Visual Salience-Guided Mesh Decomposition

Hsueh-Yi Sean Lin, Hong-Yuan Mark Liao, *Senior Member, IEEE*, and Ja-Chen Lin

***Abstract*—In this paper, we propose a novel mesh-decomposition scheme called "visual salience-guided mesh decomposition." The concept of "part salience," which originated in cognitive psychology, asserts that the salience of a part can be determined by (at least) three factors: the protrusion, the boundary strength, and the relative size of the part. We try to convert these conceptual rules into real computational processes, and use them to guide a three-dimensional (3-D) mesh decomposition process in such a way that the *significant components* can be precisely identified and efficiently extracted from a given 3-D mesh. The proposed decomposition scheme not only identifies the parts' boundaries defined by the minima rule, but also labels each part with a quantitative degree of visual salience during the mesh decomposition process. The experimental results show that the proposed scheme is indeed effective and powerful in decomposing a 3-D mesh into its significant components.**

***Index Terms*—Mesh decomposition, perceptual organization, visual salience.**

I. Introduction

DECOMPOSITION is a leverage to obtain the componential representation of a whole object. After the decomposition step is executed, the decomposed components can be individually selected, grouped, and analyzed based on the properties of interest. In recent years, a variety of applications have benefited from decomposing a three-dimensional (3-D) object into its component parts. These applications include collision detection [21], radiosity simulations [9], robust transmission and streaming [2], [3], texture mapping [20], [34], metamorphosis [33], simplification and compression [40], 3-D shape retrieval [7], [40], and control-skeleton extraction for key-frame animation [15]. The requirements that an effective object decomposition method has to satisfy usually depend on the application. In this study, we emphasize high-level abstraction and organization for human object understanding.

High-level organization imposed on perceived data has been explored extensively in both human visual processes and computer vision systems. Related studies can be found in [1], [11], [12], [28], [30]. In order to account for human visual processes, cognitive psychologists have identified a set of properties (or rules) that are fairly important in the perception of a form or a shape. In the field of computer vision, perceptual organization has shown that computational resources can be effectively applied to extract structural and meaningful organization from perceived data. Furthermore, perceptual organization can be regarded as an intelligent process that can perform high-level abstraction for image understanding. In the literature, perceptual organization has been applied to the segmentation of range images [6], two-dimensional (2-D) images [31], textures [35], patterns [38], and contours [39], respectively. However, for some reason, existing 3-D object decomposition techniques lack a direct link to perceptual organization. Below, we briefly review some existing 3-D object decomposition techniques and specify their possible links to perceptual organization. Furthermore, we point out their common shortcomings in terms of perceptual organization.

In [36], Wu and Levine introduced a simulated electrical charge distribution scheme to perform surface characterization. Using their representation, the concave boundaries can be located at the minima of the local charge density. In [24], Mangan and Whitaker extended the watershed algorithm, which was originally designed for image segmentation [27], to partition both volumetric and mesh surfaces. In [21], Li *et al.* employed skeletonization and space sweeping procedures to extract organic parts. In [29], Svensson and Baja introduced the concept of distance transform to decompose 3-D volumetric objects into kernels and elongated parts. Shlafman *et al.* [33] proposed a *k*-means based clustering algorithm that separates distant faces, but clusters close faces together. In [15], Katz and Tal proposed a general framework for mesh decomposition, in which a maximum-flow (minimum-cut) algorithm is applied to construct boundaries without jagged effects. In [17], Lien and Amato identified and resolved the non-convex features in order of importance to achieve approximate convex decomposition. In [19], Liu and Zhang introduced a spectral clustering method that favors segmentation along concave regions. In [16], Katz *et al.* introduced multidimensional scaling representation and spherical mirroring operation to extract prominent feature points and core component, respectively. The boundaries between the extracted features are then constructed and refined using Katz and Tal's algorithm [15].

From the perspective of perceptual organization, the underlying assumption of the above methods is based on psychologists' definitions of a part, which are regularized by "a uniformity of nature" [11]. In cognitive psychology, the principle of transversality[1] is regarded as one of the guidelines for finding a part's shapes. Among existing methods, those that trace concave regions [15], [24], [26], [36] adopt the minima rule[2] to

Manuscript received March 1, 2005; revised March 8, 2006. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zhengyou Zhang.

H.-Y. S. Lin and J.-C. Lin are with the Department of Computer Science, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: hylin@cis.nctu.edu.tw; jclin@cs.nctu.edu.tw).

H.-Y. M. Liao is with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan, R.O.C. (e-mail: liao@iis.sinica.edu.tw).

Digital Object Identifier 10.1109/TMM.2006.886344

[1]Transversality regularity: When two arbitrarily shaped surfaces are made to interpenetrate, they always meet in a contour of concave discontinuity of their tangent planes. For a detailed discussion of transversality, please refer to [8].

[2]Minima rule: All negative minima of the principal curvatures (along their associated lines of curvature) form boundaries between parts. For a detailed discussion of the minima rule, please refer to [12].

construct a part's boundaries. With regard to clustering-based methods [15], [33] and merging processes [24], [29], the Gestalt law of similarity and proximity is frequently adopted to capture the homogeneous characteristics of parts. Obviously, some existing 3-D object decomposition methods do attempt to extract a part's shapes and boundaries by mimicking human visual perception of 3-D shapes. However, none of the existing methods takes the salience of parts into account. According to the findings of cognitive psychologists [12], the salience of parts usually plays an important role in the 2-D silhouette and 3-D shape partitioning processes.

In this paper, we propose a novel mesh decomposition scheme called "visual salience-guided mesh decomposition," which bases decomposition on the theory of part salience borrowed from cognitive psychology [12]. The theory asserts that the salience of a part is usually determined by three factors: the protrusion, the boundary strength, and the relative size of the part. The computational processes designed for realizing two of these salient features are presented in Section III. We use these features to guide the decomposition process so that the visually significant components can be extracted from a given 3-D mesh. This new approach has a number of potential applications. For example, in 3-D shape databases, the organization of each object should be in accordance with our visual perception. Specifically, the organization from the parts to the whole would allow us to conduct a "part-in-whole" search process (as in [7]). In addition, extracting significant components based on different salient features would allow us to construct an efficient and valid set of visual parts from a 3-D model. In this way, one could realize "*query-by-significant-components*" in a 3-D shape retrieval system.

The remainder of this paper is organized as follows. In Section II, we introduce the theory of part salience and illustrate its importance in the perception of parts. In Section III, we propose the computational processes for realizing the qualitative salient features, and describe in detail how to incorporate visual salience into the mesh decomposition process. The experimental results are presented in Section IV. Finally, in Section V, we present our conclusions.

II. Review of Hoffman and Singh's Theory of Part Salience

In [12], Hoffman and Singh proposed the theory of part salience, which states that at least three factors determine the salience of a part: the protrusion, the boundary strength, and the relative size of the part. We now give the quantitative definitions of these salient factors and then describe their importance in visual processes.

A. Protrusion of a Part

This factor is the degree to which a part protrudes from its main body. For 2-D silhouettes, it can be quantified as the ratio of the perimeter of the part (excluding its bases) to the sum of its base lengths. For 3-D shapes, the base of a part is referred to as the minimal surface formed by the boundary curve of the part. Hence, the protrusion of a 3-D part can be quantified as the ratio of the area of the part's surface to the area of its base surface.

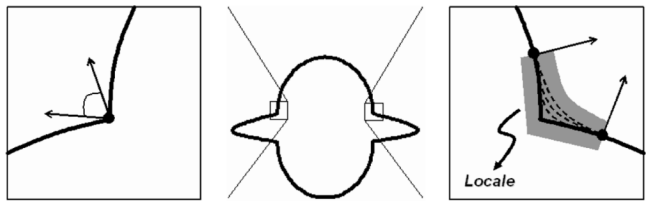


Fig. 1. Illustration of turning normals and locale turning at the boundary of a 2-D silhouette. (a) Turning normals. (b) 2-D silhouette. (c) Locale turning.

