JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 25, 971-987 (2009)

Principal Component Analysis-based Mesh Decomposition*

JUNG-SHIONG CHANG, ARTHUR CHUN-CHIEH SHIH†, HSIAO-RONG TYAN‡
AND WEN-HSIEN FANG
Department of Electronic Engineering
National Taiwan University of Science and Technology
Taipei, 106 Taiwan
†*Institute of Information Science*
Academia Sinica
Taipei, 115 Taiwan
‡*Department of Information and Computer Engineering*
Chung Yuan Christian University
Chungli, 320 Taiwan

We propose an automatic mesh decomposition technique based on principal component analysis (PCA) and Boolean operations. First, we calculate the normalized protrusion degree of each dual vertex on a smoothed 3-D mesh. The protrusion degree of a vertex and the vertex's 3-D coordinates form a 4-D feature vector, which we use to represent the polygon mesh. Then, we apply PCA to the set of 4-D feature vectors. The projected data along the first principal axis reveals the salient structures of the 3-D object. Therefore, by using the first component axis as the search basis, we can identify all the salient parts of an arbitrary 3-D object.

Keywords: mesh decomposition, Boolean operation, PCA, protrusion degree, 3-D object

1. INTRODUCTION

Mesh decomposition is a major procedure that is critical to generating an efficient representation of a mesh object. After proper decomposition, the decomposed components can be individually selected, grouped, and analyzed according to the properties of interest [1]. A variety of applications have been developed based on this decomposition method, for example, 3-D shape retrieval [4, 13, 15], collision detection [10], metamorphosis [13, 17], skeleton extraction [8, 16], simplification and compression [15], mesh reduction and computer-aided design [12], and animation [14, 18, 19].

Mesh decomposition methods can be based on pure geometric properties or semantic features. In the former case, it may be necessary to segment polygon meshes into a number of patches, while in the latter case, the segmentation process attempts to translate raw data into higher level descriptions [7]. Katz and Tal [8] proposed an algorithm for hierarchically decomposing meshes. However, for each node in the hierarchy, the number of patches must be determined first, and some termination conditions are needed. Subsequently, Katz *et al.* [9] introduced a multi-dimensional scaling representation that can extract the prominent feature points and core components of a mesh object. They demonstrated hierarchical segmentation of meshes at different levels. In [10], Li *et al.*

Received April 25, 2008; revised May 1, 2008; accepted May 8, 2008.
Communicated by Tsan-sheng Hsu.
* The preliminary version has been presented in IEEE International Workshop on Multimedia Signal Processing (MMSP 2007), Chania, Crete, Greece, October 2007.

employed edge contraction and space sweeping to automatically decompose an object. However, several restrictions are needed to define the geometric and topological functions. In [11], Liu and Zhang applied spectral clustering to 3-D mesh segmentation, and found that the sparsity of an affinity matrix and the precision of the eigenvectors can influence the segmentation result. Mangan and Whitaker [12] modified the watershed algorithm to partition 3-D surface meshes into regions, but the result is sensitive to a user-defined threshold for merging regions. In [13], Zhou and Huang presented a mesh decomposition process based on critical points. Information about the critical points is the stop condition for decomposition, which is similar to Li *et al.*'s approach [10]. In [14], Lee *et al.* proposed a method to segment a deforming mesh into several near-rigid submeshes which are potentially helpful to a variety of applications such as LOD of an animation mesh and deformation transfer. However, their partitioned result is vulnerable to over-segmentation while segmenting a mesh that has large degree of deformation. In [1], Lin *et al.* proposed a mesh-decomposition scheme called "visual salience-guided mesh decomposition" that can accurately extract significant components from 3-D meshes; however, a number of thresholds have to be pre-determined by the user. Most of the previous mesh decomposition approaches except [1] are bottom-up approaches. Therefore, geometrical features such as surface normal are commonly adopted to group surface patches with similar orientations together. With a bottom-up decomposition strategy, it is difficult to come up a systematic way for determining the main body of an arbitrary 3-D object.

In Lin *et al.*'s work [1], a visual salience-guided mesh decomposition scheme is proposed. However, one of the major issues of their scheme, *i.e.*, a systematic way for determining the exact boundaries of the main body was unsolved. In this work, we use PCA to transform a pre-defined 4-D vector into a new coordinate frame. In the new coordinate frame, we can easily observe that the domain of protrusion degree always ranges from 0 to a value that is less than or equal to 1 for every component (or part). Since the zero value of protrusion degree of a part corresponds to the root of the main body along a specific direction, we can thus determine the main body by performing Boolean operations on the domains corresponding to those constituent parts.

Generally speaking, using only the 3-D world coordinates of vertices in the mesh decomposition process cannot reflect the local structure of an object, whereas, the value that quantifies the degree of protrusion of every vertex on a mesh object reflects the local property rather well. However, a protrusion degree is not distinguishable if other points on the same object have the same protrusion degree. To take advantage of both types of feature, *i.e.*, the 3-D coordinates and the protrusion degree of a vertex, we propose using them simultaneously in a mesh decomposition process. Since the two features are correlated, it is hard to characterize them in the original coordinate system.

In physics, it is not easy to distinguish the energy difference of particles in the time domain. However, when the coordinate system is transformed into a frequency domain, the energy difference of the particles can be easily identified because the frequency is proportional to the energy. In this work, we combine the 3-D coordinates and the protrusion degrees of the dual vertices of a 3-D mesh object to form a set of 4-D feature vectors. A hybrid 4-D feature vector includes the absolute position of a dual vertex in the world coordinate and the vertex's protrusion degree, which indicates the normalized distance between the dual vertex and the object's center point. To decompose a mesh, we first

perform PCA on the set of 4-D feature vectors derived from a 3-D mesh-based object. After the PCA transformation, we take the top three principal components as the three axes of a new coordinate system and observe the set of transformed feature vectors in the new coordinate system. Surprisingly, we find that the projected data along the first axis reveals the salient structures of the 3-D object. Therefore, using the first component axis as the search basis, we can identify all the salient components of an arbitrary 3-D object. We then apply some Boolean operations to precisely separate the salient components of the 3-D object from its main body.

The remainder of the paper is organized as follows. In section 2, we describe the proposed approach in detail. Section 3 contains the experiment results. Then, in section 4, we present our conclusions.

2. MESH DECOMPOSITION BY PCA

Fig. 1 shows the flowchart of our approach. First, in section 2.1, we use an approximate Laplacian operator (the umbrella operator) to adjust the positions of polygon vertices iteratively so that the meshes' lumpy vertices can be smoothed. Next, we form a 4-D feature vector for every mesh and apply PCA to the set of 4-D feature vectors that correspond to the dual vertices of a 3-D mesh-based object. Since a 4-D feature vector is composed of the 3-D coordinates and the protrusion degree of a dual vertex, in section 2.2, we explain how the normalized protrusion degree of a dual vertex is calculated. Then, in section 2.3, we describe how to perform mesh decomposition using PCA. And, in section 2.4, the salient parts are labeled. Finally, in section 2.5, we apply some Boolean operations to precisely separate the salient components of the 3-D object from its main body.

