Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

A level set based segmentation approach for point-sampled surfaces*

MIAO Yong-wei†[1,2], FENG Jie-qing[1], ZHENG Guo-xian[1], PENG Qun-sheng[1]
([1]*State Key Lab. of CAD & CG, Zhejiang University, Hangzhou 310027, China*)
([2]*College of Science, Zhejiang University of Technology, Hangzhou 310032, China*)
†E-mail: miaoyw@cad.zju.edu.cn

Received Sept. 15, 2006; revision accepted Dec. 7, 2006

Abstract: Segmenting a complex 3D surface model into some visually meaningful sub-parts is one of the fundamental problems in digital geometry processing. In this paper, a novel segmentation approach of point-sampled surfaces is proposed, which is based on the level set evolution scheme. To segment the model so as to align the patch boundaries with high curvature zones, the driven speed function for the zero level set inside narrow band is defined by the extended curvature field, which approaches zero speed as the propagating front approaches high curvature zone. The effectiveness of the proposed approach is demonstrated by our experimental results. Furthermore, two applications of model segmentation are illustrated, such as piecewise parameterization and local editing for point-sampled geometry.

Key words: Point-sampled surfaces, Segmentation, Level set method, Extended curvature field
doi:10.1631/jzus.2007.A0575 **Document code:** A **CLC number:** TP391.7

INTRODUCTION

With the rapid development of 3D digital photographic and scanning devices, large-scale point-sampled surfaces are now becoming substantially increased and popular in computer graphics. Many researcher efforts were dedicated to the investigation of surface reconstruction, efficient processing and modelling, and rendering for point-sampled geometry (Alexa *et al.*, 2001; 2003; Adamson and Alexa, 2003; 2006a; 2006b; Amenta and Kil, 2004; Fleishman *et al.*, 2005; Kobbelt and Botsch, 2004; Liu *et al.*, 2006; Pauly *et al.*, 2002; 2003; 2006; Zwicker *et al.*, 2002; Zwicker and Gotsman, 2004). For 3D surface model, extracting distinct features is crucial for several applications like shape recognition and matching (Mangan and Whitaker, 1999; Page *et al*., 2003; Liu and Zhang, 2004), texture mapping (Levy *et al.*, 2002), surface parameterization (Yamauchi *et al.*, 2005b), metamorphosis (Shlafman *et al.*, 2002), shape modelling and editing (Funkhouser *et al*., 2004), etc. Model features can be regarded as its distinct sub-parts that can characterize the model, such as tail of a bunny, eyes of a dinosaur, etc. In computer graphics, part-type model segmentation always refers to the partitioning of a given model (a complex mesh or point-sampled geometry) into distinct meaningful sub-parts according to its geometric features (Shamir, 2004). Unfortunately, current approaches to part-type model segmentation typically assume the surface model to be represented by mesh (Mangan and Whitaker, 1999; Katz and Tal, 2003; Page *et al.*, 2003; Shamir, 2004; Liu and Zhang, 2004; Yamauchi *et al.*, 2005a; 2005b; Katz *et al.*, 2005; Attene *et al.*, 2006). It is necessary to develop new technique for partitioning a given point-sampled geometry into different components without explicit construction of triangle mesh.

In this paper, based on the level set evolution scheme, a novel segmentation approach for point-sampled geometry is proposed. In order to segment the given model along high curvature zones, driven speed function for the level set evolution is defined by

* Project supported by the National Basic Research Program (973) of China (No. 2002CB312101), the National Natural Science Foundation of China (Nos. 60503056, 60373036, 60333010), and the Education Department of Zhejiang Province, China (No. 20060797)

the extended curvature field, which approaches zero speed as the propagating front approaches high curvature zone. As a result, our approach can decompose the point-sampled model into meaningful components. The contributions of this paper include:

(1) segmentation of point-sampled surfaces directly without explicit construction of mesh;

(2) estimation of local surface differentials for sample points by projection method;

(3) evolution of propagating front on the narrow band of manifold;

(4) illustration of two direct applications for model segmentation.

The rest of this paper is organized as follows. The related works about model segmentation are briefly reviewed in Section 2, and some preliminary knowledge about the level set method is given in Section 3. In Section 4, an overview of our segmentation approach is described. Section 5 describes the details of evolving the propagating fronts under the extensive curvature field. Section 6 shows some experimental results. Two applications of our approach are illustrated in Section 7. Finally, Section 8 summarizes our method, and discusses some directions for further research.

RELATED WORK

Recently, various approaches for model segmentation have been used successfully in digital geometry processing, such as clustering methods, region-growing methods, watershed segmentation, geometric snake, etc. However, these approaches always assume that the model is provided explicitly via triangle mesh.

One automatic mesh segmentation approach is to partition the polygonal mesh into a set of face clusters by greedy face clustering (Garland *et al.*, 2001). Extending the idea of Garland *et al.*(2001)'s plane fitting for each cluster, Attene *et al*.(2006) segmented the mesh hierarchically into some patches that best fit a pre-defined set of primitives, such as planes, spheres, and cylinders, etc. Using spectral analysis for the affinity matrix, Liu and Zhang (2004) applied spectral clustering to 3D mesh segmentation, which can partition given mesh along concave regions. Inspired by Lloyd's Max quantization method, Shlafman *et al.*(2002) proposed a *K*-means based clustering algorithm to decompose mesh into some meaningful components. The decomposition is based on the dihedral angle and the 'physical' distance between the faces. Katz and Tal (2003) decomposed a mesh hierarchically using fuzzy clustering approach based on geodesic distance and angular distance of the dual graph of the surface mesh. Based on the extraction of prominent feature point and core component of the mesh, Katz *et al.*(2005) proposed a hierarchical segmentation algorithm for generating segmentation that is insensitive to pose and proportions. Yamauchi *et al.*(2005a) presented a mean shift clustering scheme for clustering mesh normal and achieved feature sensitive segmentation result.