B. Strength of a Part's Boundary

According to the principle of transversality, a part's boundaries are usually located at the concave creases, as shown in Fig. 1(b). In [12], Hoffman and Singh proposed that possible quantitative definitions of the boundary strength include the turning normals and locale turning, as shown in Fig. 1(a) and (c), respectively. Obviously, the indication of the normal direction must have a global orientation consistency so that the boundary strength can be captured precisely. The discriminating capability of turning normals and locale turning is shown by the following examples. For 2-D silhouettes, two sides of a crease boundary usually have two normals, and the angle between them can, in one sense, represent the strength of that boundary. On the other hand, for potentially smooth boundaries, which are represented by the dotted lines in Fig. 1(c), there is one normal at every point along a curve. To tackle this problem, Hoffman and Singh [12] proposed obtaining the measure of turning in an appropriate region near the boundary. As shown in Fig. 1(c), the gray region is the so-called *locale*[3] and the normals on its two sides (i.e., the so-called *locale turning*) are used to characterize the strength of the smooth boundary. For 3-D shapes, the principal curvatures can be used to measure the strength of a part's boundary.

C. Relative Size of a Part

This factor indicates the size of a part relative to the whole object. For 2-D silhouettes, it can be defined as the ratio of the area of a part to the area of the whole object. For 3-D shapes, the relative volume can be used to measure a part's relative size.

Having reviewed the factors that may be used to determine the salience of a part, we now discuss their effects on both visual and decomposition processes. For simplicity, the following discussion is based on 2-D silhouettes; however, the concept can be easily extended to 3-D models. Fig. 2 shows the boundaries and cuts of parts of a 2-D silhouette, indicated by isolated points and dotted lines, respectively. Note that, in Fig. 2(a), the four boundaries are used to form possible cuts; and, in Fig. 2(b) and (c), each part is generated by exactly one cut. According to the visually salient properties of interest, a 2-D silhouette may have different interpretations. For example, the 2-D silhouette might be interpreted as an alien's head with a pair of protrusive ears when the salience of the part is determined primarily by its protrusion [(i.e., the part's cuts in Fig. 2(b)]. On the other hand, the 2-D silhouette might be interpreted as an unidentified

[3]By definition [12], a locale is an appropriate region near (but not just infinitesimally near) a negative minimum of the curvature, in which we can explore how the curve evolves.

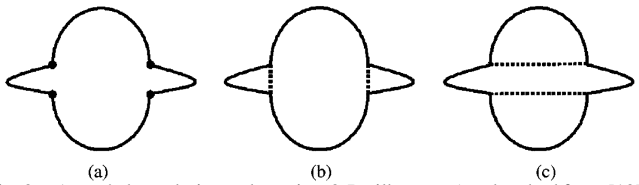


Fig. 2. A part's boundaries and cuts in a 2-D silhouette (re-sketched from [12]). (a) A part's boundaries. (b) A part's cuts. (c) A part's cuts.

flying object when the part salience is determined primarily by its relative size [i.e., the part's cuts in Fig. 2(c)]. As a result, the part salience would affect not only the high-level visual processes that determine the interpretation of a shape, but also the low-level visual processes that determine how the shape is really decomposed. In order to precisely determine a part's cuts, another independent theory that incorporates *a priori* knowledge about the shape is usually required. In the early 1980s, 3-D object recognition was a popular research topic [6]. Also, among the large number of research issues, 2-D perceptual organization [28] and recognition-by-components (or parts) [1], [11], [30] were two important directions. However, their development was hindered by some ill-posed early vision problems, such as edge detection and image segmentation. Since these problems could not be solved, 2-D perceptual organization and recognition by 2-D components (or parts) could not be converted into "complete" computational processes, so they both failed. Nowadays, there are large numbers of 3-D models distributed worldwide. Since 3-D models (or meshes) are not restricted by the limitations of 2-D images, perceptual organization is now possible in 3-D cases. Furthermore, Hoffman and Singh's theory of part salience means that *a priori* knowledge may not be necessary in 3-D shape-decomposition processes. However, the quantitative definitions for part salience proposed by Hoffman and Singh [12] were made under the assumption that a part and its boundary are found in advance. In terms of perceptual organization, this is a drawback that, to some extent, limits the power of Hoffman and Singh's theory. In this paper, we propose a new mesh-decomposition scheme that incorporates the cognitive psychology theory into the mesh-decomposition process such that the visually significant components can be extracted from a given 3-D mesh.

III. Visual Salience-Guided Mesh Decomposition

We now discuss the computational processes for realizing two of the visually salient features, namely, the protrusion and the boundary strength. We also describe how to incorporate each visually salient feature into the mesh-decomposition process. As to the third salient feature, the relative size of components, we can easily calculate it once the protrusion and the boundary strength are known. We shall use the relative size feature in the mesh retrieval process. This section is organized as follows. In Section III-A, we present the computational process for characterizing the protrusion of an arbitrary surface mesh. Based on protrusion characterization, a local maximum approach for choosing the salient representatives of parts is proposed and described in Section III.B. In Section III-C, we describe in detail the proposed computational process for modeling the boundary strength. The proposed measure of boundary strength is used as the guideline to find the locale of a part's boundary. In Section III-D, a coarse-to-fine approach is proposed for finding the locale of a part's boundary. In Section III-E, Katz and Tal's algorithm for determining the boundary of a part (presented in [15]) is described for the purpose of completeness.

A. Modeling the Protrusion as the Degree of Center

In this section, we propose a suitable way to characterize the protrusion of a shape. It is intuitive that a protrusion is closely related to the skeletal structure of a shape. As a result, some existing skeletonization methods [5], [21], [22], [32] may be useful for characterizing the protrusion. In our investigations, however, we found that the integral function proposed by Hilaga *et al.* [14] is more suitable for protrusion characterization. The main reasons are as follows. First, the integral function can be constructed on any type of polygonal meshes, including non-orientable, non-closed, and non-manifold surfaces. Second, the function is very stable so that there is no initial point selection problem. Third, the integral can be calculated over the entire surface. As a result, the protrusion of every vertex is accessible to any salience-guided process. Finally, the function is not only invariant to geometrical transformations (such as rotation, translation, and scaling), but is also resistant to noise added to vertex coordinates. Therefore, we adopt the integral function described in [14] to characterize the protrusion of a part.

In [14], the degree of center at the point v on the surface S is defined as follows [14]:

$$\mu(v) = \int_{p \in S} g(v,p)\, dS \tag{1}$$

where $g(v,p)$ represents the geodesic distance between v and p on S. The continuous integral function $\mu(v)$ is defined as the total sum of geodesic distance from the point v to all points on S. In other words, the value of $\mu(v)$ can be interpreted as a distance from the point v to arbitrary points on S. More precisely, a smaller value of $\mu(v)$ indicates that the point v is closer to the center of the surface S. On the other hand, a larger value of $\mu(v)$ means that the point v is farther from the center of the surface. It can be seen from (1) that calculating the integral based on geodesic distance is computationally prohibitive. To trade off accuracy for computational efficiency, Hilaga *et al.* employed Dijkstra's algorithm to approximate geodesic distance based on edge length of a 3-D mesh.

Here, in contrast to [14], the integral function is constructed on the dual graph of a given 3-D mesh, $G = (V, E)$, where V and E represent the set of dual vertices and the set of dual edges, respectively. A dual vertex $v \in V$ is referred to as 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 edge shared by the two faces. For computational efficiency, we segment the mesh into small patches of approximately equal size, which we called base patches. Each base patch is represented by a single dual vertex, b_i, located at its approximate center. Such a base patch is constructed by a modified version of Dijkstra's algorithm such that the shortest distance between the base vertex and any vertex within the base patch is less than a radius value. As shown in Fig. 3, the darker region is the base patch of the radius thr_μ with the base dual-vertex b_i in its center. Obviously, by increasing the number of base patches,

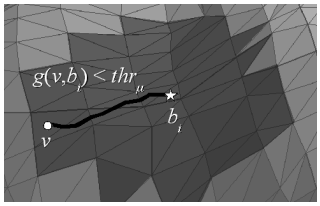


Fig. 3. Illustration of the base patch construction for protrusion characterization: the darker region is the base patch occupied by b_i.