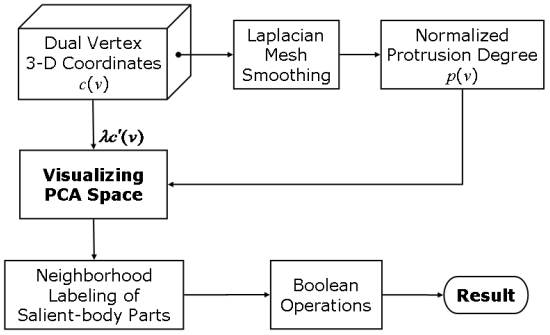


Fig. 1. Flowchart of our approach.

2.1 Nonlinear Mesh Smoothing

In general, a 3-D object can be constructed with a set of triangular polygon meshes, but some resultant polygon surfaces of the object may look ragged due to noise or inaccurate approximation [1]. To decompose a 3-D mesh-based object into a number of meaningful components, the ragged areas must be smoothed in advance. Thus, a pre-process

of polygon mesh smoothing is indispensable before mesh decomposition can be performed.

In the field of image processing, mean filtering is a simple yet powerful process used to remove noise. In addition, because of its special design, the filter retains useful information, such as edges. Thus, in this paper we use a mean filter to nonlinearly adjust the positions of polygon vertices iteratively so that most of the uneven faces of a mesh object can be gradually smoothed. An iterative process for nonlinear mesh smoothing can be defined as follows [2]:

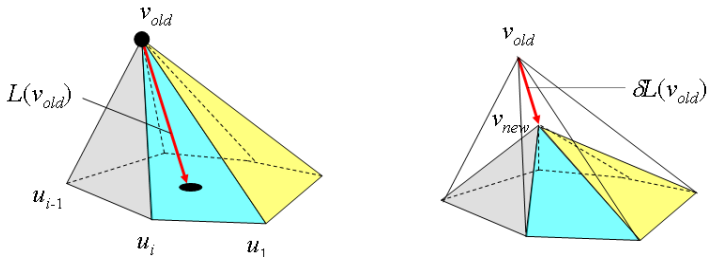
$$v_{new} \leftarrow v_{old} + \delta \cdot L(v_{old}) \tag{1}$$

where $L(v_{old})$ is a displacement vector of v_{old} (Fig. 2 (a)), and δ is a user-defined ratio for each step that can control the rate of convergence (Fig. 2 (b)). A simple formula of $L(v)$ can be defined by the following umbrella operator, which approximates the Laplacian operator [3]:

$$L(v) = \frac{1}{n} \sum_{i \in N(v)} (u_i - v) \tag{2}$$

where v is a central vertex of $N(v)$, and $N(v) = \{u_1, u_2, \ldots, u_n\}$ is the set of neighboring vertices of v. As shown in Fig. 2, when Eq. (1) is applied, the original position of a mesh vertex v_{old} is moved to a new position, v_{new}.

Using this simple procedure, the positions of some lumpy meshes can be smoothed so that the protrusion degrees of some rough surface meshes will be more consistent with those of their neighbors.



(a) (b)

Fig. 2. An example of the nonlinear mesh smoothing process.

2.2 How to Compute the Normalized Protrusion Degree of a Dual Vertex

The protrusion degree is defined as the distance between part of an object and the object's center. To calculate the protrusion degree, we adopt the normalized protrusion degree proposed by Hilaga *et al.* [4] because their method is the most suitable for protrusion characterization. Usually, the protrusion degree of a given point v on a surface S in continuous form is defined as follows [4]:

$$\mu(v) = \int_{p \in S} g(v, p) dS \tag{3}$$

where $g(v, p)$ represents the geodesic distance between two points, v and p, on S. The continuous integral function $\mu(v)$ is defined as the sum of the geodesic distances from a point v to all the other points on S. In practice, an integral function, such as that defined in [1], is constructed on the dual graph of a given 3-D mesh, $G = (V, E)$, where V and E represent the sets of dual vertices and edges, respectively. A dual vertex $v \in V$ is called the *center-of-mass* of a face in the original mesh, while a dual edge $(u, v) \in E$ links the *center-of-mass* of two adjacent faces and intersects at the midpoint of the boundary shared by the two faces. As a result, the original meshes are segmented into smaller patches, called base patches, of approximately equal size [1]. Each base patch is represented by a single dual vertex, b_i, located at the approximate center of the patch.

Therefore, the protrusion degree of a dual vertex v in discrete form can be defined as follows [4]:

$$\mu(v) = \sum_{i=1}^{k} g(v, b_i) \cdot area(P_i) \tag{4}$$

where b_1, b_2, …, and b_k are the base dual-vertices representing the base patches P_1, P_2, …, and P_k, respectively; and k is the total number of base patches. In addition, $area(P_i)$ is the area of the entire base patch and $g(v, b_i)$ denotes the geodesic distance between a dual vertex v and a base vertex b_i. Since the function $\mu(v)$ defined in Eq. (4) is not scaling invariant, a normalized version of $\mu(v)$ is therefore defined as follows [4]

$$p(v) = \frac{\mu(v) - \min_{u \in V} \mu(u)}{\max_{u \in V} \mu(u)}. \tag{5}$$

Using Eq. (5), we can calculate the normalized protrusion degree (in the range 0 to 1) of each dual vertex located on a 3-D mesh.

2.3 PCA Transformation of 4-D Composite Feature Vectors of 3-D Mesh Objects

In section 2.2, we defined the normalized protrusion degree, $p(v)$, of a dual vertex of a 3-D mesh. Thus, when combining $p(v)$ with the coordinate of a dual vertex v, $c(v) = [x(v), y(v), z(v)]$, we need to normalize all the coordinate's components, $x(v)$, $y(v)$, and $z(v)$, as follows:

$$x'(v) = \frac{x(v) - \min_{u \in V} x(v)}{\max_{u \in V} x(v) - \min_{u \in V} x(v)} \tag{6}$$

$$y'(v) = \frac{y(v) - \min_{u \in V} y(v)}{\max_{u \in V} y(v) - \min_{u \in V} y(v)} \tag{7}$$

and

$$z'(v) = \frac{z(v) - \min_{u \in V} z(v)}{\max_{u \in V} z(v) - \min_{u \in V} z(v)}. \tag{8}$$

Since the normalized version of $c(v)$ is $c'(v) = [x'(v)\ y'(v)\ z'(v)]$, let $X = \{X(v) \mid X(v) = [\lambda c'(v)\ p(v)] \in R^4, v \in V\}$, where λ is a weighted value ($0 \le \lambda \le 1$) and $p(v)$ is the normalized protrusion degree of a dual vertex v.