Region-growing is another mesh segmentation approach, which is closely related with clustering scheme. The generated charts are growing so as to align chart boundaries with high curvature feature zones (Levy *et al.*, 2002). Aiming at producing charts that can be flattened efficiently for low distortion parameterization, Yamauchi *et al.*(2005b) proposed a segmentation method based on integrated Gaussian curvature. The segmentation approach evenly distributes Gaussian curvature over the charts and generates close to developable surface charts.

Inspired by image processing, watershed-based scheme can divide a model into sub-parts. The earlier watershed segmentation by Mangan and Whitaker (1999) favored partition boundaries along high curvature regions and did not single out concavity. Page *et al.*(2003) proposed a fast marching watersheds algorithm where the height map used for their watershed algorithm impeded climbing up negative principal curvature hills, honoring the minima rule.

As an extension of the active contour model (Kass *et al.*, 1988) for image snake, Lee and Lee (2002) proposed the geometric snake as an interactive tool for feature detection and segmentation on a triangular mesh, which slithers from the user-specified initial position to a nearby feature while minimizing an energy functional.

These segmentation approaches for surface meshes always depend heavily upon the globally consistent connectivity information between sampled vertices (Garland *et al.*, 2001; Lee and Lee, 2002; Katz and Tal, 2003; Page *et al.*, 2003; Liu and Zhang, 2004; Katz *et al.*, 2005; Attene *et al.*, 2006), the re-

lated angle measurements and the area measurements for the triangle faces (Yamauchi *et al.*, 2005a; 2005b). However, point-based representation of 3D geometry can be regarded as a discrete sampling of a continuous surface and is a valuable alternative representation of surface meshes for several special applications. Due to lack of topological information, it is difficult to segment point-sampled geometry efficiently and robustly using the above mesh-based approaches. Although topological information may be maintained by a graph which connects each sample point to its nearest neighbors (Zwicker *et al.*, 2002; Kobbelt and Botsch, 2004; Zwicker and Gotsman, 2004), the highly complex connectivity information for the large numbers of sample points leads to an inefficient or time-consuming segmentation procedure.

As far as the authors know, work on segmentation of point-sampled surfaces directly is rare. One exception is the recent work of (Yamazaki *et al.*, 2006). Without explicit construction of a mesh, they introduced a technique for segmenting a point-sampled surface into distinct features through a three-phase process. Under their topological definition of features, the input surface can be coarsened as super-nodes. In order to partition the given point set, their method bisects the set of super-nodes by graph cut technique. Repeated application of the bisection procedure results in a hierarchical segmentation of the input point set. However, as they mentioned, their method can result in some artifacts due to their feature definition.

THEORETICAL BACKGROUND

Image segmentation is one of the fundamental problems in computer vision and medical image processing, i.e., decomposing a 2D image into sub-parts or sub-structures according to its visual features. Kass *et al.*(1988) proposed an active contour model namely Snake. A snake is represented as a parametric curve in an image and its final position can be obtained through an energy minimization procedure. The snake model can semi-automatically detect image features. Recently, image segmentation based on level set involves solving active contours minimization problem by computing geodesics or minimal distance curves (Xu *et al.*, 2004). In the level set scheme, a curve is embedded as a zero level set of a higher dimensional field. The entire field is evolved till minimizing a metric defined by the curvature and image gradient (Leventon *et al.*, 2000).

The level set method was first proposed by Osher and Sethian (1988). It describes the propagating fronts by a PDE (Malladi *et al.*, 1995; Museth *et al.*, 2002), and it can solve the topological change of the interface robustly. Some numerical methods proposed to accelerate the solution (Sethian, 1999), such as narrow band method, fast marching method, higher order difference schemes. Now, the level set method has been successfully applied to image processing, computer vision and computer graphics (Osher and Fedkiw, 2001), etc.

The level set method constructs a higher dimensional field function, and embeds the propagating front as the zero iso-surface of the field function. The front is evolved by solving a PDE on a regular sampling of the field function (Osher and Sethian, 1988). In general, the propagating front moves with a specified speed function along its normal direction.

Let $\Gamma(0)$ be an initial close 2D propagating front (interface surface) in Euclidean space $\mathbb{R}^3$, and $\Gamma(t)$ be interface surface at time t which results from $\Gamma(0)$ moving along its normal direction with a speed function F. Let Ω be the spatial region which Γ encloses. To solve the interface evolution robustly, the $\Gamma(t)$ is embedded into a signed distance function $\Phi(X,t)$, which is negative/positive when a point is inside/outside the spatial region $\Omega(t)$.

The level set method takes the interface $\Gamma(t)$ as the zero level set of $\Phi(X,t)$, that is

$$\Gamma(t)=\{X\in\mathbb{R}^3\,|\,\Phi(X,t)=0\}. \tag{1}$$

The problem of propagating the interface is equivalent to updating the signed distance function. Let $X(t)$ be the particle trajectory on the interface $\Gamma(t)$, which moves with the velocity $\boldsymbol{F}=\mathrm{d}X(t)/\mathrm{d}t$. From the definition of zero level set, we have $\Phi(X(t),t)=0$. By differentiating it with respect to t, then we get

$$\Phi_t+\boldsymbol{F}\nabla\Phi=0. \tag{2}$$

By projecting the velocity vector $\boldsymbol{F}$ on the normal direction of the interface $\boldsymbol{n}=\nabla\Phi/\|\nabla\Phi\|$, we can get the normal velocity to the interface $F_n=\boldsymbol{F}\cdot\boldsymbol{n}$. The evolution equation for the interface may be represented as the following initial value problem:

$$\text{Hamilton-Jacobi: } \begin{cases} \Phi_t + F_n \|\nabla \Phi\| = 0, \\ \Phi(X,0) = \text{given value}. \end{cases} \quad (3)$$

It is also named as Hamilton-Jacobi equation.