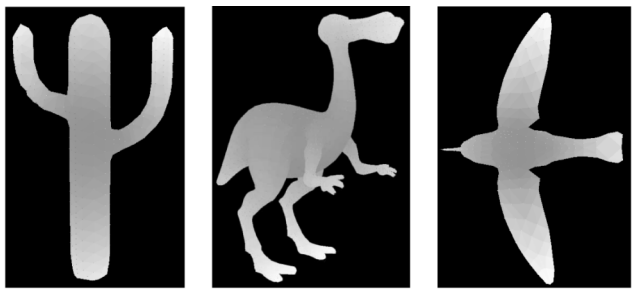


Fig. 4. Protrusion degree calculated on different 3-D meshes. (a) Cactus. (b) Dinopet. (c) Hummingbird.

a more accurate integral can be obtained; however, the drawback is an increase of computation time. Let $\mathrm{area}(v)$ denote the area of the mesh face corresponding to a dual vertex v and $\mathrm{area}(V)$ denote the total area of the object surface. The protrusion degree at a dual vertex v can be defined as in [14]

$$\mu(v) = \sum_i g(v, b_i) \cdot \mathrm{area}(P_i) \tag{2}$$

where $\{b_0, b_1, \ldots\}$ are the base dual-vertices representing the base patches $\{P_0, P_1, \ldots\}$. In addition, $\mathrm{area}(P_i)$ is the area of the entire base patch $\mathrm{area}(P_i) = \sum_{v_j \in P_i} \mathrm{area}(v_j)$ and $\sum_i \mathrm{area}(P_i) = \mathrm{area}(V)$, while $g(v, b_i)$ returns the geodesic distance between the dual vertex v and the base vertex b_i. Since the function $\mu(v)$ defined in (2) is not invariant to scaling transformation, a normalized version of $\mu(v)$ is defined as in [14]

$$\mathrm{Protrusion}(v) = \frac{\mu(v) - \min_{u \in V} \mu(u)}{\max_{u \in V} \mu(u)}. \tag{3}$$

The calculation of the integral function has the complexity $O(|V|\log|V|)$, where $|V|$ is the number of faces on the mesh. Using the normalized protrusion degree defined in (3), we can calculate a numeric value (ranging from 0 to 1) for each dual vertex located on a 3-D mesh. The farther a dual vertex is from the center of a 3-D mesh, the larger the protrusion degree will be. Fig. 4 illustrates the protrusion degree calculated on different 3-D meshes. Note that a darker color represents a protrusion degree close to 0, while a lighter color means the protrusion degree is close to 1.

B. Choosing the Salient Representatives of Parts

Here, we describe how to select a set of salient representatives from a given 3-D mesh. The local maxima of protrusion degrees is the criterion used to select salient features. After the selection process is completed, each identified local maximum can be regarded as a salient representative of a part. Given a dual vertex $r \in V$, the dual vertex is chosen as a salient representative of a part if the following condition is satisfied:

$$\mathrm{Protrusion}(r) = \max_{W_r} \{\mathrm{Protrusion}(v)\} \tag{4}$$

where $W_r = \{v \in V | g(r, v) < \mathrm{thr}_p\}$ is an observation window for finding a local maximum of protrusion degrees; and thr_p represents the size of the observation window, with which we can control the range of influence of a protrusive stimulus. The observation window can be constructed using the modified version of Dijkstra's algorithm mentioned in the previous section. By replacing b_i and thr_μ in Fig. 3 with r and thr_p, respectively, the darker region shown in Fig. 3 can be interpreted as the observation window for choosing the salient representative. Moreover, if the protrusion degree of the vertex r (i.e., the star shown in Fig. 3) is the largest value within the local window, the vertex is chosen as the salient representative. Note that since the observation windows of local maxima are subject to overlap, only one of them is chosen as a salient representative.

C. Modeling the Boundary Strength Based on the Border Area Change

In this section, we describe how to convert the concept of boundary strength into a computational process. Since the boundary of a part is completely unknown, we start from the surface mesh and the salient representatives obtained in the previous section. Motivated by the concept of locale turning (described in Section II), we use Dijkstra's algorithm [4] to explore how the surface evolves in the locale of a boundary. For clarity, we split the computational process for modeling the boundary strength into two steps:

1) Step 1. Establishing the Candidate Locales: Given a salient representative of a part, r, a set of candidate locales, $\{L_r^x\} = \{L_r^0, L_r^1, \ldots\}$, is established. For simplicity and later use, we drop the subscript r in subsequent descriptions and denote the xth candidate locale as

$$L^x = \{v \,|\, \forall v \in V, x \cdot e \le D(v) < (x+1) \cdot e\} \quad \text{for } x \in \{0, \ldots, l-1\} \tag{5}$$

where e represents the extent of a candidate locale, in which the boundary evolution is explored; and $l = \lfloor \max_{v \in V} D(v)/e \rfloor$ is the number of candidate locales established. $D(v)$ returns the shortest distance from the source, r, to a dual vertex, v, in terms of geodesic distance and protrusive difference. Fig. 5 illustrates that based on the new distance measure $D(\cdot)$, the first two candidate locales, L^0 and L^1, are established using the modified version of Dijkstra's algorithm (as in Section III-A). To compute the shortest distance $D(\cdot)$, the weight for each edge $(u, v) \in E$ in the dual graph is defined as follows:

$$\mathrm{Weight}(u, v) = \delta \cdot \frac{\mathrm{Len}(u, v)}{\mathrm{avg(Len)}} + (1 - \delta) \cdot \frac{\mathrm{Prot}(u, v)}{\mathrm{avg(Prot)}} \tag{6}$$

where $\mathrm{Len}(u, v)$ is the length of a dual edge between u and v. Here, $\mathrm{Prot}(u, v)$ represents the absolute protrusion degree of difference between two dual vertices, u and v. Also, $\mathrm{avg(Len)}$

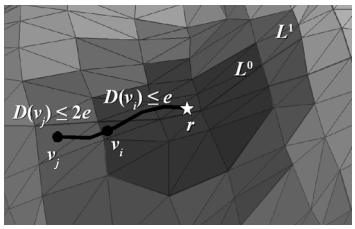


Fig. 5. Illustration of the candidate locales construction: the two darker regions are the first two candidate locales of the salient representative of a part.

and $\mathrm{avg}(\mathrm{Prot})$ represent the average length and the average protrusion degree difference, respectively. In order to fulfill the proximity and similarity requirement of the Gestalt laws, the first term on the right-hand side of (6) is usually considered in Dijkstra's algorithm to determine the shortest distance (or path) on a graph in terms of the geodesic proximity. The purpose of the second term is to balance the effect caused by the geodesic proximity, while creating the candiate locales containing similar protrusion degrees; δ is the weighting between the two constraints. Moreover, including the protrusive similarity is helpful in maintaining a locale's boundaries approximately parallel to a part's boundaries.

Since each salient representative produces a set of candidate locales and eventually grows into the whole 3-D mesh, certain of candidate locales must "march" by the potential region of its corresponding part boundary. However, the sets of candidate locales will overlap one another. To prevent candidate locales from marching into the regions occupied by other parts, a constrained set of candidate locales is constructed such that the region-growing process always ends whenever a termination base is touched. To do so, we first define the termination base, K, as follows:

$$K = \{v | \forall v \in V, \mathrm{Protrusion}(v) \le \mathrm{thr}_b\} \tag{7}$$

where thr_b is the parameter used to collect the set of faces that forms the termination base. Next, the constrained set of candidate locales, $\mathbf{L}$, is defined as the union of $(m+1)$ *consecutive* locales

$$\mathbf{L} = \bigcup_{x=0}^{m<l-1} L^x \tag{8}$$

satisfying the following constraint:

$$L^x \cap K \neq \emptyset, \qquad \text{for } x \in \{m - \Delta_b, \ldots, m\} \tag{9}$$

where Δ_b is used to specify that the last $(\Delta_b + 1)$ locales in $\mathbf{L}$ overlap with the termination base, K. By the above construction, the overlap between a constrained set of locales and the termination base provides a potential region in which to find the correct boundary of a part.

2) Step 2. Modeling the Boundary Strength: With the constrained set of candidate locales established in Step 1, we now consider two adjacent locales in $\mathbf{L}$ to determine how the surface evolves in candidate locales. Let V_{L^x} denote the set of dual vertices in $L^{x+1} \in \mathbf{L}$ that has a dual edge joining $L^x \in \mathbf{L}$ in the graph G. We then associate the following geometric property to the xth candidate locale in $\mathbf{L}$:

$$f(x) = \sum_{v \in V_{L^x}} \mathrm{area}(v). \tag{10}$$

Since V_{L^x} is a set of dual vertices that collects the direct neighbors between L^x and L^{x+1}, $f(x)$ can be regarded as the total-area-of-border between two adjacent candidate locales. Based on the geometric property defined in (10), we model the boundary strength as the total-area-of-border change in response to the boundary's evolution. The modeling is reasonable because, at the border of two adjacent parts, the total-area-of-border defined above will usually undergo a significant change. Therefore, to judge whether a locale contains a boundary using the total-area-of-border change is a justifiable choice. As a consequence, the boundary strength at the xth candidate locale can be defined as follows:

$$\mathrm{Boundary\_Strength}(x) = |f(x+1) - f(x)|. \tag{11}$$

By obtaining the measure of boundary evolution for the boundary strength, we can explore how the surface evolves in the locale of a part's boundary. Moreover, by treating $f(x)$ as a one-dimensional function defined in $\mathbf{L}$, we can make the process for finding the locale of a part's boundary analytic.

