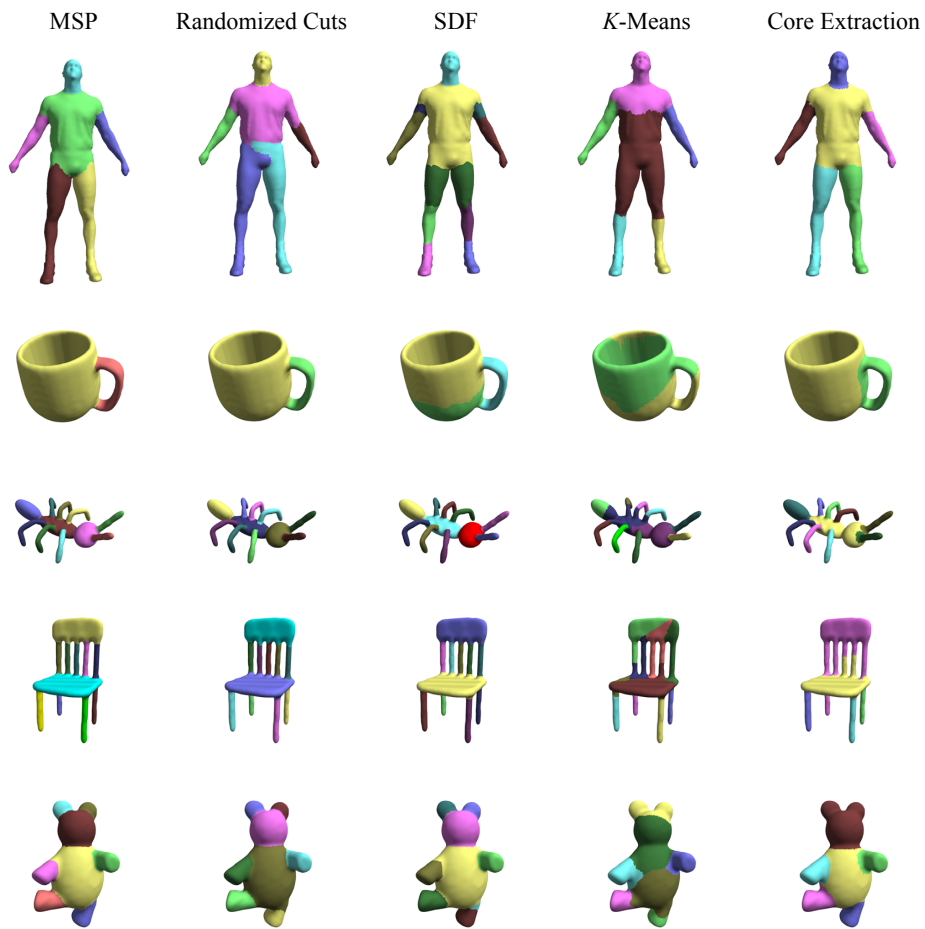
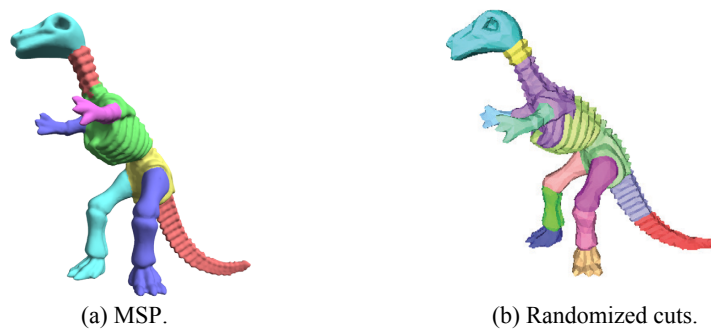Fig. 11. Comparison of the segmentation methods.

(a) MSP.   (b) Randomized cuts.

Fig. 12. Comparison of dinosaur result using MSP and randomized cuts [32].

(a) Cut discrepancy.

(b) Hamming distance.

(c) RandIndex.