Surprisingly, after applying the PCA transform [5, 6] to the composite 4-D feature vectors X, we find that the distribution of their magnitudes and signs along the direction of the first component reveals the salient structures of a 3-D mesh object when we project the data onto the first three principal axes. The first principal component is the most important feature because it reflects the salient characteristics of the parts. After all parts of the body have been labeled, the main body and its salient parts can be separated by applying some Boolean operations. Next, we describe how to identify the salient parts based on the results of PCA transformation.

2.4 Neighborhood Labeling of Salient Parts

To separate the main body of a 3-D mesh object from its salient parts, we observe the transformed data spread along the first three principal axes. Fig. 3 shows a Dinopet in which the first three principal axes have been PCA-transformed. It is obvious that the transformed data points converge along the axis of the first principal component. Similar to a diffusion process, each salient component of the 3-D mesh can be obtained by using a simple neighborhood labeling method to search along the first principal component from the local maximum to zero. The neighborhood labeling process is described below.

Let L represent the labeling operation of a triangle that covers a dual vertex v, and S^x be the neighborhood vertices of a vertex x. $S^x = \{v \mid \forall v \in L(x), \textit{iff}\ \mathrm{c1}(v) \neq 0\}$ for $x \in \{v_i \mid \forall v_i \in V\}$ where V is a sequence which is sorted by the value of the first component of the dual vertex v_i, namely $\mathrm{c1}(v_i)$, in descending order. Then, for each salient part k, S_k is the set of vertices in the diffusion process S^x. The diffusion process repeats recursively until $\mathrm{c1}(v_i)$ is equal to zero. The procedure is repeated until other salient parts are labeled. For the case of Fig. 3, the number of times of repeated process is six, *i.e.* $k = 6$.

By setting the minimum value to zero, we can guarantee that a salient part and the main body of an object will be covered simultaneously. Fig. 4 shows the neighborhood labeling results of the Dinopet example in Fig. 3; a total of six salient parts are located in the PCA transformed space. These six parts correspond, respectively, to the arms, feet, head, and tail of the Dinopet. The six neighborhood labeling results are shown in Figs. 4 (a)-(f) respectively.

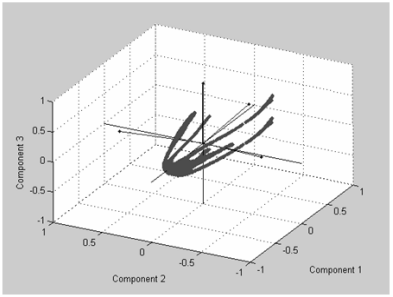


Fig. 3. A Dinopet: PCA transformation of the first three principal axes.

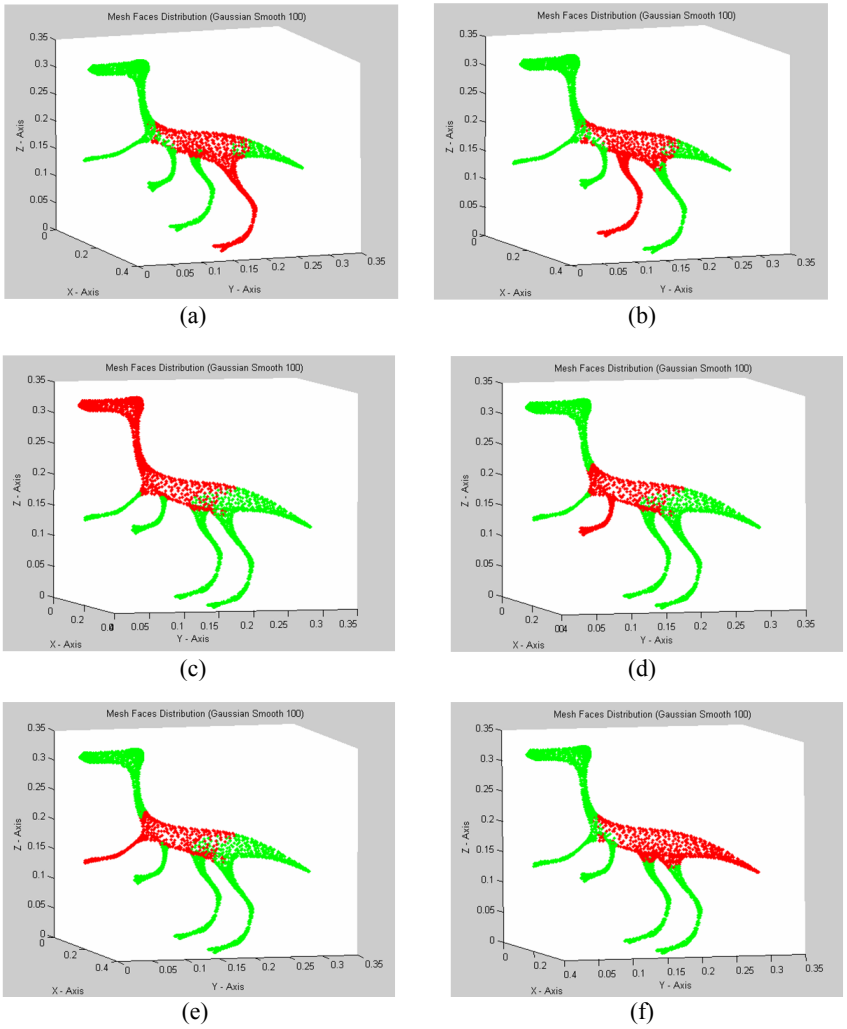


Fig. 4. The neighborhood labeling results of the Dinopet example in Fig. 3.