OVERVIEW OF OUR SEGMENTATION ALGORITHM

Our proposed segmentation approach based on the level set method is described as follows.

(1) At the preprocessing step, the local surface differential property at each sample point of the model is estimated by the projection method for adaptive neighboring points (Section 5.1).

(2) A 3D uniform voxel grid is constructed in the object space.

(3) The initial seed points on the 3D model are interactively specified. Then the corresponding initial propagating fronts (interface surfaces) and initial partitions are generated.

(4) In order to compute the level set evolution efficiently, a narrow band along the interface is defined and the signed distance function is evaluated only within the narrow band (Section 5.2).

(5) The level set $\Phi(X)$ is updated within the narrow band according to the level set equation by using the up-wind scheme, where the evolution is driven by the speed function $\boldsymbol{F}$ (Section 5.3).

(6) Extracting the interface surface at the zero level set of the signed distance function within the narrow band.

(7) Update the partition of point-sampled geometry, which is $B(t)=\{X|\Phi(X,t)\leq 0\}\cap S$, and go back to Step 4 for constructing a new narrow band according to the new interface and evolving the level set repeatedly (Section 5.4), until the interface evolution stops, i.e., the final segmentation is obtained.

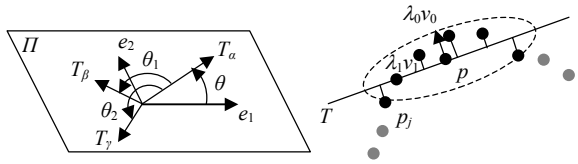
LEVEL SET BASED POINT-SAMPLED SEGMENTATION

Estimating differential property by the projection method

The local differential properties of the point-sampled geometry, such as surface normal $\boldsymbol{n}$, principal directions e_1, e_2, principal curvatures k_1, k_2, etc. are estimated according to local neighborhood of the sample point (Pauly *et al.*, 2002). For the efficiency consideration, the neighbor size for the sample point p is determined adaptively, so that the local sampling density $\rho=k/r^2$ is a constant, where r is the radius of the enclosing sphere of the k-nearest neighbors of the sample point p. The local surface differentials at the sample point can be estimated by the projection method.

The projection method is based on the classical differential geometry (Do Carmo, 1976). The algorithm of estimating the surface differentials at regular surface point can be obtained according to their definitions and the following fact (Jia *et al.*, 2006).

In the tangent plane Π_p at sample point p, it is supposed that the angle from a chosen tangent direction T_α to the principal direction e_1 is θ and the angle from T_α to T_β and T_γ are θ_1 and θ_2, respectively (Fig.1a).



(a) (b)

Fig.1 (a) Analysis of principal directions and principal curvatures in the tangent plane; (b) Covariance analysis

The normal curvatures for directions T_α, T_β and T_γ can be expressed in terms of the two principal curvatures, respectively:

$$\left.\begin{aligned} k_\alpha &= k_1 \cos^2\theta + k_2 \sin^2\theta, \\ k_\beta &= k_1 \cos^2(\theta+\theta_1) + k_2 \sin^2(\theta+\theta_1), \\ k_\gamma &= k_1 \cos^2(\theta+\theta_2) + k_2 \sin^2(\theta+\theta_2). \end{aligned}\right\} \quad (4)$$

Thus, the relation between angles of tangent direction and the corresponding normal curvatures can be deduced:

$$\tan(2\theta+\theta_2) = \frac{\sin(\theta_1-\theta_2)}{\dfrac{k_\alpha-k_\beta}{k_\alpha-k_\gamma}\cdot\dfrac{\sin\theta_2}{\sin\theta_1}-\cos(\theta_1-\theta_2)}. \quad (5)$$

So, according to the three normal curvatures k_α, k_β, k_γ for three sampled directions and corresponding

θ_1, θ_2, the angle θ and the principal directions can be determined from Eq.(5). Furthermore, substituting θ and θ_1, θ_2 into the linear system (1), the principal curvatures k_1 and k_2 can be calculated easily.

Our projection method for estimating surface differentials can be performed as follows:

Step 1: For each regular point p, neighboring points are determined adaptively, which makes the local sampling density a constant. Surface normal $\boldsymbol{n}$ at sample point p may be provided, or we can estimate it using the principal component analysis (PCA) approach to the adaptive neighborhoods (Pauly *et al.*, 2002), and the tangent plane Π_p can be easily obtained.

Step 2: In the tangent plane Π_p, three different tangent directions T_α, T_β, T_γ are sampled. For each sampled tangent direction, a normal plane is defined by the surface normal and the tangent direction.

Step 3: The adaptively selected neighboring points of p are then projected onto the normal plane. The approximated normal curvature for the given tangent direction of p can be computed as the discrete curvature of the projected curve in the normal plane, e.g., the revised principal component analysis (revised-PCA) approach is adopted (Fig.1b), and estimates the normal curvature approximately as variation $\lambda_0/(\lambda_0+\lambda_1)$, where $\lambda_0 \le \lambda_1$ are two eigenvalues of the 2×2 covariance matrix for 2D projection points.

Step 4: According to the three estimated normal curvatures k_α, k_β, k_γ for three tangent directions T_α, T_β, T_γ respectively, the principal directions and corresponding principal curvatures can be determined as the above scheme. Thus the Gaussian curvature and the mean curvature can be computed directly.

The example of local differentials estimation for Stanford bunny model is illustrated in Fig.2. The estimated Gaussian curvature and mean curvature are the second order information of the point-sampled geometry which can be used to define the speed function in the subsequent level set evolution.