(d) Consistency error.
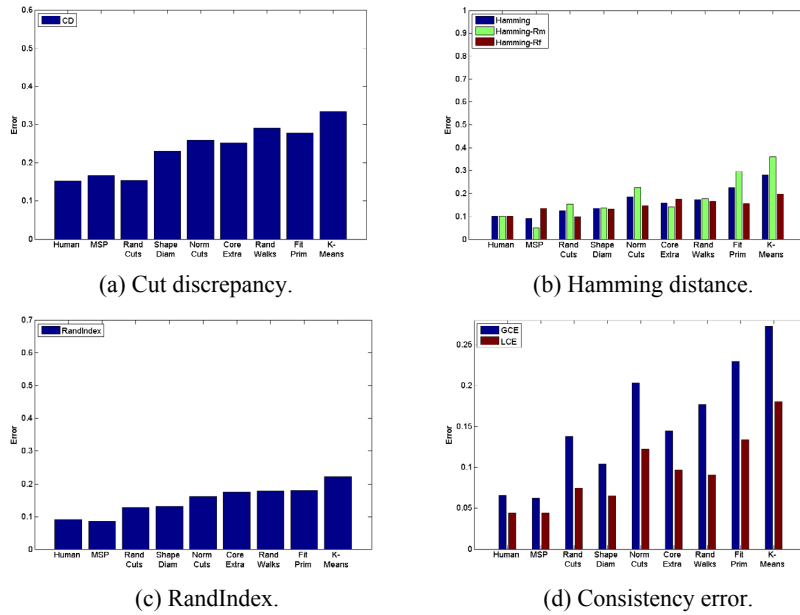
Fig. 13. Comparison of dinosaur result using MSP and randomized cuts [27].

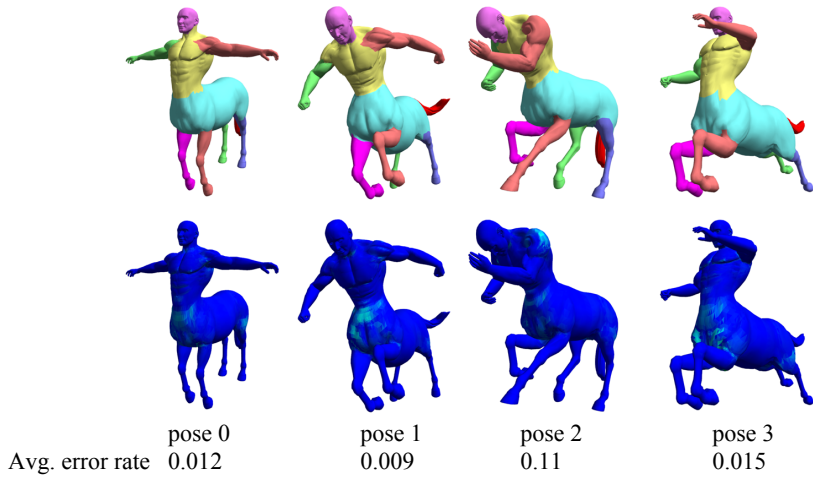|	pose 0	pose 1	pose 2	pose 3
Avg. error rate	0.012	0.009	0.11	0.015

Fig. 14. Segmentation and average error rate for different poses of the animated centaur model.