2.5 Differentiation of the Main Body and the Salient Parts

In the previous section, we described how to derive the salient parts of a 3-D object through a neighborhood labeling process. However, we found that the process labeled the main body of the object as well as its salient parts. To clearly separate the salient parts from the main body, we need to know the exact boundary of the body. Therefore, we developed a method to extract the main body from all the labeled neighborhoods. For simplicity, we use an object (G) composed of three salient parts (P_A, P_B, and P_C) and the main body (M) as an example to explain how our method works (Fig. 5). Assume the light blue, light orange and light yellow ellipsoids represent A, B, and C, respectively. The three ellipsoids represent three labeled neighborhoods. The overlapping parts between

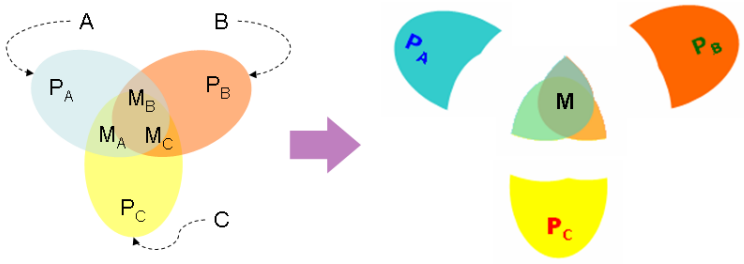


Fig. 5. The main body and the three salient parts are clearly defined after applying Boolean operations.

the neighborhoods and the main body are M_A, M_B, and M_C, respectively. In summary, $A = P_A + M_A$, $B = P_B + M_B$, $C = P_C + M_C$, $G = A + B + C$, and $M = M_A + M_B + M_C$.

According to the truth table of XNOR logic gate, A HIGH output (1) comes out if both of the inputs to the gate are the same. If one but not both inputs is HIGH (1), a LOW output (0) comes out. Based on the characteristics of XNOR logic gate, we can find the common part shared by the main body (M) and the object (G), *i.e.* G⊙M. Here, "⊙" represents an "XNOR" operation.

$$
\begin{aligned}
G\odot M &= (\overline{A+B+C})\overline{M}+(A+B+C)M \\
&= \overline{A}\cdot\overline{B}\cdot\overline{C}\cdot\overline{M}+A\cdot M+B\cdot M+C\cdot M \\
&= \overline{A}\cdot\overline{B}\cdot\overline{C}\cdot\overline{M}+A\cdot M(1+\overline{B}\cdot\overline{C})+B\cdot M(1+\overline{A}\cdot\overline{C})+C\cdot M(1+\overline{A}\,\overline{B}) \\
&= \overline{A}\cdot\overline{M}\cdot\overline{B}\cdot\overline{C}+A\cdot M\cdot\overline{B}\cdot\overline{C}+B\cdot M\cdot\overline{A}\cdot\overline{C}+C\cdot M\cdot\overline{A}\cdot\overline{B}+A\cdot M+B\cdot M+C\cdot M \\
&= (A\odot M)\cdot\overline{B}\cdot\overline{C}+(B\odot M)\cdot\overline{A}\cdot\overline{C}+(C\odot M)\cdot\overline{A}\cdot\overline{B}+(A+B+C)\cdot M \\
\Rightarrow\ \overline{G\overline{M}} &= (A\odot M)\cdot\overline{B}\cdot\overline{C}+(B\odot M)\cdot\overline{A}\cdot\overline{C}+(C\odot M)\cdot\overline{A}\cdot\overline{B} \\
\Rightarrow \overline{G\overline{M}}\cdot M &= \left[(A\odot M)\cdot\overline{B}\cdot\overline{C}+(B\odot M)\cdot\overline{A}\cdot\overline{C}+(C\odot M)\cdot\overline{A}\cdot\overline{B}\right]\cdot M \\
\Rightarrow\ 0 &= A\cdot M\cdot\overline{B}\cdot\overline{C}+B\cdot M\cdot\overline{A}\cdot\overline{C}+C\cdot M\cdot\overline{A}\cdot\overline{B} \\
\Rightarrow\ 0 &= M\cdot(A\cdot\overline{B}\cdot\overline{C}+\overline{A}\cdot B\cdot\overline{C}+\overline{A}\cdot\overline{B}\cdot C) \\
\Rightarrow\ M\cdot\overline{M} &= M\cdot(A\cdot\overline{B}\cdot\overline{C}+\overline{A}\cdot B\cdot\overline{C}+\overline{A}\cdot\overline{B}\cdot C) \\
\Rightarrow\ \overline{M} &= A\cdot\overline{B}\cdot\overline{C}+\overline{A}\cdot B\cdot\overline{C}+\overline{A}\cdot\overline{B}\cdot C \\
\Rightarrow\ M &= \overline{A\cdot\overline{B}\cdot\overline{C}+\overline{A}\cdot B\cdot\overline{C}+\overline{A}\cdot\overline{B}\cdot C} \\
&= \overline{A\cdot\overline{B}\cdot\overline{C}}\cdot\overline{\overline{A}\cdot B\cdot\overline{C}}\cdot\overline{\overline{A}\cdot\overline{B}\cdot C} \\
&= (\overline{A}+B+C)\cdot(A+\overline{B}+C)\cdot(A+B+\overline{C}) \\
&= \overline{A}\cdot\overline{B}\cdot\overline{C}+A\cdot B+A\cdot C+B\cdot C \\
&= \overline{G}+A\cdot B+A\cdot C+B\cdot C \\
&= A\cdot B+A\cdot C+B\cdot C
\end{aligned}
\tag{9}
$$

In Eq. (9), the main body (M) can be derived by $M = A \cdot B + A \cdot C + B \cdot C$. The inverse of M, *i.e.*, $\overline{M}$, is equal to

$$\overline{M} = A\cdot\overline{B}\cdot\overline{C}+\overline{A}\cdot B\cdot\overline{C}+\overline{A}\cdot\overline{B}\cdot C. \tag{10}$$

To derive the clean salient part A, we apply the "AND" operation between A and $\overline{M}$. The salient part A is derived as follows:

$$A \cdot \overline{M} = (P_A + M_A) \cdot \overline{M} = P_A \cdot \overline{M} + M_A \cdot \overline{M} = P_A + \phi = P_A. \tag{11}$$

Substituting Eq. (10) into Eq. (11), the salient part $A(P_A)$ can be determined by

$$P_A = A \cdot (A \cdot \overline{B} \cdot \overline{C} + \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot \overline{B} \cdot C) = A \cdot \overline{B} \cdot \overline{C} \tag{12}$$

Similarly, P_B and P_C are equal to $\overline{A} \cdot B \cdot \overline{C}$ and $\overline{A} \cdot \overline{B} \cdot C$, respectively. Although the example only shows three salient parts, for objects with more salient parts, the main body and salient parts can be derived by using extended versions of Eqs. (9) and (12).