D. Finding the Locale of a Part's Boundary

In this section, we describe how to use the previously defined boundary strength to locate the locale of a part's boundary. As mentioned in the previous section, the boundary strength is quantified in response to the boundary's evolutionary process. Hence, the locale of a part's boundary should possess the maximum boundary strength. However, the function $f(x)$ is very jagged (or noisy), since the faces in the immediate neighbor of the xth candidate locale can never have a regular area due to the nature of a mesh-based object. This makes finding the locale of a part's boundary very difficult. To overcome this, based on Haar wavelet representation [23], [25], the function $f(x)$ is transformed into w different scales, $f_1(x), f_2(x), \ldots, f_w(x)$. Then, the candidate locale that possesses the most significant boundary strength is traced from a coarser scale $f_j(x)$ to a finer scale $f_{j-1}(x)$ until a predefined finer scale is reached. In this way, one can conduct a coarse-to-fine search to identify the locale that contains a part's boundary. Let k_{j-1} denote the index of the candidate locale that possesses the maximum boundary strength in $f_{j-1}(x)$. Then, the index k_{j-1} is determined by the following recursion (see (12)), shown at the bottom of the page, where $I_j = \{2k_j, \ldots, 2k_j + 2\}$ is the search range derived from the index k_j (i.e., the k_jth candidate locale in $f_j(x)$). $\mathrm{Boundary\_Strength}_{j-1}(x)$ is equal to $|c_{j-1,x+1} - c_{j-1,x}|$,

$$\begin{cases} k_w = \arg\max_x \mathrm{Boundary\_Strength}_w(x), & \text{initially} \\ k_{j-1} = \arg\max_{x \in I_j} \mathrm{Boundary\_Strength}_{j-1}(x), & \text{if } 1 \le j < w \end{cases} \tag{12}$$

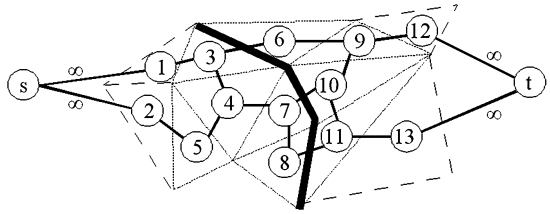


Fig. 6. Illustration of constructing the boundary by solving the maximum-flow (minimum-cut) problem.

which can be regarded as the boundary strength characterized in the finer scale $f_{j-1}(x)$. Note that the above recursion is stopped at the finer scale $f_1(x)$, since the original function $f(x)$ is too noisy to correctly find the locale of the part's boundary. Furthermore, since the extent of a candidate locale may be too small to contain a part's boundary, we can extend it as follows:

$$\widetilde{\mathbf{L}} = \bigcup_{x=2k_1-\Delta_+}^{2k_1+\Delta_-} L^x, \qquad L^x \in \mathbf{L} \tag{13}$$

where Δ_- and Δ_+ are used to, respectively, control the left and right extent of the candidate locale found above. After the region containing a part's boundary is found, the next step is to construct the boundary. In [15], Katz and Tal have shown that by defining an appropriate capacity function, the boundary can be found by solving a maximum-flow (minimum-cut) problem [4], [10]. We, therefore, apply Katz and Tal's method [15] to construct the boundary.

E. Determining the Boundary of a Part

In this section, for the purpose of completeness, we describe Katz and Tal's method [15] for determining a part's boundary. As described in [15], the problem of how to construct a part's boundary within the region containing the boundary is formulated as a maximum-flow (minimum-cut) in an undirected constrained flow network graph problem. To construct the flow network graph, we denote the locale, $\widetilde{\mathbf{L}}$, found in the previous section by the dual graph $G_{\widetilde{\mathbf{L}}} = (V_{\widetilde{\mathbf{L}}}, E_{\widetilde{\mathbf{L}}})$ and the remaining two regions A and B, which are separated by $\widetilde{\mathbf{L}}$, by $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$, respectively (i.e., $G_A \cup G_{\widetilde{\mathbf{L}}} \cup G_B = G$). In addition, the set of all dual-vertices in V_A whose corresponding faces in A share an edge with faces in $\widetilde{\mathbf{L}}$ is denoted by $V_{\widetilde{\mathbf{L}}A}$ (resp. $V_{\widetilde{\mathbf{L}}B}$). Next, we construct an undirected flow network graph $G' = (V', E')$ by adding two new vertices, s (source) and t (sink), as in [15]

$$\begin{aligned} V' &= V_{\widetilde{\mathbf{L}}} \cup V_{\widetilde{\mathbf{L}}A} \cup V_{\widetilde{\mathbf{L}}B} \cup \{s,t\}, \\ E' &= E_{\widetilde{\mathbf{L}}} \cup \{(s,v), \forall v \in V_{\widetilde{\mathbf{L}}A}\} \cup \{(t,v), \forall v \in V_{\widetilde{\mathbf{L}}B}\} \\ &\quad \cup \{e_{uv} \in E | u \in V_{\widetilde{\mathbf{L}}}, v \in \{V_{\widetilde{\mathbf{L}}A} \cup V_{\widetilde{\mathbf{L}}B}\}\}. \end{aligned} \tag{14}$$

As illustrated in Fig. 6, the dotted mesh is the region $\widetilde{\mathbf{L}}$ (i.e., the nodes 3–11) while the solid lines and circle nodes together form the constrained flow network graph. In addition, the two nodes 1 and 2 shown in Fig. 6 belong to the region A while the two nodes 12 and 13 belong to the region B. Note that in (14), the goal of adding the two sets of vertices, $V_{\widetilde{\mathbf{L}}A}$ (i.e., the nodes 1 and 2 in Fig. 6) and $V_{\widetilde{\mathbf{L}}B}$ (i.e., the nodes 12 and 13 in Fig. 6), is to consider the case that the boundary is on either side of the locale, that is either the boundary between $V_{\widetilde{\mathbf{L}}A}$ and $\widetilde{\mathbf{L}}$ or the boundary between $V_{\widetilde{\mathbf{L}}B}$ and $\widetilde{\mathbf{L}}$.

After the flow network graph is constructed, the capacity function Cap on an edge $(u,v) \in E'$ is defined as in [15]

$$\mathrm{Cap}(u,v) = \begin{cases} \frac{1}{1+\mathrm{Ang\_Dist}(\alpha_{uv})\mathrm{avg}(\mathrm{Ang\_Dist})}, & \text{if } \{u,v \neq s,t\} \\ \infty, & \text{else} \end{cases} \tag{15}$$

where α_{uv} represents the angle between the two faces sharing the same edge (u,v) and $\mathrm{Ang\_Dist}$ is a conversion function

TABLE I
TRIANGULATED MESHES USED IN OUR EXPERIMENTS

Model name	Number of vertices	Number of faces	Running time (sec.)
Bunny	3752	7500	5.880
Cactus	620	1236	0.306
Cat	2779	5544	3.355
Cheetah	4704	9404	8.077
Cheetah2	5027	10050	9.036
Dinopet	2039	3999	2.303
Duck	716	1428	0.462
Female	1792	3572	1.573
Hand	1577	3110	1.472
Horse	2502	5000	3.284
Hummingbird	830	1640	0.611
Manatee	1977	3940	2.043
Santa	2502	5000	3.385
Stingray	997	1990	1.142
Tiger	956	1908	0.727

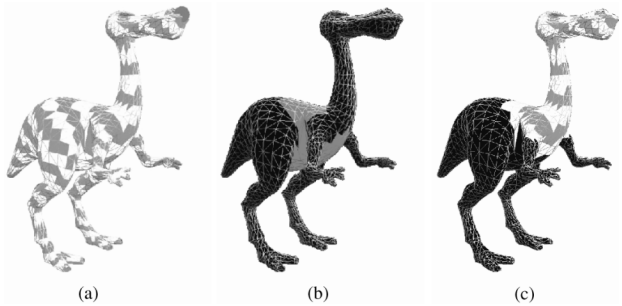


Fig. 7. Illustration of the (a) 32 candidate locales, (b) termination base, and (c) constrained candidate locales construction.