nificance. Fig. 16 depicts the histogram plots of salience-measure for the dinosaur model. The parts having similar meaning tend to have similar value of salience-measure and will be decomposed at the same level. Fig. 17 lists the hierarchical segmentation result using SDF [3]. The boundaries of core part for the dinosaur and armadillo are varying among different levels and moreover, parts at the same level might differ greatly in salience significance and parts at lower levels may not have less salience significance.
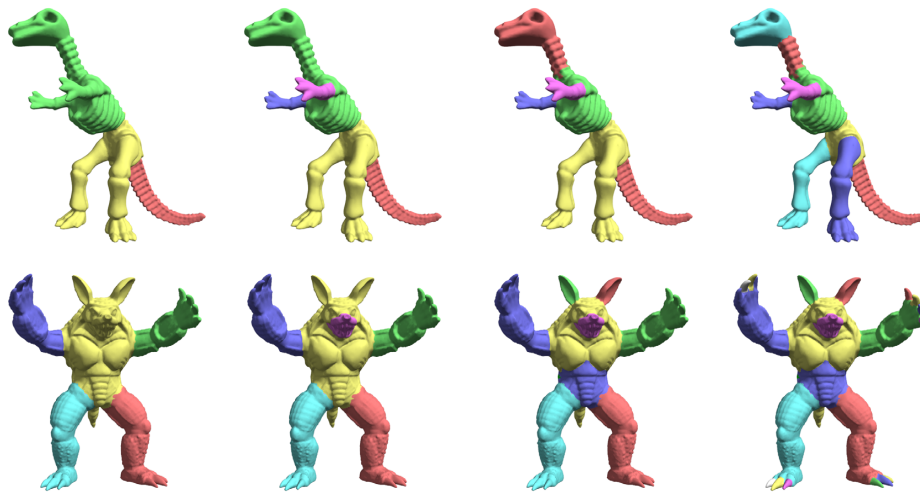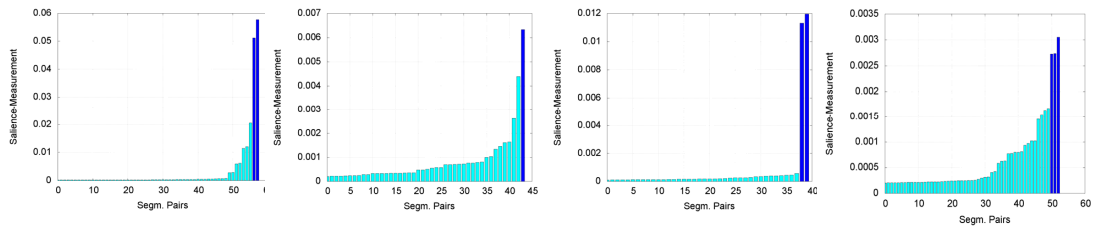
Fig. 15. Hierarchical segmentation result of the dinosaur and armadillo models.

(a) Level 1.   (b) Level 2.   (c) Level 3.   (d) Level 4.

Fig. 16. Histograms of the salience-measure at four levels for the dinosaur model.
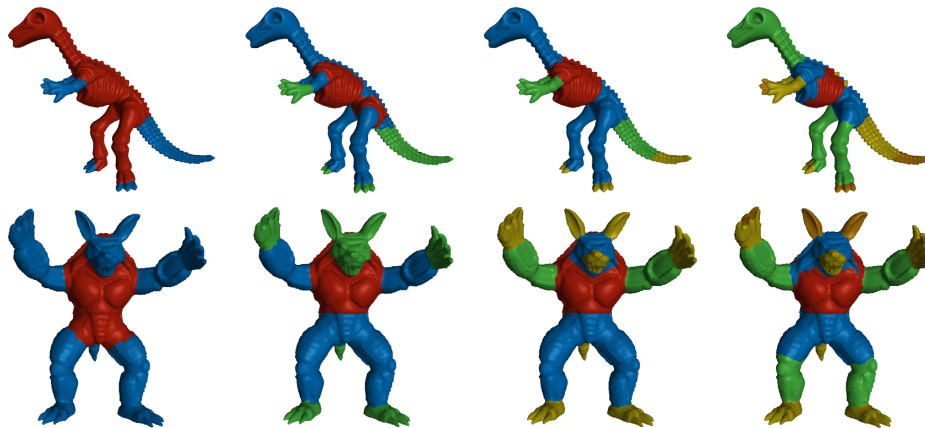

Fig. 17. Hierarchical segmentation result of the dinosaur and armadillo models using SDF [3].

5. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we have proposed a new hierarchical segmentation scheme that decomposes an object into several levels of meaningful parts. The segmentation hierarchy obtained possesses some desired properties. For example, components on a higher level reveal higher degree of salience than their descendant parts and the components on each level of hierarchy have similar degree of salience. Moreover, the number of boundaries on each level of the hierarchy is determined automatically. The segmentation regions are found by using the gradient of Minimum Slice Perimeter (MSP) function [1]. MSP function is a new volume-based surface function that is capable of encoding more global volume information around a surface point than previous MAT related functions. We have also evaluated the visual significance of a segmentation region by taking into account the salience information of the parts adjacent to it and based on that salience measure we select some most perceptually significant segmentation regions for each level of the segmentation hierarchy. The information encoded by MSP function is more global than surface functions that represents local geometry detail, such as curvature, and less global than functions such as geodesic function. We have performed a benchmark study using the segmentation benchmark [33] and found that the proposed method performs better than the methods supported in the benchmark.