We use the Dinopet mesh-based object to show how the complete automatic mesh decomposition procedure works. Using an extended version of Eq. (9), we can determine the main body of the Dinopet model from its six labeled salient parts (Fig. 4) by applying the following series of Boolean operations:

$$\begin{aligned} M = {} & A \cdot B + A \cdot C + A \cdot D + A \cdot E + A \cdot F + \\ & B \cdot C + B \cdot D + B \cdot E + B \cdot F + \\ & C \cdot D + C \cdot E + C \cdot F + D \cdot E + D \cdot F + E \cdot F \end{aligned} \tag{13}$$

where A, B, C, D, E, and F are the six labeled neighborhoods shown in Fig. 4. Using Eq. (13), we can determine the main body of the Dinopet model, as shown in Fig. 6 (the part in red). Once the main body has been determined, the salient part $A(P_A)$ of the Dinopet model can be derived by the following composite Boolean operation:

$$P_A = A \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} \cdot \overline{E} \cdot \overline{F}. \tag{14}$$

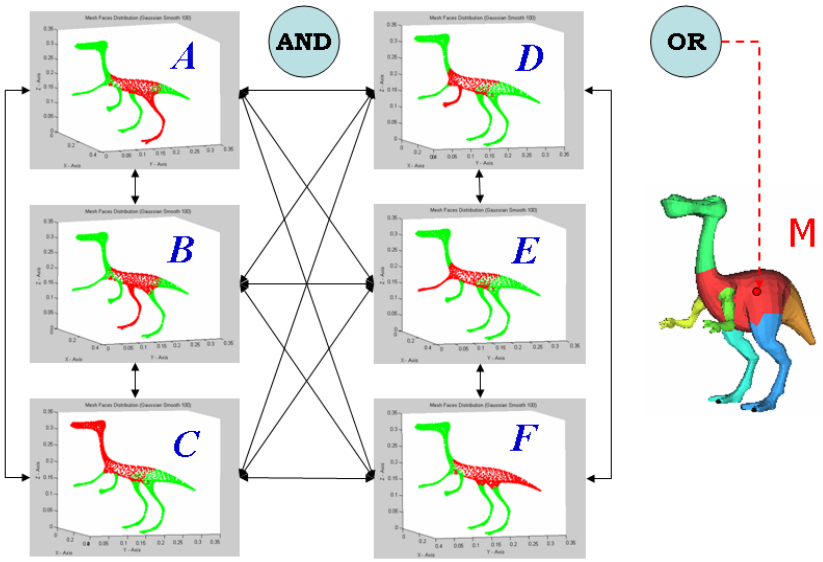


Fig. 6. The main body of the Dinopet polygon meshes (red part) can be determined by applying Boolean operations on labeling results of the six neighborhoods.

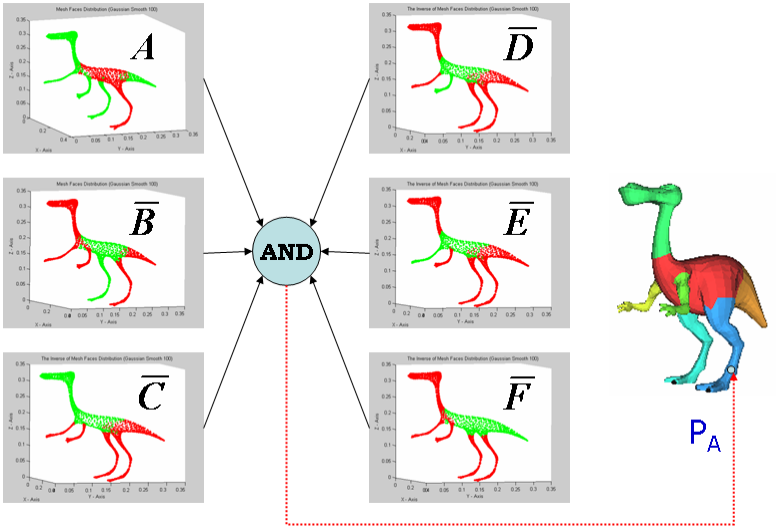


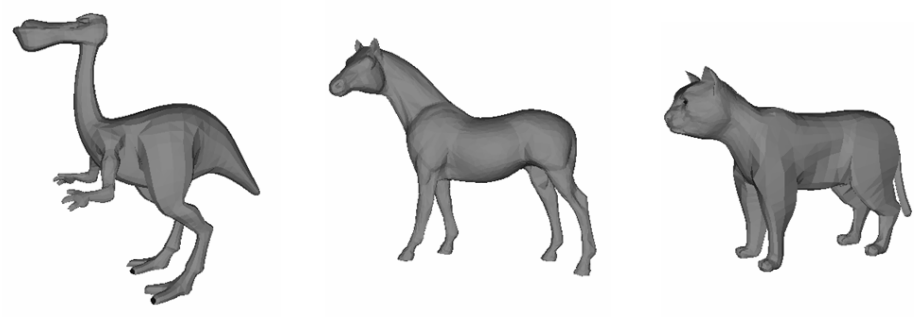
Fig. 7. The salient part P_A of the Dinopet polygon meshes (blue part) can be derived by applying the "AND" operation on A, $\overline{B}$, $\overline{C}$, $\overline{D}$, $\overline{E}$, and $\overline{F}$.

Using Eq. (14), the salient part A of the Dinopet model can be determined as shown in Fig. 7 (the part in blue). The other five salient parts can be determined similarly.

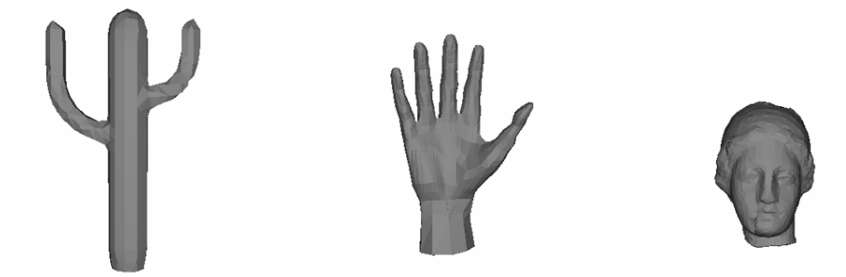
With the proposed measures for the degree of protrusion and PCA transformation, the significant components of an arbitrary 3-D mesh can be identified and extracted by executing some Boolean operations. In terms of efficiency, the total complexity of the proposed method is $O(|V|\log|V|)$. The protrusion degree characterization process can be computed in $O(|V|\log|V|)$ [4]. Since we perform PCA on the set of 4-D feature vectors derived from a 3-D mesh-based object, the computational complexity of PCA algorithm is $O(4^2|V|)$. In addition, the process for Laplacian mesh smoothing costs $O(|V|)$. The process for neighborhood labeling can be calculated in $O(K|V|)$, and the process for Boolean operation costs $O(K)$, where K is a constant which stands for the number of salient parts of a mesh-based object. In [1], Lin *et al.* proposed a visual salience-guided mesh decomposition scheme. The complexity of their algorithm is $O(|V|^2\log|V|)$.

