Computers & Graphics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

SMI 2011: Full Paper

CAD mesh model segmentation by clustering

Dong Xiao, Hongwei Lin *, Chuhua Xian, Shuming Gao

State Key Lab of CAD&CG, Zhejiang University, Hangzhou 310058, China

ARTICLE INFO

Article history:
Received 10 December 2010
Received in revised form
13 March 2011
Accepted 14 March 2011

Keywords:
CAD mesh segmentation
Hierarchical clustering
Mean shift
Mean curvature
Hough transformation

ABSTRACT

CAD mesh models have been widely employed in current CAD/CAM systems, where it is quite useful to recognize the features of the CAD mesh models. The first step of feature recognition is to segment the CAD mesh model into meaningful parts. Although there are lots of mesh segmentation methods in literature, the majority of them are not suitable to CAD mesh models. In this paper, we design a mesh segmentation method based on clustering, dedicated to the CAD mesh model. Specifically, by the agglomerative clustering method, the given CAD mesh model is first clustered into the sparse and dense triangle regions. Furthermore, the sparse triangle region is separated into planar regions, cylindrical regions, and conical regions by the Gauss map of the triangular faces and Hough transformation; the dense triangle region is also segmented by the mean shift operation performed on the mean curvature field defined on the mesh faces. Lots of empirical results demonstrate the effectiveness and efficiency of the CAD mesh segmentation method in this paper.

1. Introduction

CAD mesh models are ordinarily created from CAD data (including primitives, free-form surfaces, etc.). In general, the CAD data contains the information for segmentation. However, when we discretize the CAD data into CAD meshes, and store them in some standard format, such as STL format, the information for segmentation is lost.

Currently, the CAD mesh models have been widely employed in CAD/CAM applications, where, it is often required to recognize the features of the CAD mesh model. For example, in FEA mesh generation, to improve the mesh quality and reduce the computational load, the features in the mesh model need to be recognized, and some tiny features should be depressed. The first step of feature recognition is to segment the mesh model into meaningful parts.

Although there are lots of mesh segmentation methods in literature, the majority of them are suitable only to computer graphics mesh models, with dense mesh vertices and relatively uniform vertex distribution [1,2]. However, the CAD mesh models (Fig. 1) are quite different from the computer graphics mesh models.

As illustrated in Fig. 1, a CAD mesh model usually contains two typical regions, dense and sparse regions. Specifically, the sparse regions are generated from the triangulation of the planar, cylindrical and conical surfaces in a CAD model, while the dense regions are from the triangulation of the blending surfaces. The sparse region consists of many slender triangles, and the mesh quality of the sparse region is usually very *poor* in the sense of mesh segmentation methods for computer graphics models. Therefore, the mesh segmentation methods for graphics models are unable to be employed in CAD mesh model segmentation [3,4].

In this paper, we devise a method dedicated to the CAD mesh model segmentation. First, the triangles in the CAD mesh model are classified into sparse and dense regions by the agglomerative hierarchical clustering method [5]. To further segment the sparse region, the normalized normals of the triangle faces in the sparse region are mapped into the points on the Gauss sphere surface, and then the randomized Hough transformation [6] is employed to recognize the planes which intersect with the Gauss sphere in a circle. On the other hand, the dense region is further segmented by the mean shift operation [7–9] on the mean curvature field defined on the triangle faces. The method presented in this paper can automatically segment a CAD mesh model into meaningful parts, while the user can also improve the result conveniently by controlling several parameters.

The contributions of the CAD mesh segmentation method developed in this paper include:

- The CAD mesh model is classified into sparse and dense regions by the agglomerative hierarchical clustering method.
- The sparse region is partitioned into planar, cylindrical, and conical regions by the Gauss map and randomized Hough transformation.
- The dense region is segmented by performing the mean shift operation on the mean curvature field.

* Corresponding author.
E-mail addresses: xiaodong@cad.zju.edu.cn (D. Xiao), hwlin@cad.zju.edu.cn (H. Lin), xianchuhua@cad.zju.edu.cn (C. Xian), smgao@cad.zju.edu.cn (S. Gao).

0097-8493/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cag.2011.03.020

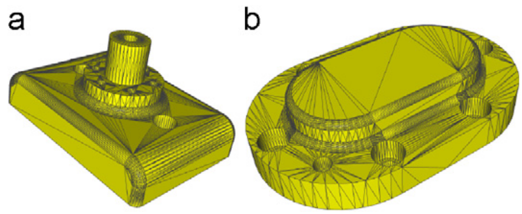

Fig. 1. The mesh quality of CAD mesh model is poor in the sense of mesh segmentation methods for computer graphics models, which contains sparse and dense regions. (a) and (b) are two typical CAD mesh models.