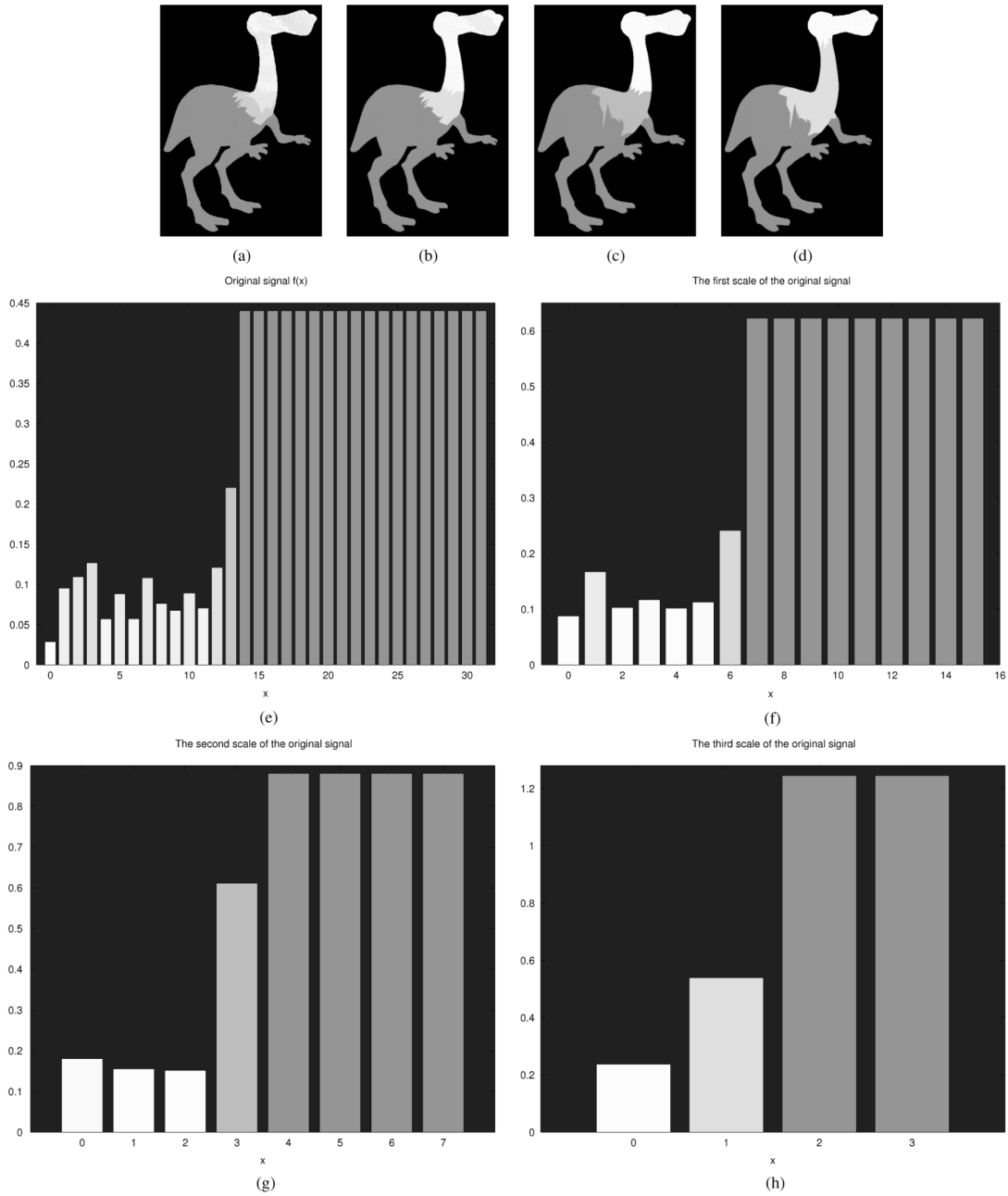


Fig. 8. Each candidate locale is associated with its corresponding geometric property, which can be regarded as a one-dimensional function that defines an object surface. (a)–(d) Different scaled versions of the function $f(x)$ plotted on the object surface. (e)–(h) Different scaled versions of the function $f(x)$ corresponding to (a)–(d), respectively. (a)–(e) $f(x) \in \mathcal{V}_0$ for $0 \leq x \leq 31$, (f) $k_1 = 6$, (g) $k_2 = 2$ and $I_3 = \{4, 5, 6\}$, (h) $k_3 = 1$, and $I_3 = \{2, 3, 4\}$.

$\mathrm{Ang\_Dist}(\alpha_{uv}) = \eta(1 - \cos(\alpha_{uv}))$. The conversion function is used to map the dihedral angle α_{uv} to a positive value; $\mathrm{avg(Agn\_Dist)}$ represents the average angle distance over the entire mesh. In addition, a small positive value for η is used for convex angles while $\eta = 1$ is used for concave angles. In [15], Katz and Tal have shown that by the definition of $\mathrm{Cap}(u, v)$, the minimum cut found in the flow network graph tends to pass through edges having highly concave dihedral angles. As shown in Fig. 6, the thicker line represents the boundary corresponding to the maximum-flow (minimum-cut) in the flow network graph.

With the proposed measures for the protrusion and the boundary strength, the significant components of an arbitrary 3-D mesh can be identified and extracted according to their visual salience (i.e., visual significance). In terms of efficiency, the total complexity of the proposed method is $O(|V|^2 \log |V|)$. The protrusion characterization can be performed in $O(|V| \log |V|)$ [14]. In addition, the process for choosing a part's salient representative costs $O(|V|^2 \log |V|)$. The process for modeling the boundary strength can be performed in $O(R|V| \log |V|)$, where R denotes the number of salient representatives.

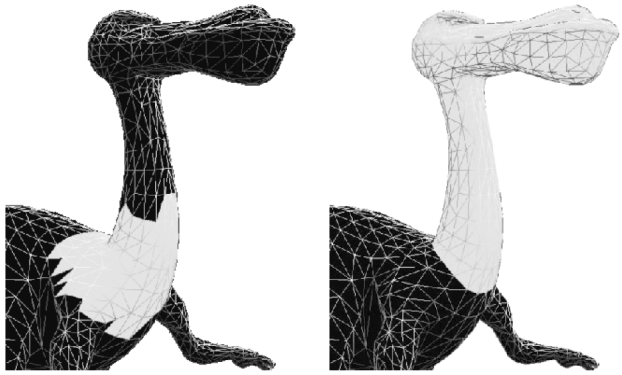
IV. Experimental Results

A series of experiments were conducted to test the effectiveness of the proposed method. We used the set of triangulated meshes listed in Table I as the data set in our experiments.

Moreover, all the parameters used in our experiments were the same and set as follows. To choose the salient representatives of parts, we found that most of the salient representatives could be properly chosen by assigning $\mathrm{thr}_p = 5 \cdot thr_\mu$, where thr_μ was used to generate the base patches for protrusion characterization (as mentioned in Section III-A). In the implementation process, we adopted the radius value $\mathrm{thr}_\mu = \sqrt{0.005 \cdot \mathrm{area}(V)}$ described in [14] to generate the set of base patches. Note that since the parameter, thr_p, is related to the total area of a mesh surface, the number of salient representatives depends on the mesh itself. As shown in Fig. 10, different meshes have different numbers of salient representatives. (i.e., the balls on the surfaces). To generate the constrained set of candidate locales, the threshold value $\mathrm{thr}_b = 0.05$ was adopted to generate the termination base. To perform Dijkstra's algorithm, the parameter $\delta = 0.5$ was used to balance the weighting between the geodesic proximity and the protrusive similarity. To obtain the wavelet transform of $f(x)$, a fixed number of candidate locales $l = 32$ was adopted. As a result, we assigned the extent of a locale as $e = \lfloor \max_{v \in V} D(v)/32 \rfloor$ during the process of collecting candidate locales. In addition, for $x \in \{m+1, \ldots, l-1\}$ the value of $f(x)$ was padded with $f(m)$.