3. EXPERIMENT RESULTS

In the experiments, we used six 3-D mesh-based objects to test the effectiveness of our approach. The six test models are a Dinopet (Fig. 8 (a)), a Horse (Fig. 8 (b)), a Big Cat (Fig. 8 (c)), a Cactus (Fig. 8 (d)), a Palm (Fig. 8 (e)), and a Venus Head (Fig. 8 (f)). Figs. 9 (a)-(f) show the distribution of the protrusion degrees of the smoothed mesh objects in Figs. 8 (a)-(f), respectively. In the first step of Laplacian mesh smoothing, we used a mean filter to adjust the position of a polygon vertex iteratively. The parameter of the user-defined ratio for each step, δ, was consistently set to 0.5, and the number of iterations of the smoothing process for each example was fixed at 100. Note that the larger the number of iterations, the more easily the salient part can be distinguished. However, a

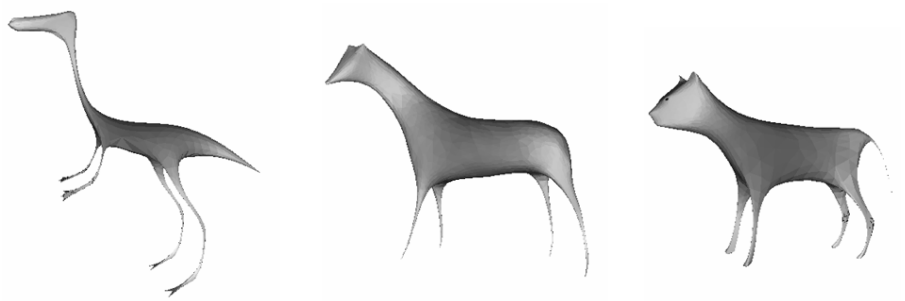


(a) Dinopet. (b) Horse. (c) Big cat.

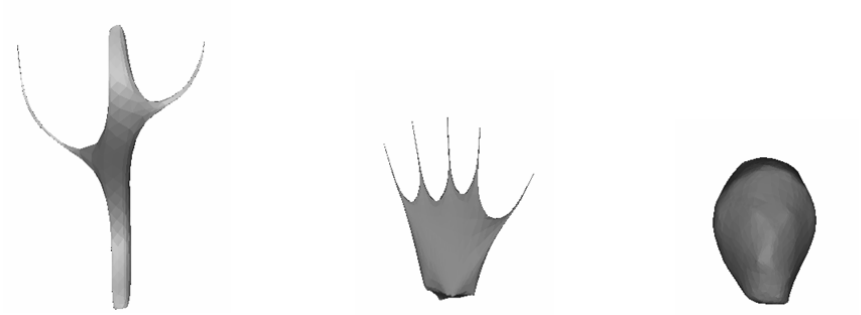


(d) Cactus. (e) Palm. (f) Venus head.

Fig. 8. The six test models used in our experiments.



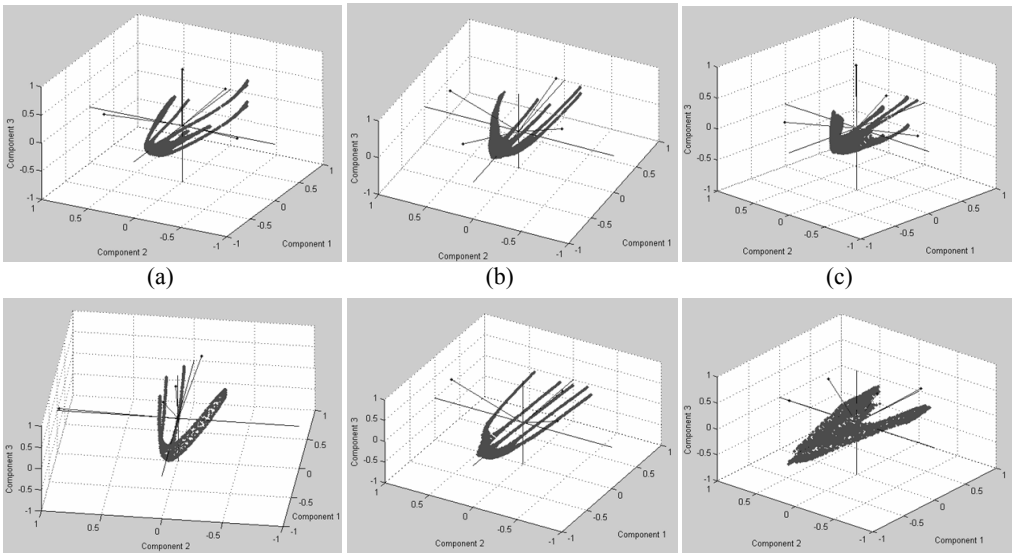
(a) Dinopet. (b) Horse. (c) Big cat.



(d) Cactus. (e) Palm. (f) Venus head.

Fig. 9. Distribution of the protrusion degrees of six smoothed 3-D polygon meshes.

polygon mesh model not only becomes deformed, but also shrinks after the recursive smoothing process is applied. Therefore, the number of iterations has to be confined. To balance the degree of importance of the 3-D coordinates and the protrusion degree, we set the weighting factor λ of the composite 4-D feature vector X at one third. Figs. 10 (a)-(f) show, respectively, the data obtained from the first three principal components after performing PCA transformation on the six test models. Table 1 lists the cumulative percentages of the degree of representation in the first three principal components. From the table, we observe that the projections on the first three principal axes already approximate at least 90% of the original data. However, the boundaries between decomposed salient components and the main body are usually saw-toothed, as some meshes are overlapped in the neighborhood labeling process. Therefore, we applied Dijkstra's algorithm as a post-processing step to determine the shortest path on a graph of the boundaries.



(a) (b) (c)

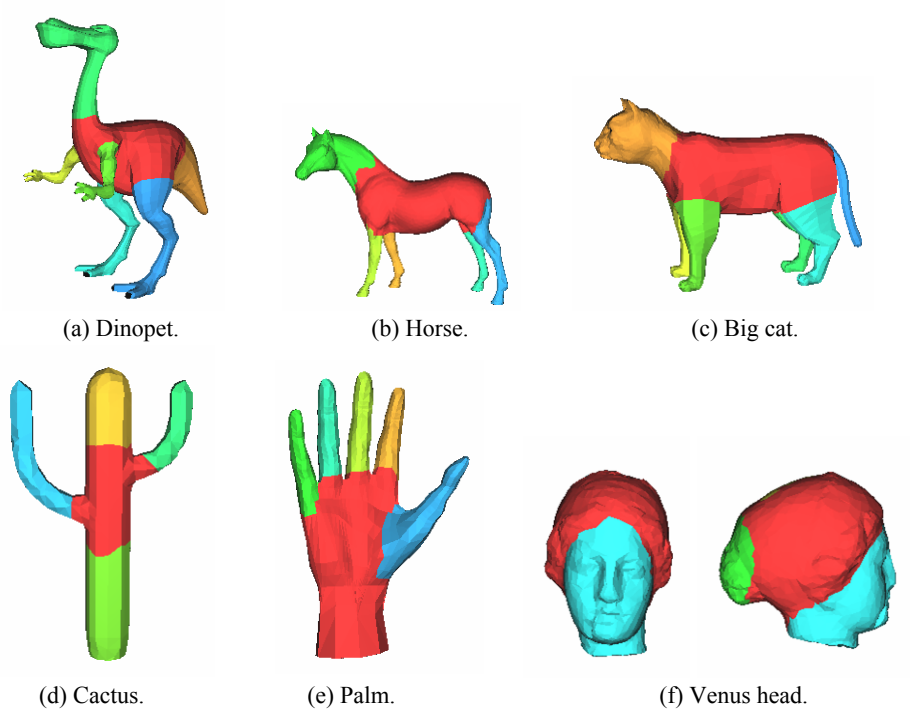
(d) (e) (f)