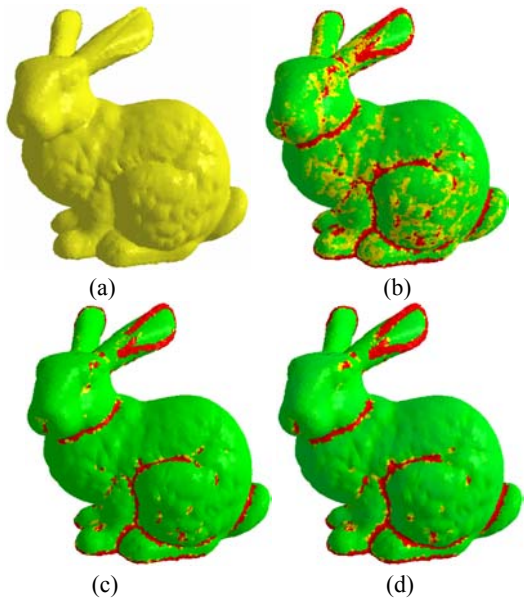


Fig.2 Local surface differential estimations of Stanford bunny model: (a) original bunny model; (b) mean curvature map estimated by the projection method; (c) the Gaussian curvature map estimated by the projection method; (d) the optimized Gaussian curvature map

Obtaining the narrow band and evaluating the distance field

Generally, a level set model is evolved in a distance field, which is a scale function Φ: $\mathbb{R}^3 \to \mathbb{R}$. The propagating front (interface surface) is taken as the zero iso-surface of the distance function. Since we are only interested in the evolution of zero level set, an efficient strategy is to perform the computation in the neighborhood of the zero level set, which is also called narrow band.

To compute the distance field, the initial narrow band is embedded into a uniform 3D Cartesian grid. For each grid point in the narrow band, the signed distance value to initial interface can be calculated as the Euclidean distance by fast marching method. In order to adopt fast marching approach to narrow band, Memoli and Sapiro (2001; 2002) pointed out that the width of the narrow band ϖ should be larger than $d \cdot Step$, where d is the dimensional number of the Euclidean grid, *Step* is the maximal grid step among three directions, i.e., $\max(\|\Delta x\|, \|\Delta y\|, \|\Delta z\|)$. In fact, they proved theoretically that the intrinsic distance of the grid point in narrow band to the zero level set on the manifold can be approximated by the Euclidean distance as the following inequality (Memoli and Sapiro, 2001; 2002):

$$\left\| d_{\text{intrinsic}}(p,q) - d_{\text{Euclidean}}(p,q) \right\| \le C\varpi, \qquad (6)$$

where coefficient C is a constant, $d_{\text{intrinsic}}(p,q)$ and $d_{\text{Euclidean}}(p,q)$ mean the intrinsic distance on the manifold and the Euclidean distance between the point p and q, respectively. The width of the narrow band ϖ is related to the computation accuracy.

In our implementations, the narrow band width is ±4 layers, i.e., L_{-4}, L_{-3}, L_{-2}, L_{-1} and L_{+1}, L_{+2}, L_{+3}, L_{+4}, where the subscripts denote the city block distance from the nearest active grid point, and the negative sign represents the layers inside the zero level set. The layer of the active grid points, i.e. zero level set, is L_0. In order to evolve the propagating front efficiently, only signed distance field in the narrow band is computed.

Evolving level set according to the extended curvature field

During segmenting the point-sampled geometry, the interface surface evolves along its normal direction driven by a specified speed function $\boldsymbol{F}$. In order to segment the model along the high curvature zones, the driven speed function inside narrow band should be defined according to the extended curvature field.

The extended curvature field can be defined as follows: First, the Gaussian curvature $K(x,y,z)$ for each sample point in the narrow band is estimated. Then, for the grid point labelled (i,j,k) inside narrow band, its extended Gaussian curvature $K^*(i,j,k)$ is approximated as $K(x,y,z)$, where the sample point (x,y,z) is the closest to the grid point (i,j,k).

The speed function $\boldsymbol{F}$ is decomposed into two parts F_A and F_I. Then the level set equation can be reformulated as

$$\Phi_t + F_A \|\nabla\Phi\| + F_I \|\nabla\Phi\| = 0. \tag{7}$$

The term F_A is called the advection term which is independent of the curvature field. In general, the constant term F_A is several multiples of voxel gird size. The term F_I is dependent on the surface curvature field, and is defined as

$$F_I = \frac{-F_A}{M-m}\{\|K^*(i,j,k)\| - m\}, \tag{8}$$

where M and m are the maximal and minimal magnitudes of the extended Gaussian curvature at the grid points in the narrow band, respectively. From the definition, the value of F_I ranges from 0 to $-F_A$ as the magnitude of the curvature varies from m to M. Thus, when the magnitude of $K^*(i,j,k)$ approaches its maximum M, the front evolution speed gradually tends to zero. As a result, the propagating front gets closer to the high curvature zones, and eventually comes to a stop. So, the point-sampled geometry will be segmented along the sharp feature zones.

Solving the level set equation

During the model segmentation, the interface surface evolution is governed by the level set equation—Hamilton-Jacobi equation. Several numerical schemes for this equation have been presented (Osher and Sethian, 1988; Sethian, 1999), such as up-wind finite difference scheme, etc. In this scheme, the spatial and temporal derivatives are approximated as finite differences on a discrete gird, and a provably monotone discretization is adopted for the Hamilton-Jacobi equation. In a discrete 3D object space, the Φ_{ijk}^n is denoted as the approximation of the distance field solution at the nth time step for grid node (i,j,k) in the narrow band. The numerical solution of the Hamilton-Jacobi equation can be expressed by using the following up-wind finite difference scheme:

$$\Phi_{ijk}^{n+1} = \Phi_{ijk}^n - \Delta t[\max(F_{ijk},0)\nabla^+ + \min(F_{ijk},0)\nabla^-], \tag{9}$$

where F_{ijk} means evolution velocity F_A+F_I at grid node (i,j,k) and ∇^+, ∇^- mean:

$$\begin{aligned}\nabla^+ = [&\max(D_{ijk}^{-x}\Phi,0)^2 + \min(D_{ijk}^{+x}\Phi,0)^2 \\ &+ \max(D_{ijk}^{-y}\Phi,0)^2 + \min(D_{ijk}^{+y}\Phi,0)^2 \\ &+ \max(D_{ijk}^{-z}\Phi,0)^2 + \min(D_{ijk}^{+z}\Phi,0)^2]^{1/2},\end{aligned} \tag{10}$$

$$\begin{aligned}\nabla^- = [&\min(D_{ijk}^{-x}\Phi,0)^2 + \max(D_{ijk}^{+x}\Phi,0)^2 \\ &+ \min(D_{ijk}^{-y}\Phi,0)^2 + \max(D_{ijk}^{+y}\Phi,0)^2 \\ &+ \min(D_{ijk}^{-z}\Phi,0)^2 + \max(D_{ijk}^{+z}\Phi,0)^2]^{1/2},\end{aligned} \tag{11}$$

where the difference operators D_{ijk}^{+x} and D_{ijk}^{-x} are defined as

$$D_{ijk}^{+x}\Phi = (\Phi_{i+1,j,k} - \Phi_{i,j,k})/\Delta x, \tag{12}$$

$$D_{ijk}^{-x}\Phi = (\Phi_{i,j,k} - \Phi_{i-1,j,k})/\Delta x, \tag{13}$$

Once the propagating front reaches the boundary of the current narrow band, a new narrow band should be reconstructed by adopting the old distance values at the grid points inside the old narrow band as initial values. The distance values at the grid points inside the new narrow band can be obtained by the up-wind

finite difference scheme and the fast marching algorithm.

For each seed point on the point-sampled geometry, the corresponding interface surface is constructed. Then the interface is evolved separately. According to the definition of the speed function $\boldsymbol{F}$, the propagating fronts will keep evolving at the low curvature zones, and will eventually terminate at the high curvature zones. The whole model will be segmented completely after all propagating fronts come to a stop.

Discussions

(1) Selecting seed points interactively. In order to decompose the given point-sampled geometry into some meaningful components, seed points should be chosen by the user. For example, one seed point for each meaningful sub-part is chosen. Then, for each seed point, an initial propagating front for subsequent level set evolution can be obtained as an iso-surface with small distance from the seed point.

(2) Optimizing curvature field. For evolving the propagating front with the extended curvature field effectively, a preprocessing step for building curvature field should be performed (Fig.2d). The following Gaussian-weighted average of the curvature can smooth the curvature field, i.e., the smooth curvature for the sample point p can be computed as

$$G(p)=\frac{\sum_{q\in N_p} K(p)\exp[-\|q-p\|^2/(2\sigma^2)]}{\sum_{q\in N_p}\exp[-\|q-p\|^2/(2\sigma^2)]}. \quad (14)$$

(3) Optimizing partition boundaries. With the level set evolution eventually terminating at high curvature zones (such as the red region of the optimized Gaussian curvature map in Fig.2d), the patches for two sides of the region were generated, however, the optimized split boundary should be found further. Similar to Katz and Tal (2003) and Zhou *et al.*(2004), this problem can be formulated as a graph cutting problem. After defining the high curvature zones as a medial region, an undirected flow network graph can be built from the medial region using the neighboring relationship similar to the method in (Katz and Tal, 2003). To ensure that the model is segmented along the sharp feature zones, the capacity between adjacent sample points p and q can be defined as

$$C(p,q)=\left[1+\frac{d_{\text{norm}}(p,q)}{Avg(d_{\text{norm}})}\right]^{-1}, \quad (15)$$

where angular distance $d_{\text{norm}}(p,q)=1-\cos\alpha$, and α means the angle between normal directions of sample points p and q, and $Avg(d_{\text{norm}})$ means the average of angular distance between neighboring sample points. Then, the optimized partition boundary can be constructed by applying graph cut technique to the network graph, which makes the cut through sharp feature zones without being too jaggy.

EXPERIMENTAL RESULTS

Fig.3 shows an example of segmenting one chart on the bunny model. After one seed point is chosen interactively at back-leg of the bunny, an initial signed distance field and initial propagating front for the seed point are constructed. Then the distance field is evolved by the level set equation. For illustration purpose, some intermediate interfaces are extracted, and the corresponding segmented partitions of the back-leg are shown in Figs.3b~3e. While the propagating front approaches the high curvature zones, the evolving speed will approach to zero.

Finally, the propagating front comes to a stop, and one sub-patch for the back-leg is obtained, which is shown in Fig.3f.

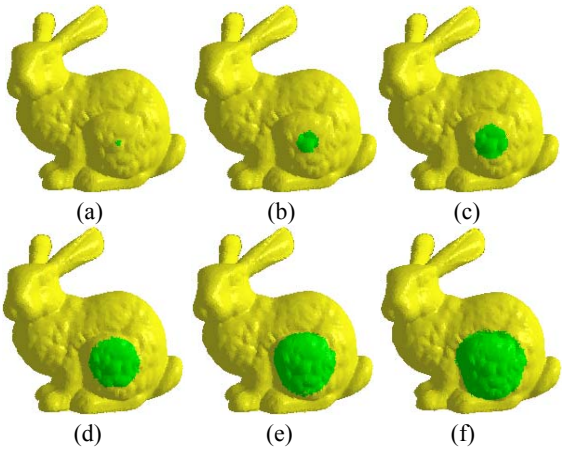


Fig.3 Evolving one chart of the bunny model: (a) bunny model; (b)~(e) evolving one of chart of bunny model; (f) final partition for back-leg

The example of the whole bunny model segmentation is shown in Fig.4. In order to decompose bunny

model into some meaningful components, e.g. bunny face, bunny left ear, bunny right ear, bunny tail, etc. at each meaningful sub-part, one seed point is chosen interactively. For the whole bunny model, a total 12 seed points are selected, then accordingly 12 meaningful sub-patches are generated. Fig.5 is another example of segmenting a dinosaur head into several charts. For dinosaur head model, a total 8 seed points are chosen, and the model is segmented into 8 meaningful components.