A. Results of Finding the Locale of a Part's Boundary

This experiment was comprised of two parts. The intent of the first part was to show how to construct the candidate locales, the termination base, and the constrained set of candidate locales. As shown in Fig. 7(a), 32 candidate locales were collected by applying Dijkstra's algorithm to the source (i.e., the ball near the dinopet's mouth). Meanwhile, Fig. 7(b) shows the termination base extracted from the dinopet model. Based on the results of Fig. 7(a) and (b), the constrained set of candidate locales was collected such that the consecutive candidate locales (starting with L^0) were gathered together and the last three candidate locales overlapped with the termination base. As shown in Fig. 7(c), these constrained candidate locales were effective in localizing the visual part. The intent of the second part of this experiment was to show the effectiveness of the proposed method in finding the locale of a boundary. As mentioned in Section III-C, the change of total-area-of-border between two adjacent locales is utilized to judge how a part's boundary evolves in the candidate locales. For visualization purposes, the measure of total-area-of-border (i.e., $f(x)$) is converted into a grayscale; thus, the darker the grayscale, the larger the total-area-of-border. Fig. 8(a) shows the original function $f(x)$, shown in Fig. 8(e), plotted on the surface of the dinopet model. More precisely, the grayscale of the xth candidate locale on the surface mesh corresponds to the total-area-of-border $f(x)$. It can be seen from Fig. (e) that the shape of the original function is rough and uneven. Fig. 8(f)–(h) show three different scales of the original function, $f_1(x)$, $f_2(x)$, and $f_3(x)$, respectively. Their corresponding plots on the surface mesh are shown in Fig. 8(b)–(d), respectively. To find the locale of a part's boundary, we started from the coarsest scale $f_3(x)$ and then found the most significant boundary strength within this scale (i.e., $k_3 = 1$). This shows that the locale of the boundary can be found within the 8th-16th candidate locales. Next, within the second scale, $f_2(x)$, the most significant boundary strength was located at $k_2 = 2$. The search range for the locale of the boundary was then shrunk (i.e., within the 8th-12th candidate locales). Finally, we stopped at the finer scale, $f_1(x)$, and $k_1 = 6$ was found within this scale. The locale of the boundary was then formed by the union of the 11th-13th candidate locales. As shown in Fig. 9, the proposed method described in Section III-D is effective in finding the locale of a part's boundary.



(a) (b)

Fig. 9. Locale of the part's boundary and its decomposition result: (a) the region that contains a part's boundary was found by applying the proposed method; (b) the nearly concave boundary was constructed within the region in (a) using the method proposed by Katz and Tal [15]. (a) Locale of the part's boundary (i.e., $L^{11} \cup L^{12} \cup L^{13}$). (b) Visual part and its boundary.

B. Visual Salience-Guided Mesh Decomposition for Extracting Significant Components

This experiment was comprised of three parts. The intent of the first part of this experiment was to show the effectiveness of the proposed method in decomposing a 3-D mesh into parts. In this experiment, each model listed in Table I was decomposed into a set of parts using the proposed decomposition method[4]. The last column of Table I lists the running time of decomposing the 15 test models into parts, on a Pentium IV, 1.98-GHz, 1-GB RAM PC. Each individual part of a model was then associated with its visually salient features so that the significant component could be identified. Fig. 10 shows the 15 test models decomposed into visual parts after the visual salience-guided mesh-decomposition method was applied. Since protrusion characterization was used to choose the salient representatives, it can be seen from Fig. 10 that most of the representatives were located at the tips of parts. On the other hand, as shown in Fig. 10, the boundaries between the parts and the main "body" were constructed precisely according to boundary strength.

Since human visual perception of parts is insensitive to noise and small undulations applied to the vertex coordinates of a 3-D object, the proposed method mimics the same visual processes.

[4]Note that in the proposed mesh-decomposition scheme, each part is individually decomposed from a 3-D mesh. As a result, it is possible that a part can be covered by other parts. To deal with this issue, the overlapped faces can be separated according to their geodesic distance to the salient representatives. On the other hand, the overlapped parts may be merged in the case where the amount of the overlapped area is larger than a predefined threshold. Currently, these two features are implemented by our system. However, a more intelligent process for merging parts should be developed to deal with the situation where the salient representatives are chosen incorrectly.

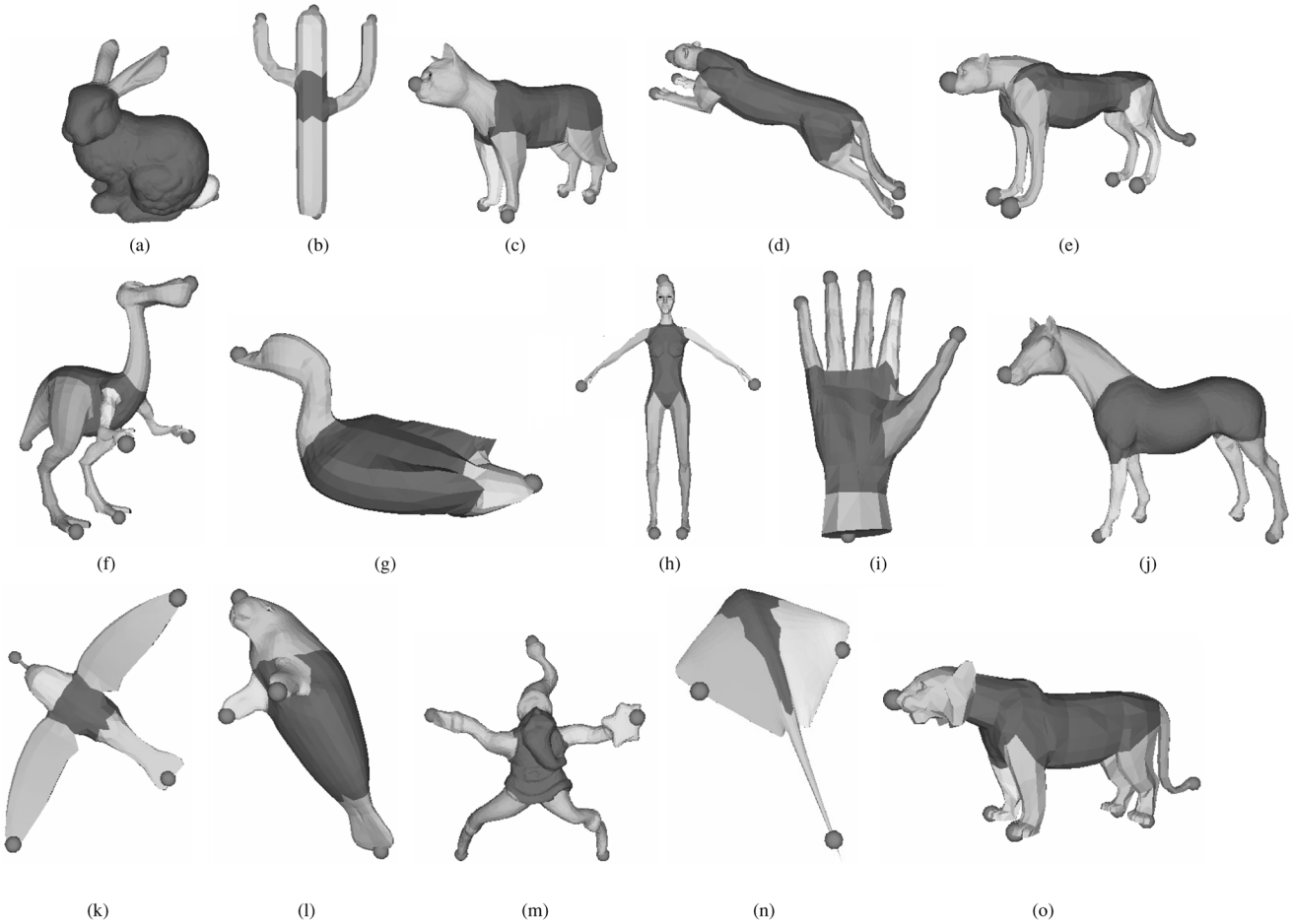
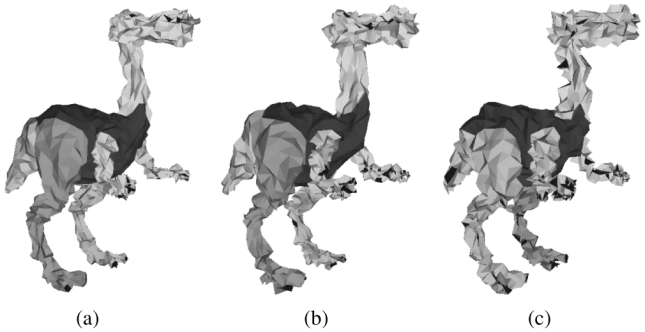


Fig. 10. Visual salience-guided mesh-decomposition results, where the salient representatives of parts chosen from different meshes are indicated by balls on the surfaces: (a) bunny—three representatives; (b) cactus—four representatives; (c) cat—six representatives; (d) cheetah—six representatives; (e) cheetah two to six representatives; (f) dinopet—six representatives; (g) duck—two representatives; (h) female—five representatives; (i) hand—six representatives; (j) horse—five representatives; (k) hummingbird—four representatives; (l) manatee—four representatives; (m) santa—five representatives; (n) stingray—three representatives; (o) tiger—six representatives.