Fig. 10. Visualization of the projections on the first three principal components of: (a) a dinopet, (b) a horse, (c) a big cat, (d) a cactus, (e) a palm, and (f) a Venus head.

Table 1. The cumulative percentages of the degree of representation in the first three principal components.

Mesh Object	Component 1	Component 1 + Component 2	Component 1 + Component 2 + Component 3
Dinopet	74.9708%	89.8334%	97.5543%
Horse	71.8391%	90.7804%	96.2159%
Big Cat	63.8539%	80.5739%	94.7240%
Cactus	73.2575%	83.5216%	93.1713%
Palm	86.1423%	94.9363%	98.4768%
Venus Head	68.9752%	80.8511%	90.8861%

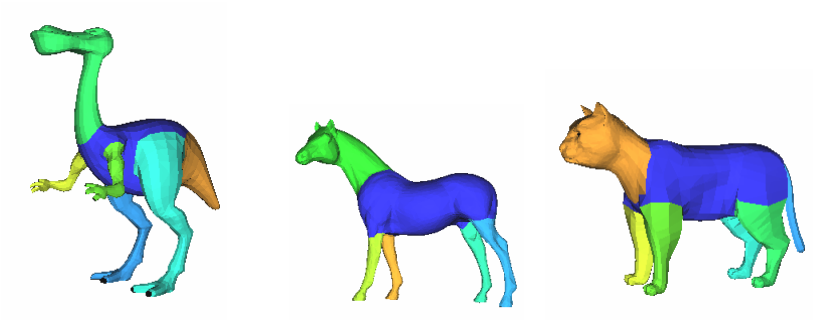
The decomposition results of our approach and Lin *et al.*'s approach [1], are respectively shown in Figs. 11 and 12. In Lin *et al.*'s work [1], they failed to decompose the Venus Head model (Fig. 12 (f)) since the head model contains less protrusive features. As shown in Fig. 10 (f), the transformed data points of Venus Head model still converge along the axis of the first principal component. Fig. 11 (f) indicated that our result is better than that of Lin *et al.*'s method [1] (shown in Fig. 12 (f)). Moreover, as to the comparison with [1], our proposed approach is now able to "systematically" determine the "territory" of the main body of an object. However, the method proposed in [1] cannot do the same.



(a) Dinopet. (b) Horse. (c) Big cat.

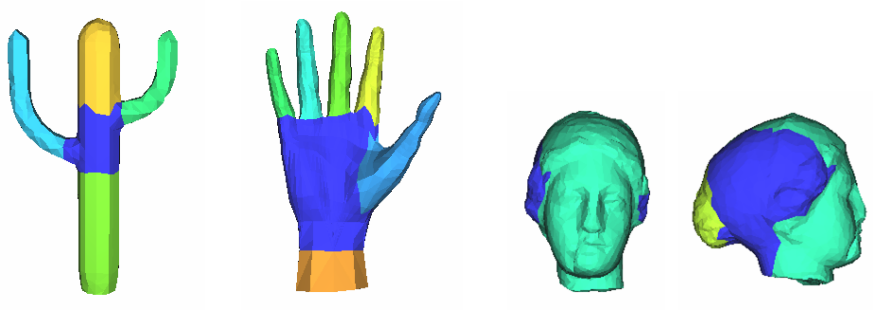
(d) Cactus. (e) Palm. (f) Venus head.

Fig. 11. Decomposition results of our approach.



(a) Dinopet. (b) Horse. (c) Big cat.

Fig. 12. Decomposition results of Lin *et al.*'s approach [1].



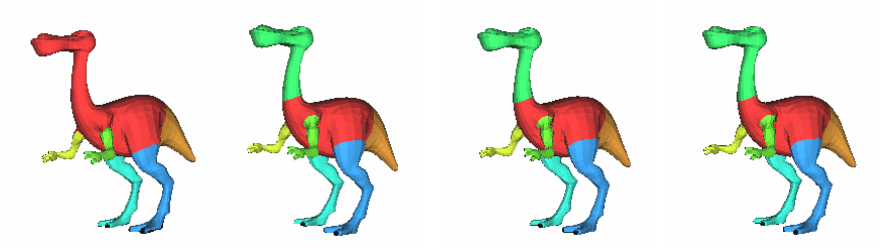
(d) Cactus. (e) Palm. (f) Venus head.

Fig. 12. (Cont'd) Decomposition results of Lin *et al*.'s approach [1].

Strength of the proposed method is that it requires only few parameters and the parameters are all the same for the models shown in the experiments. However, choosing inappropriate weighting factor λ for the composite 4-D feature vector X may affect the decomposition result. Particularly, the first principal component is the most important feature because it reflects the salient characteristics of the parts. Table 2 lists the cumulative percentages of the degree of representation in the first three principal components on the Dinopet model with different factor λ. From the table, we observe that the projection on the first principal axis is only 44.9424% of the original data when λ is equal to 1, and the decomposition result was failed as shown in Fig. 13 (a).

Table 2. The cumulative percentages of the degree of representation in the first three principal components on the Dinopet Model with different factor λ.

Mesh Object	λ	Component 1	Component 1 + Component 2	Component 1 + Component 1 + Component 3
Dinopet	1	44.9424%	70.4799%	93.2395%
Dinopet	1/2	57.5293%	82.7140%	95.8670%
Dinopet	1/3	74.9708%	89.8334%	97.5543%
Dinopet	1/4	84.1252%	93.5540%	98.4462%



(a) $\lambda = 1$. (b) $\lambda = 1/2$. (c) $\lambda = 1/3$. (d) $\lambda = 1/4$.

Fig. 13. Decomposition results of Dinopet model with different λ.