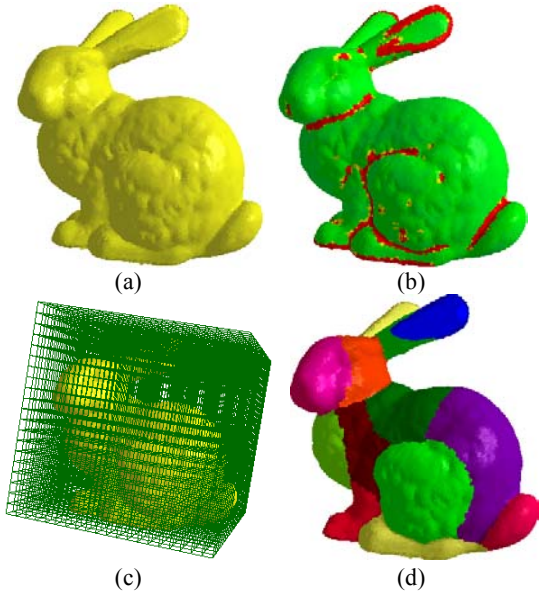


Fig.4 Segmenting the whole bunny model: (a) Stanford bunny model; (b) curvature estimation for bunny model; (c) a 3D uniform voxel grid for bunny model; (d) segmentation result for bunny model

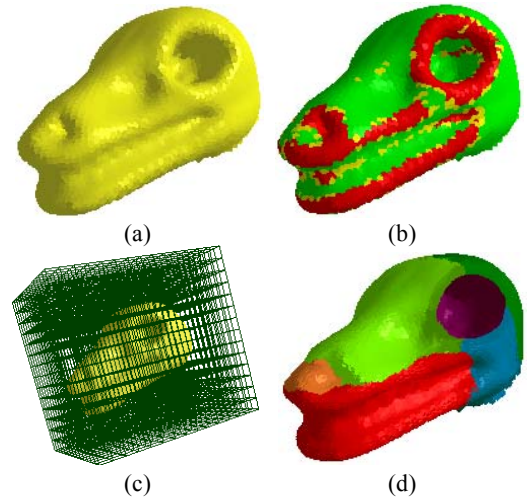


Fig.5 Segmenting the dinosaur-head model: (a) dinosaur-head model; (b) curvature estimation for dinosaur-head model; (c) a 3D uniform voxel grid for dinosaur-head model; (d) segmentation result for dinosaur-head model

APPLICATIONS

Segmenting a surface model into several sub-patches or charts is a fundamental problem in digital geometry processing. In this section, two examples are given as illustration of the potential applications of model segmentation. They are piecewise parameterization and local geometry editing.

Piecewise parameterization for point-sampled geometry

The problem of computing low-distortion parameterization for a point-sampled surface is essential to many applications (Floater and Hormann, 2004), such as texture mapping and texture synthesis, surface remeshing and multiresolution analysis, surface morphing and editing, etc. Since most of the complex point-sampled surfaces are non-developable, the piecewise approach is regarded as an efficient strategy to achieve low-distortion parameterization.

The piecewise parameterization approach first partitions the surface model into a set of patches, each of which is homeomorphic to a disc. They are also called charts. Each chart is flattened onto the plane, i.e. a parameterization. Finally, all parameterized charts are packed into a rectangular texture parametric domain, and the operation of texture mapping is applied at texture parametric domain (Levy *et al.*, 2002; Zhou *et al.*, 2004).

The example of piecewise texture mapping is shown in Fig.6.

Fig.6 Piecewise texture mapping of bunny model: (a) original bunny model; (b) segmenting the bunny model; (c) final texture mapping result of bunny model

Local editing for point-sampled surfaces

Local editing and deformation are necessary for many shape modelling applications (Zwicker *et al.*, 2002; Pauly *et al.*, 2003). The operations usually include local twisting, stretching, bending and constrained texture mapping, etc.

Namely, the user should always define a region of interest (ROI) and a subset of the region, called handles. After the user drags the handles to a new location in 3D space, within the ROI, the shape should follow this manipulation in a natural way or satisfy some specific constraints.

Recently, many researchers are interested in manipulating and modifying 3D shape while preserving geometric details. Geometric detail is an intrinsic property of a surface and can be expressed as an intrinsic surface representation, such as multiresolution representation (Zorin *et al.*, 1997; Kobbelt *et al.*, 1998; Pauly *et al.*, 2006), Laplacian coordinates (Sorkine *et al.*, 2004), Poisson-based gradient field (Yu *et al.*, 2004), discrete differential forms (Lipman *et al.*, 2005), etc. For local surface editing, the 3D shape within the ROI should be reconstructed with respect to new configuration of the handles under preservation of intrinsic surface property as much as possible. This aim can be reconstructed in a least-squares sense, and usually can be translated into solving a linear system to minimize shape distortion.

The example of detail-preserving local editing is shown in Fig.7. The local editing results of right hand of santa model are shown in Figs.7a and 7b. The local editing results for nose of human face are shown in Figs.7c and 7d.