Thus, the intent of the second part of this experiment was to show the robustness of the proposed method under the randomization of vertex coordinates. The randomization was controlled by means of the noise strength, which is defined as the ratio of the largest displacement to the longest edge of the object's bounding box. Fig. 11(a)–(c) show the effects of different levels of noise on the randomization applied to vertex coordinates of the object's surface. It is obvious that the proposed method still succeeds in decomposing the dinopet model into its component parts.

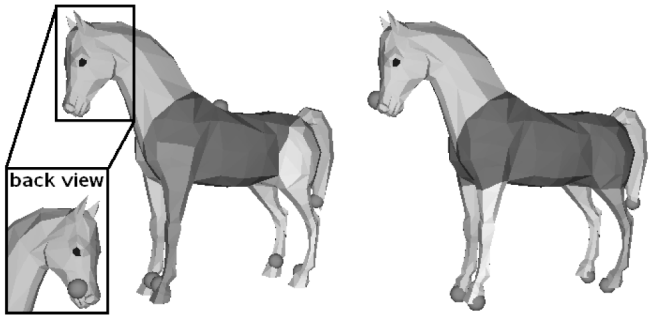
The intent of the third part of this experiment was to compare the proposed scheme against the method of Katz and Tal [15]. As mentioned in [15], there are two versions of Katz and Tal's method: the (recursive) binary decomposition and the fuzzy k-means decomposition. Since the proposed method is type of k-way decomposition, we implemented Katz and Tal's fuzzy k-means decomposition method for the purpose of comparison. Fig. 12 shows two different results of decomposing the donkey model used in [15] into parts using the k-means based method and the proposed method, respectively. As shown in Fig. 12(a), seven representatives were chosen in order to generate the corresponding patches. Fig. 12(b) shows that six salient representatives chosen using the method described in Section III-B were located at the tips of the donkey's four legs, the tip of the donkey's head, and the tip of the donkey's tail, respectively. In the proposed decomposition scheme, contrary to the k-means based method [15], the termination base represents the main body of the model while each feature point represents the part that protrudes from the main body. As a result, the total number of patches generated by the proposed method is equal to the number of salient representatives plus one main body. Moreoever, as shown in Fig. 12(b), the proposed decomposition algorithm favors the boundaries between the protrusive parts and the main body. With regard to efficiency, the running time of decomposing the donkey model using the k-means based approach, which required four iterations to converge, was 2.707 s while that of using the proposed method was 0.452 s. Obviously, the proposed decomposition scheme is more efficient than the k-means scheme. However, the proposed method, like other decomposition techniques, has its limitations. In the following paragraph, we describe the limitations of the proposed method.

One limitation of the proposed method is that the method failed in decomposing the models containing complex topology



(a) (b) (c)

Fig. 11. Robustness of the proposed visual salience-guided mesh-decomposition method under the randomization of vertex coordinates, which is controlled by means of the noise strength ns (i.e., the ratio of the largest displacement to the longest edge of the object's bounding box). (a) $ns = 0.3$. (b) $ns = 0.4$. (c) $ns = 0.5$



(a) (b)

Fig. 12. Results of decomposing the donkey model used in [15] into parts using (a) the fuzzy k-means clustering method (as in [15]) and (b) the proposed method, respectively.

(a) (b)

Fig. 13. Situations in which the proposed method failed to decompose a 3-D mesh into parts: (a) the coffee mug model, which has genus-1 topology, and (b) the Venus head model, which contains more highly concave features than protrusive features.

and more concave features. As shown in Fig. 13(a), the proposed method succeeded in detecting the three feature points representing the handle, the interior, and the exterior of the coffee mug model; however, the proposed method failed to decompose the three parts because of the failure in boundary strength characterization. Fig. 13(b) shows that the proposed method failed to decompose the Venus head model since the head model contains less protrusive features but more concave features. According to Hoffman and Singh's theory [12], to decompose the model containing highly concave features, the boundary strength would be more useful for guiding the decomposition process than the protrusion-based features. As a result, Katz and Tal's method would properly decompose the models containing highly concave features into parts while the proposed scheme would be more efficient and effective in extracting the elongated parts from a given mesh. Another limitation of the proposed method is the ambiguity in parts decomposition. For example, the cactus shown in Fig. 10(b) should be divided into three parts, that is one trunk and two branches; however, as shown in Fig. 10(b), the proposed method cannot solve the problem. Finally, it may be useful to develop an optimization scheme that simultaneously incorporates all the three visually salient factors. Our current implementation is based on the two visually salient features: the protrusion and the boundary strength of a part. However, as mentioned in Section II, human visual perception determines a part's salience by three factors: the protrusion, the boundary strength, and the relative size of a part. Although the relative size feature was not used in the mesh-decomposition process, it can be easily calculated and used in the 3-D mesh retrieval process.

V. Conclusion

We have presented a visual salience-guided mesh-decomposition scheme based on Hoffman and Singh's theory of part salience [12] for extracting significant components from 3-D meshes. More specifically, the protrusion and the boundary strength are modeled as the degree of center on the surface and the total-area-of-border change, respectively. To extract the visually significant components from a given 3-D mesh, these salient features are incorporated into the mesh-decomposition process. The proposed scheme has three remarkable features: 1) the protrusion characterized over the entire surface is used as a guide to choose the salient representatives of the parts; 2) the total-area-of-border change characterized over the entire surface is used as a guide to find the locale of a part's boundary; and 3) the robustness against randomization of vertex coordinates benefits greatly from the incorporation of visual salience into the decomposition process. To the best of our knowledge, this is the first 3-D mesh-decomposition scheme that not only identifies the part's boundaries defined by the minima rule, but also associates the part with its visual salience.

Acknowledgment

The authors would like to thank the associate editor, Dr. Z. Zhang, and the anonymous reviewers for their comments and suggestions which have improved the readability and technical content of this paper. Polygonal meshes used in this paper were provided courtesy of the University of Washington and Cyberware, and were downloaded from the World Wide Web for our experiments. For the use of the donkey surface model, we would like to thank S. Katz and A. Tal, Department of Electrical Engineering, Technion—Israel Institute of Technology.

References

[1] I. Biederman, "Recognition-by-components: A theory of human image understanding," *Psych. Rev.*, vol. 94, pp. 115–147, 1987.

[2] S. Bischoff and L. Kobbelt, "Ellipsoid decomposition of 3-D models," in *Proc. 1st Int. Symp. 3-D Data Processing, Visualization, and Transmission*, Padova, Italy, 2002, pp. 480–488.