4. CONCLUSION

We have proposed a novel mesh decomposition technique based on PCA transformation and Boolean operations. First, we combine the 3-D coordinates and the protrusion degrees of the dual vertices of a 3-D mesh object to form a set of 4-D feature vectors. A hybrid 4-D feature vector comprises the absolute position of a dual vertex in the world coordinate and the protrusion degree of the vertex, which indicates the normalized distance between the dual vertex and the object's center point. Next, we perform PCA on the set of 4-D feature vectors derived from the 3-D mesh-based object. After PCA transformation, we take the top three principal components as the three axes of a new coordinate system and observe the set of transformed feature vectors in the new coordinate system. Surprisingly, we find that the projected data along the first axis reveals the salient structures of the 3-D object. Therefore, using the first component axis as the search basis, we can identify all the salient components of an arbitrary 3-D object. We then apply some Boolean operations to precisely separate the salient components of the 3-D object from its main body. The main contribution of this work is that our proposed approach is able to "systematically" determine the "territory" of the main body of an object. In other words, the proposed approach can be "completely" automatic for 3-D mesh decomposition.

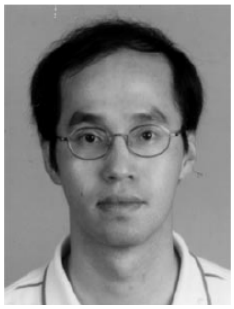
REFERENCES

1. H. Y. Lin, H. Y. M. Liao, and J. C. Lin, "Visual salience-guided mesh decomposition," *IEEE Transactions on Multimedia*, Vol. 9, 2007, pp. 47-56.
2. H. Yagou, Y. Ohtake, and A. G. Belyaev, "Mesh smoothing via mean and median filtering applied to face normals," in *Proceedings of Geometric Modeling and Processing*, 2002, pp. 124-131.
3. L. Kobbelt, S. Campagna, J. Vorsatz, and H. P. Seidel, "Interactive multi-resolution modeling on arbitrary meshes," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, 1998, pp. 105-114.
4. M. Hilaga, Y. Shingawa, T. Kohmura, and T. L. Kunii, "Topology matching for fully automatic similarity estimation of 3-D shapes," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 2001, pp. 203-212.
5. I. T. Jolliffe, *Principal Component Analysis*, 2nd ed., Springer-Verlag, New York, 2002.
6. C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, 1995.
7. M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal, "Mesh segmentation – A comparative study," in *Proceedings of IEEE International Conference on Shape Modeling and Applications*, 2006, pp. 7-18.
8. S. Katz and A. Tal, "Hierarchical mesh decomposition using fuzzy clustering and cuts," in *Proceedings of International Conference on Computer Graphics and Interactive Techniques*, 2003, pp. 954-961.
9. S. Katz, G. Leifman, and A. Tal "Mesh segmentation using feature point and core

extraction," *The Visual Computer*, Vol. 21, 2005, pp. 865-875.
10. X. Li, T. W. Woon, T. S. Tan, and Z. Huang, "Decomposing polygon meshes for interactive applications," in *Proceedings of ACM Symposium on Interactive 3-D Graphics*, 2001, pp. 35-42.
11. R. Liu and H. Zhang, "Segmentation of 3D meshes through spectral clustering," in *Proceedings of Pacific Conference on Computer Graphics and Applications*, 2004, pp. 298-305.
12. A. Mangan and R. Whitaker, "Partitioning 3D surface meshes using watershed segmentation," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, 2001, pp. 308-321.
13. Y. Zhou and Z. Huang, "Decomposing polygon meshes by means of critical points," in *Proceedings of the 10th International Multimedia Modelling Conference*, 2004, pp. 187-195.
14. 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.
15. E. Zuckerberger, A. Tal, and S. Shlafman, "Polyhedral surface decomposition with applications," *Computers and Graphics*, Vol. 26, 2002, pp. 733-743.
16. S. Schaefer and C. Yuksel, "Example-based skeleton extraction," in *Proceedings of Eurographics Symposium on Geometry Processing*, 2007, pp. 1-10.
17. S. Shlafman, A. Tal, and S. Katz, "Metamorphosis of polyhedral surfaces using decomposition," in *Proceedings of Computer Graphics Forum*, 2002, pp. 219-228.
18. 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.
19. D. L. James and C. D. Twigg, "Skinning mesh animations," *ACM Transactions on Graphics*, Vol. 24, 2005, pp. 399-407.

Jung-Shiong Chang (張俊雄) received the B.S. degree in Industrial Education from National Taiwan Normal University, Taipei, Taiwan, in 1988, and the M.S. degree in Electrical Engineering from National Chung Cheng University, Chiayi, Taiwan, in 1995, respectively.

He is currently pursuing the Ph.D. degree at the Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan. His areas of research interest include image processing, computer vision, and 3-D mesh processing.



Arthur Chun-Chieh Shih (施純傑) was born in Taipei, Taiwan, on September 26, 1966. He received the B.S. degree in Electrical Engineering from the Chinese Culture University, Taipei, in 1992, the M.S. degree also in Electrical Engineering from National Chung Cheng University, Chiayi, Taiwan, in 1994 and a Ph.D. degree in Computer Science and Information Engi-

neering from National Central University, Chung Li, Taiwan, in 1998.

From October 1998 to July 2002, he worked for the Institute of Information Science, Academia Sinica, Taiwan, and the Department of Ecology and Evolution, the University of Chicago, Chicago, IL, as a Postdoctoral Fellow. He joined the Institute of Information Science, Academia Sinica, as an Assistant Research Fellow in July 2002 and became an Associate Research Fellow in 2008. His current research interests include molecular evolution, bioinformatics, and multimedia signal processing.

Hsiao-Rong Tyan (田筱榮) received the B.S. degree in Electronic Engineering from Chung Yuan Christian University, Chungli, Taiwan, in 1984, and the M.S. and Ph.D. degrees in Computer Science from Northwestern University, Evanston, IL, in 1987 and 1992, respectively.

She is an Associate Professor of the Department of Information and Computer Engineering, Chung Yuan Christian University, Chungli, Taiwan, where she currently conducts research in the areas of computer networks, computer security, and intelligent systems.

Wen-Hsien Fang (方文賢) was born in Taipei, Taiwan, Republic of China, in 1961. He received his B.S. degree from the National Taiwan University in 1983, and his M.S.E. degree and Ph.D. degree from the University of Michigan, Ann Arbor, in 1988 and 1991, respectively, in Electrical Engineering and Computer Science.

In fall 1991, he joined the faculty of National Taiwan University of Science and Technology, where he currently holds a position as a professor in the Department of Electronic Engineering. His research interests include adaptive array processing, statistical signal processing, signal processing for wireless communications, and multimedia signal processing.