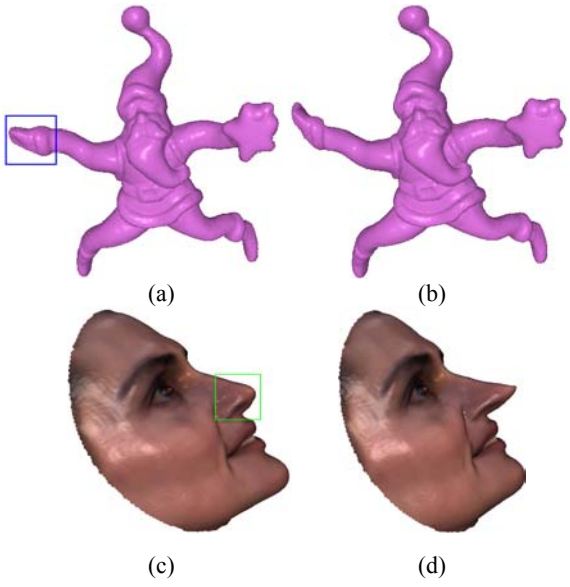


Fig.7 Local editing for 3D model: (a) original santa model; (b) local editing for right hand of the santa model; (c) original face model; (d) local editing for nose of the face model

CONCLUSION AND REMARKS

A novel segmentation approach for the point-sampled surfaces is proposed, which is based on the level set method. The approach simulates the propagating fronts starting from some seed points. The propagating fronts evolved with the specified speed function. In order to align such patch boundaries with high curvature zones, the speed function is defined according to the extended curvature field, which can drive the front to approach high curvature zones. Finally the front comes to a stop when the speed is zero. Compared with other segmentation approaches, our approach can decompose the given point-sampled geometry into some meaningful components. Two applications of our approach, i.e. piecewise parameterization and local editing for point-sampled geometry are given.

Based on the proposed segmentation approach, future researches should focus on geometry processing of point-sampled models, such as local texture mapping, local editing, morphing, compression, etc.

References

Adamson, A., Alexa, M., 2003. Approximating and Intersecting Surfaces from Points. Proc. Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, p.230-239.

Adamson, A., Alexa, M., 2006a. Anisotropic Point Set Surfaces. Proc. Afrigraph 2006, p.7-13. [doi:10.1145/1108590.1108592]

Adamson, A., Alexa, M., 2006b. Point-sampled cell complexes. *ACM Trans. on Graphics*, **25**(3):671-680. [doi:10.1145/1141911.1141940]

Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T., 2001. Point Set Surfaces. Proc. IEEE Visualization 2001, p.21-28. [doi:10.1109/VISUAL.2001.964489]

Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T., 2003. Computing and rendering point set surfaces. *IEEE Trans. on Visualization and Computer Graphics*, **9**(1):3-15. [doi:10.1109/TVCG.2003.1175093]

Amenta, N., Kil, Y.J., 2004. Defining point-set surface. *ACM Trans. on Graphics*, **23**(3):264-270. [doi:10.1145/1015706.1015713]

Attene, M., Falcidieno, B., Spagnuolo, M., 2006. Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer*, **22**(3):181-193. [doi:10.1007/s00371-006-0375-x]

Do Carmo, M., 1976. Differential Geometry of Curves and Surfaces. Prentice Hall, Englewood Cliffs, NJ, p.51-314.

Fleishman, S., Cohen-Or, D., Silva, C.T., 2005. Robust moving least-squares fitting with sharp features. *ACM Trans. on*

Graphics, **24**(3):544-552. [doi:10.1145/1073204.1073227]

Floater, M., Hormann, K., 2004. Surface Parameterization: A Tutorial and Survey. Advances in Multiresolution Analysis of Geometric Modeling'04, p.259-284.

Funkhouser, T., Kazhdan, M., Shilane, P., Min, P., Kiefer, W., Tal, A., Rusinkiewicz, S., Dobkin, D., 2004. Modeling by example. *ACM Trans. on Graphics*, **23**(3):652-663. [doi:10.1145/1015706.1015775]

Garland, M., Willmott, A., Heckbert, P., 2001. Hierarchical Face Clustering on Polygonal Surfaces. Proc. ACM Symposium on Interactive 3D Graphics'01, p.49-58. [doi:10.1145/364338.364345]

Jia, Y.B., Mi, L., Tian, J., 2006. Surface Patch Reconstruction via Curve Sampling. Proc. IEEE International Conference on Robotics and Automation'06, p.1371-1377.

Kass, M., Witkin, A., Terzopoulos, D., 1988. Snakes: active contour models. *Int. J. Computer Vision*, **1**(4):321-331. [doi:10.1007/BF00133570]

Katz, S., Tal, A., 2003. Hierachical mesh decomposition using fuzzy clustering and cuts. *ACM Trans. on Graphics*, **22**(3):954-961. [doi:10.1145/882262.882369]

Katz, S., Leifman, G., Tal, A. 2005. Mesh segmentation using feature point and core extraction. *The Visual Computer*, **21**(8-10):649-658. [doi:10.1007/s00371-005-0344-9]

Kobbelt, L., Campagna, S., Vorsatz, J., Seidel, H.P., 1998. Interactive Multiresolution Modeling on Arbitrary Meshes. ACM SIGGRAPH'98, p.105-114. [doi:10.1145/280814.280831]

Kobbelt, L., Botsch, M., 2004. A survey of point-based techniques in computer graphics. *Computers & Graphics*, **28**(6):801-814. [doi:10.1016/j.cag.2004.08.009]

Lee, Y., Lee, S., 2002. Geometric snakes for triangular meshes. *Computer Graphics Forum*, **21**(3):229-238. [doi:10.1111/1467-8659.t01-1-00582]

Leventon, M., Faugeraus, O., Grimson, W., 2000. Level Set Based Segmentation with Intensity and Curvature Priors. Proc. Workshop on Mathematical Methods in Biomedical Image Analysis, p.4-11. [doi:10.1109/MMBIA.2000.852354]

Levy, B., Petitjean, S., Ray, N., Maillot, J., 2002. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. on Graphics*, **21**(3):362-371. [doi:10.1145/566654.566590]

Lipman, Y., Sorkine, O., Levin, D., Cohen-Or, D., 2005. Linear rotation-invariant coordinates for meshes. *ACM Trans. on Graphics*, **24**(3):479-487. [doi:10.1145/1073204.1073217]

Liu, R., Zhang, H., 2004. Segmentation of 3D Meshes through Spectral Clustering. Proc. Pacific Graphics'04, p.298-305.