The cutting threshold for the largest gap on the sorted salience histogram can distinguish the most significant segmentation pairs from others. However, the salience-measure defined as the combination of MSP and other local properties is still not good enough and may result in some improper segmentation results. A better metric for describing the salience of the object parts is expected. Moreover, the minimum perimeter slice used for computing the MSP value is originated from the short-cut rule [2] and provides hints for evaluating how good an object part is. We are working on the design of a new metric for evaluating the goodness of an object part based on the observation that to be a good part all minimum perimeter slices of points inside an object part tend to fall inside the part region.

REFERENCES

1. T. C. Ho, W. S. K. Wong, and J. H. Chuang, "Mesh skeletonization using minimum slice perimeter function," Technical Report, No. CS-TR-2010-0001, Department of Computer Science, National Chiao Tung University, 2010.
2. M. Singh, G. D. Seyranian, and D. D. Hoffman, "Parsing silhouettes: The short-cut rule," *Perception and Psychophysics*, Vol. 61, 1999, pp. 636-660.
3. L. Shapira, A. Shamir, and D. Cohen-Or, "Consistent mesh partitioning and skeletonisation using the shape diameter function," *The Visual Computer*, Vol. 24, 2008, pp. 249-259.
4. M. Garland, A. Willmott, and P. S. Heckbert, "Hierarchical face clustering on polygonal surfaces," in *Proceedings of Symposium on Interactive 3D Graphics*, 2001, pp. 49-58.
5. S. Katz and A. Tal, "Hierarchical mesh decomposition using fuzzy clustering and cuts," *ACM Transactions on Graphics*, Vol. 22, 2003, pp. 954-961.
6. S. Katz, G. Leifman, and A. Tal, "Mesh segmentation using feature point and core extraction," *The Visual Computer*, Vol. 21, 2005, pp. 649-658.

 7. 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*, Vol. 22, 2005, pp. 444-465.
 8. Y. K. Lai, Q. Y. Zhou, S. M. Hu, and R. R. Martin, "Feature sensitive mesh segmentation," in *Proceedings of ACM Symposium on Solid and Physical Modeling*, 2006, pp. 17-25.
 9. A. Shamir, "A survey on mesh segmentation techniques," *Computer Graphics Forum*, Vol. 27, 2008, pp. 1539-1556.
10. B. Lévy, S. Petitjean, N. Ray, and J. Maillot, "Least squares conformal maps for automatic texture atlas generation," *ACM Transactions on Graphics*, Vol. 21, 2002, pp. 362-371.
11. A. Sheffer, "Model simplification for meshing using face clustering," *Computer-Aided Design*, Vol. 33, 2001, pp. 925-934.
12. A. P. Mangan and R. T. Whitaker, "Partitioning 3 D surface meshes using watershed segmentation," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, 1999, pp. 308-321.
13. D. L. Page, A. Koschan, and M. A. Abidi, "Perception-based 3D triangle mesh segmentation using fast marching watersheds," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2003, pp. 27-32.
14. E. Zhang, K. Mischaikow, and G. Turk, "Feature-based surface parameterization and texture mapping," *ACM Transactions on Graphics*, Vol. 24, 2005, pp. 1-27.
15. D. D. Hoffman and W. Richards, "Parts of recognition," *Cognition*, Vol. 18, 1984, pp. 65-96.
16. D. D. Hoffman and M. Singh, "Salience of visual parts," *Cognition*, Vol. 63, 1997, pp. 29-78.
17. S. Shlafman, A. Tal, and S. Katz, "Metamorphosis of polyhedral surfaces using decomposition," *Computer Graphics Forum*, Vol. 21 2002, pp. 219-228.
18. R. Liu and H. Zhang, "Segmentation of 3D meshes through spectral clustering," in *Proceedings of Pacific Graphics*, 2004, pp. 298-305.
19. R. Liu and H. Zhang, "Mesh segmentation via spectral embedding and contour analysis," *Computer Graphics Forum*, Vol. 26, 2007, pp. 385-394.
20. Y. S. Wang, C. H. Lin, and T. Y. Lee, "Interactive model decomposition using protrusive graph," *International Journal of Innovative Computing, Information and Control*, Vol. 4, 2008, pp. 1887-1896.
21. H. I. Choi, S. W. Choi, and H. P. Moon, "Mathematical theory of medial axis transform," *Pacific Journal of Mathematics*, Vol. 181, 1997, pp. 57-88.
22. X. Li, T. W. Woon, T. S. Tan, and Z. Huang, "Decomposing polygon meshes for interactive applications," in *Proceedings of Symposium on Interactive 3D Graphics*, 2001, pp. 35-42.
23. O. K. C. Au, C. L. Tai, H. K. Chu, D. C. Or, and T. Y. Lee, "Skeleton extraction by mesh contraction," *ACM Transactions on Graphics*, Vol. 27, 2008, pp. 1-10.
24. D. Reniers and A. Telea, "Skeleton-based hierarchical shape segmentation," in *Proceedings of IEEE International Conference on Shape Modeling and Applications*, 2007, pp. 179-188.
25. D. Reniers and A. Telea, "Part-type segmentation of articulated voxel-shapes using the junction rule," *Computer Graphics Forum*, Vol. 27, 2008, pp. 1845-1852.