[3] S. Bischoff and L. Kobbelt, "Streaming 3-D geometry data over lossy communication channels," in *Proc. IEEE Int. Conf. Multimedia and Expo*, Lausanne, Switzerland, 2002, pp. 361–364.
[4] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. New York: McGraw-Hill, 2001.
[5] J. H. Chuang, C. H. Tsai, and M. C. Ko, "Skeletonization of three-dimensional object using generalized potential field," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1241–1251, 2000.
[6] C. Dorai and A. K. Jain, "COSMOS—A representation scheme for 3-D free-form objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 10, pp. 1115–1130, Oct. 1997.
[7] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin, "Modeling by example," in *Proc. SIGGRAPH*, Los Angeles, CA, 2004.
[8] V. Guillemin and A. Pollack, *Differential Topology*. Englewood Cliffs, NJ: Prentice-Hall, 1974.
[9] M. Garland, A. Willmott, and P. S. Heckbert, "Hierarchical face clustering on polygonal surfaces," in *Proc. ACM Symp. Interactive 3-D Graphics*, 2001, pp. 49–58.
[10] A. Goldberg and R. Tarjan, "A new approach to the maximum flow problem," *J. ACM*, vol. 35, no. 1, pp. 921–940, 1988.
[11] D. D. Hoffman, "Parts of recognition," *Cognition*, vol. 18, pp. 65–96, 1984.
[12] D. D. Hoffman and M. Singh, "Salience of visual parts," *Cognition*, vol. 63, pp. 29–78, 1997.
[13] A. Hubeli and M. Gross, "Multiresolution feature extraction from unstructured meshes," in *Proc. IEEE Visualization*, San Diego, CA, 2001, pp. 287–294.
[14] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii, "Topology matching for fully automatic similarity estimation of 3-D shapes," in *Proc. SIGGRAPH*, Los Angeles, CA, 2001, pp. 203–212.
[15] S. Katz and A. Tal, "Hierarchical mesh decomposition using fuzzy clustering and cuts," in *Proc. SIGGRAPH*, San Diego, CA, 2003, pp. 954–961.
[16] S. Katz, G. Leifman, and A. Tal, "Mesh segmentation using feature point and core extraction," *Vis. Comput.*, vol. 21, no. 8–10, pp. 865–875, 2005.
[17] J. M. Lien and N. M. Amato, "Approximate convex decomposition," in *Proc. ACM Symp.Computational Geometry*, Brooklyn, NY, Jun. 2004, pp. 457–458.
[18] H. Y. Sean Lin, H. Y. M. Liao, and J. C. Lin, "Visual salience-guided mesh decomposition," in *Proc. IEEE Int. Workshop on Multimedia Signal Processing*, Siena, Italy, Sep. 29–Oct. 1 2004.
[19] R. Liu and H. Zhang, "Segmentation of 3-D meshes through spectral clustering," in *Pacific Conf. Computer Graphics and Applications*, Seoul, Korea, Oct. 6–8, 2004, pp. 298–305.
[20] B. Lévy, S. Petitjean, N. Ray, and J. Maillot, "Least square conformal maps for automatic texture atlas generation," in *Proc. SIGGRAPH*, San Antonio, TX, 2002, pp. 362–371.
[21] X. Li, T. W. Woon, T. S. Tan, and Z. Huang, "Decomposing polygon meshes for interactive applications," in *Proc. ACM Symp. Interactive 3-D Graphics*, 2001, pp. 35–42.
[22] F. Lazarus and A. Veroust, "Level set diagrams of polyhedral objects," in *Proc. ACM Symp. Solid Modeling and Applications*, 1999, pp. 130–140.
[23] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989.
[24] A. Mangan and R. Whitaker, "Partitioning 3-D surface meshes using watershed segmentation," *IEEE Trans. Visualiz. Comput. Graph.*, vol. 5, no. 4, pp. 308–321, 2001.
[25] J. R. Parker, *Algorithms for Image Processing and Computer Vision*. New York: Wiley, 1996.
[26] D. L. Page, A. F. Koschan, and M. A. Abidi, "Perception-based 3-D triangle mesh segmentation using fast marching watersheds," *Proc. Computer Vision and Pattern Recognition*, vol. 2, pp. II27–32, 2003.
[27] J. P. Serra, *Image Analysis and Mathematical Morphology*. London, U.K.: Academic, 1982.
[28] S. Sarkar and K. L. Boyer, "Perceptual organization in computer vision: A review and a proposal for a classificatory structure," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 2, pp. 382–400, Feb. 1993.
[29] S. Svensson and G. S. Baja, "Using distance transforms to decompose 3-D discrete objects," *Image Vis. Comput.*, vol. 20, pp. 529–540, 2002.
[30] K. Siddiqi and B. B. Kimia, "Part of visual form: Computational aspects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 3, pp. 239–251, Mar. 1995.
[31] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
[32] Y. Shinagawa, T. L. Kunii, and Y. L. Kergosien, "Surface coding based on morse theory," *IEEE Comput. Graph. Applic.*, vol. 11, no. 5, pp. 66–78, 1991.
[33] S. Shlafman, A. Tal, and S. Katz, "Metamorphosis of polyhedral surfaces using decomposition," in *Proc. EUROGRAPHICS*, 2002, pp. 219–228.
[34] P. Sander, Z. Wood, G. Gortler, J. Snyder, and H. Hoppe, "Multi-chart geometry images," in *Proc. Eurographics Symp. Geometry Processing*, 2003, pp. 146–155.
[35] T. R. Reed and H. Wechsler, "Segmentation of textured images and Gestalt organization using spatial/spatial-frequency representations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 1–12, Jan. 1990.
[36] K. Wu and M. D. Levine, "3-D part segmentation using simulated electrical charge distributions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 11, pp. 1223–1235, Nov. 1997.
[37] Y. Xiao, P. Siebert, and N. Werghi, "A discrete reeb graph approach for the segmentation of human body scans," in *Proc. 3-D Digital Imaging and Modeling*, 2003, pp. 378–385.
[38] C. T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Trans. Comput.*, vol. C-20, no. 1, pp. 68–86, Jan. 1971.
[39] S. C. Zhu, "Embedding Gestalt laws in Markov random fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 11, pp. 1170–1187, Nov. 1999.
[40] E. Zuckerberger, A. Tal, and S. Shlafman, "Polyhedral surface decomposition with applications," *Comput. and Graph.*, vol. 26, no. 5, pp. 733–743, 2002.

Hsueh-Yi Sean Lin was born in Taipei, Taiwan, R.O.C., in 1975. He received the B.S. degree from Chung-Hua University, Hsinchu, Taiwan, in 1997, and the M.S. degree from Yuan-Ze University, Chung-Li, Taiwan, in 1999, both in computer science. He is currently pursuing the Ph.D. degree in the Department of Computer Science, National Chiao-Tung University, Hsinchu.

His research interests include 3-D mesh processing, retrieval, and authentication.

Hong-Yuan Mark Liao (SM'01) received the B.S. degree in physics from National Tsing-Hua University, Hsinchu, Taiwan, R.O.C., in 1981 and the M.S. and Ph.D. degrees in electrical engineering from Northwestern University, Evanston, IL, in 1985 and 1990, respectively.

He was a Research Associate with the Computer Vision and Image Processing Laboratory, Northwestern University, during 1990–1991. In July 1991, he joined the Institute of Information Science, Academia Sinica, Taipei, Taiwan, as an Assistant Research Fellow. He was promoted to Associate Research Fellow and then Research Fellow in 1995 and 1998, respectively. From August 1997 to July 2000, he served as the Deputy Director of the Institute. From February 2001 to January 2004, he served as the Acting Director of the Institute of Applied Science and Engineering Research. Currently, he is the Director of the Academia Sinica Computing Center. He is also jointly appointed as a Professor of the Computer Science Department, National Chiao-Tung University, Hsinchu. His research interests include multimedia signal processing, video-based Surveillance Systems, content-based multimedia retrieval, and multimedia protection.

Dr. Liao is the Managing Editor of the *Journal of Information Science and Engineering* and is on the editorial boards of the *International Journal of Visual Communication and Image Representation*, *EURASIP Journal on Applied Signal Processing*, and the *Tamkang Journal of Science and Engineering*. He was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA during 1998–2001. Currently, he serves as a member of the Multimedia Signal Processing Technical Committee of the IEEE Signal Processing Society. He was

the recipient of the Young Investigators' award from Academia Sinica in 1998 and the Excellent Paper Award from the Image Processing and Pattern Recognition Society of Taiwan in 1998 and 2000. He served as the Program Chair of the International Symposium on Multimedia Information Processing (1997), the Program Co-Chair of the Second IEEE Pacific-Rim conference on Multimedia (2001), the Conference Co-Chair of the 5th International Conference on Multimedia and Exposition (2004), Program Co-Chair of the IEEE International Symposium on Multimedia (2005), and will serve as Program Co-Chair of 2006 IEEE International Symposium on Multimedia. He received the Distinguished Research Award from the National Science Council of Taiwan in 2003 and the National Invention Award of Taiwan in 2004.

Ja-Chen Lin received the B.S. degree in computer science in 1977 and the M.S. degree in applied mathematics in 1979, both from National Chiao-Tung University (NCTU), Hsinchu, Taiwan, R.O.C., and the Ph.D. degree in mathematics from Purdue University, West Lafayette, IN, in 1998.

During 1981–1982, he was an Instructor at NCTU. During 1984–1988, he was a Graduate Instructor at Purdue University. He joined the Department of Computer Science, NCTU, in August 1988, and is currently a Professor there. His research interests include pattern recognition and image processing.

Dr. Lin is a member of the Phi Tau Phi Scholastic Honor Society.