Liu, Y.J., Tang, K., Joneja, A., 2006. A general framework for progressive point-sampled geometry. *J. Zhejiang Univ. Sci. A*, **7**(7):1201-1209. [doi:10.1631/jzus.2006.A1201]

Malladi, R., Sethian, J., Vemuri, B., 1995. Shape modeling with front propagation: a level set approach. *IEEE Trans. Pattern Anal. Machine Intell.*, **17**(2):158-175. [doi:10.1109/34.368173]

Mangan, A., Whitaker, R., 1999. Partitioning 3D surface meshes using watershed segmentation. *IEEE Trans. on Visualization and Computer Graphics*, **5**(4):308-321. [doi:10.1109/2945.817348]

Memoli, F., Sapiro, G., 2001. Fast computation of weighted distance functions and geodesics on implicit hypersurfaces. *J. Comput. Phys.*, **173**(1):764-795.

Memoli, F., Sapiro, G., 2002. Distance Functions and Geodesics on Point Clouds. Technical Report, 2002. Http://citeseer.ist.psu.edu/memoli02distance.html

Museth, K., Breen, D., Whitaker, R., Barr, A., 2002. Level Set Surface Editing Operators. ACM SIGGRAPH'02, p.330-338. [doi:10.1145/566570.566585]

Osher, S., Sethian, J., 1988. Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi Formulations. *J. Comp. Phys.*, **79**(1):12-49. [doi:10.1016/0021-9991(88)90002-2]

Osher, S., Fedkiw, R., 2001. Level set methods: an overview and some recent results. *J. Comp. Phys.*, **169**(2):463-502. [doi:10.1006/jcph.2000.6636]

Page, D.L., Koschan, A.F., Abidi, M.A., 2003. Perception-based 3D Triangle Mesh Segmentation Using Fast Marching Watersheds. IEEE Computer Society Conference on Computer Vision and Pattern Recognition'03, p.27-32.

Pauly, M., Gross, M., Kobbelt, L., 2002. Efficient Simplification of Point-Sampled Surfaces. Proc. IEEE Visualization'02, p.163-170.

Pauly, M., Keiser, R., Kobbelt, L., Gross, M., 2003. Shape modeling with point-sampled geometry. *ACM Trans. on Graphics*, **22**(3):641-650. [doi:10.1145/882262.882319]

Pauly, M., Kobbelt, L., Gross, M., 2006. Point-based multiscale surface representation. *ACM Trans. on Graphics*, **25**(2):177-193. [doi:10.1145/1138450.1138451]

Sethian, J., 1999. Level Set Methods and Fast Marching Methods. Cambridge University Press, Cambridge, UK, p.1-100. [doi:10.2277/0521645573]

Shamir, A., 2004. A Formulation of Boundary Mesh Segmentation. Proc. 2nd International Symposium on 3D Data Processing, Visualization, and Transmission, p.82-89. [doi:10.1109/TDPVT.2004.1335163]

Shlafman, S., Tal, A., Katz, S., 2002. Metamorphosis of polyhedral surfaces using decomposition. *Computer Graphics Forum*, **21**(3):219-228. [doi:10.1111/1467-8659.00581]

Sorkine, O., Lipman, Y., Cohen-Or, D., Alexa, M., Rossl, C., Seidel, H.P., 2004. Laplacian Surface Editing. ACM SIGGRAPH Symposium on Geometry Processing'04, p.179-188. [doi:10.1145/1057432.1057456]

Xu, M.H., Thompson, P.M., Toga, A.W., 2004. An adaptive level set segmentation on a triangulated mesh. *IEEE Trans. on Medical Image*, **23**(2):191-201. [doi:10.1109/TMI.2003.822823]

Yamauchi, H., Lee, S., Lee, Y., Ohtake, Y., Belyaev, A., Seidel, H.P., 2005a. Feature Sensitive Mesh Segmentation with Mean Shift. Shape Modeling International'05, p.236-243. [doi:10.1109/SMI.2005.21]

Yamauchi, H., Gumhold, S., Zayer, R., Seidel, H.P., 2005b. Mesh segmentation driven by Gaussian curvature. *The Visual Computer*, **21**(8-10):659-668. [doi:10.1007/s00371-005-0319-x]

Yamazaki, I., Natarajan, V., Bai, Z., Hamann, B., 2006. Segmenting Point Sets. IEEE International Conference on Shape Modeling and Applications'06, p.4-13. [doi:10.1109/SMI.2006.33]

Yu, Y., Zhou, K., Xu, D., Shi, X., Bao, H., Guo, B., Shum, H.Y., 2004. Mesh editing with Poisson-based gradient field manipulation. *ACM Trans. on Graphics*, **23**(3):644-651. [doi:10.1145/1015706.1015774]

Zhou, K., Snyder, J., Guo, B., Shum, H.Y., 2004. Iso-charts: Stretch-driven Mesh Parameterization Using Spectral Analysis. Eurographics Symposium on Geometry Processing'04, p.47-56. [doi:10.1145/1057432.1057439]

Zorin, D., Schroder, P., Sweldens, W., 1997. Interactive Multiresolution Mesh Editing. ACM SIGGRAPH'97, p.259-268. [doi:10.1145/258734.258863]

Zwicker, M., Pauly, M., Knoll, O., Gross, M., 2002. Pointshop 3D: An Interactive System for Point-based Surface Editing. ACM SIGGRAPH'02, p.322-329. [doi:10.1145/566570.566584]

Zwicker, M., Gotsman, C., 2004. Meshing Point Clouds Using Spherical Parameterization. Eurographics Symposium on Point-based Graphics'04. Zurich, p.173-180.