26. S. P. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, Vol. 28, 1982, pp. 129-137.
27. M. Attene, B. Falcidieno, and M. Spagnuolo, "Hierarchical mesh segmentation based on fitting primitives," *The Visual Computer*, Vol. 22, 2006, pp. 181-193.
28. M. Mortara, G. Patané, M. Spagnuolo, B. Falcidieno, and J. Rossignac, "Plumber: A method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies," in *Proceedings of ACM Symposium on Solid Modeling and Applications*, 2004, pp. 339-344.
29. T. Y. Lee, Y. S. Wang, and T. G. Chen, "Segmenting a deforming mesh into near-rigid components," *The Visual Computer*, Vol. 22, 2006, pp. 729-739.
30. T. Y. Lee, P. H. Lin, S. U. Yan, and C. H. Lin, "Mesh decomposition using motion information from animation sequences," *Computer Animation and Virtual Worlds*, Vol. 16, 2005, pp. 519-529.
31. T. Y. Lee, C. H. Lin, Y. S. Wang, and T. G. Chen, "Animation key-frame extraction and simplification using deformation analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 18, 2008, pp. 478-486.
32. A. Golovinskiy and T. Funkhouser, "Randomized cuts for 3D mesh analysis," *ACM Transactions on Graphics*, Vol. 27, 2008, pp. 1-12.
33. X. Chen, A. Golovinskiy, and T. Funkhouser, "A benchmark for 3D mesh segmentation," *ACM Transactions on Graphics*, Vol. 28, 2009, pp. 1-12.
34. P. Alliez, D. Cohen-Steiner, O. Devillers, B. Lévy, and M. Desbrun, "Anisotropic polygonal remeshing," *ACM Transactions on Graphics*, Vol. 22, 2003, pp. 485-493.
35. R. Gal and D. Cohen-Or, "Salient geometric features for partial shape matching and similarity," *ACM Transactions on Graphics*, Vol. 25, 2006, pp. 130-150.

Tan-Chi Ho (何丹期) received his B.S. degree in Computer Science from National Chiao Tung University, Taiwan, in 2001, and he is currently a Ph.D. student in Department of Computer Science, National Chiao Tung University. His research interests are geometric modeling, GPU based simulation, and visibility pre-processing in computer graphics.

Jung-Hong Chuang (莊榮宏) is a Professor of Computer Science Department at National Chiao Tung University, Taiwan. He received his B.S. degree in Applied Mathematics from National Chiao Tung University, Taiwan, in 1978, and M.S. and Ph.D. degrees in Computer Science from Purdue University in 1987 and 1990, respectively. Chuang's primary research interests are in the areas of 3D graphics, virtual reality, and geometric modeling. In 3D graphics, he works on the modeling and the real-time rendering techniques for 3D game design, virtual reality and visual simulation.