The rest of this paper is organized as follows. Section 2 surveys the related works. Section 3 introduces the sparse and dense region clustering method. In Section 4, the sparse region is further segmented by the randomized Hough transformation. In Section 5, the dense region is partitioned using the mean shift operation. Section 6 presents the results and the related statistical data and parameters. Finally, the conclusions and future work are proposed in Section 7.

2. Related works

The majority of the mesh segmentation methods are designed for segmenting the computer graphics mesh models, and there are only a few references on the CAD mesh segmentation till now.

2.1. CAD mesh segmentation

In Ref. [10], the CAD meshes are segmented by a region growing method and the following boundary rectification, based on the curvature tensor analysis. However, it is difficult to get the reasonable curvature tensor on the sparse region of a CAD mesh model, where the mesh vertices are just on the sharp boundaries (Fig. 1). More seriously, there are not principal directions at the umbilical points in the planar and spherical regions.

Noticing the CAD mesh model consists of sparse region and dense region, a heuristic method is developed in Ref. [3] to segment the CAD mesh model. Compared with the agglomerative clustering method employed in this paper, the heuristic method in [3] is very complex and not very robust.

In the other work on CAD mesh segmentation [4], the CAD mesh is first refined to make the sparse region dense enough, and then an improved watershed algorithm is performed on the refined mesh, to partition it into meaningful parts. Since lots of vertices are added in the mesh by mesh refinement, the computation cost of the method in [4] increases significantly.

2.2. Mean shift for segmentation

Mean shift is a non-parametric feature space analysis technique first presented in Ref. [7]. Recently, some mesh segmentation methods are presented by performing mean shift on different fields. Yamauchi et al. [11] employ mean shift to cluster the normals of triangles before segmentation, leading to the segmentation of the mesh model. Zhang et al. [12] propose a segmentation method in which the curvatures are first mean-shifted as they declared. But, actually, the mean shift in their method, is used to smooth the curvature at each vertex with its 2-neighbors till convergence, rather than cluster the curvature.

2.3. Hough transformation

Hough transformation is used to detect lines and curves by Duda et al. [13]. Xu et al. [6] propose the randomized Hough transformation, which has the same ability to detect curves with less computation overhead. Hough transformation can also be used to detect planes in 3D point cloud, and a data structure for this purpose is proposed in [14].

2.4. Computer graphics mesh segmentation

In this section, we briefly review the work on mesh segmentation suitable for computer graphics models. Mangan et al. [15] extend the watershed algorithm in 2D image segmentation to 3D mesh segmentation. A hierarchical segmentation method is proposed by Attene et al. [16] to fit the triangles with a set of primitives. Starting with some carefully chosen seeds, Lai et al. [17,18] develop a segmentation method using the concept of random walks, which is presented in image segmentation by Grady [19]. Recently, Skraba et al. [20] propose a mesh segmentation method that use persistence-based clustering.

For a comprehensive review on mesh segmentation, please refer to the excellent survey [2].

3. Dense and sparse region clustering

In this section, the triangles of the CAD mesh model is classified into dense and sparse regions by the agglomerative hierarchical clustering method [5]. Before the processing of the CAD mesh model segmentation, the length of the longest edge of its bounding box is scaled to 1.

As stated above, the CAD mesh model can be classified into sparse and dense regions (see Fig. 2). The sparse region consists of slender triangles, and the dense region is composed of small and relatively regular triangles. To distinguish the triangles in sparse and dense regions, we utilize the metric $m = s_a \times t_l$, where,

s_a: the area of a triangle;
t_l: the edge ratio between the longest and shortest edges of a triangle.

First, we compute m for each triangle, and then cluster them into two groups. Triangles in the group with smaller m are recognized as dense region, while the others are taken as sparse region. The detailed algorithm is listed as follows:

1. Calculate $m = s_a \times t_l$ for each triangle, and store them in a sorted list. Treat each one as a cluster initially.
2. Among all pairs of adjacent clusters, pick out the pair with the minimum distance and merge them to one cluster. The value

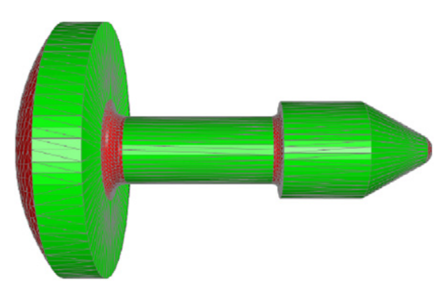

Fig. 2. The dense and sparse region clustering result of a CAD mesh model. The sparse region is in green, and the dense region is in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

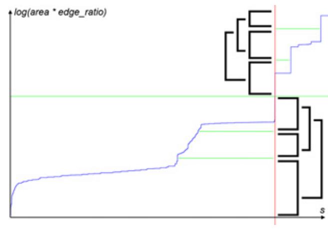

Fig. 3. The illustration of the agglomerative hierarchical clustering method for the model in Fig. 2. The blue curve is the Logarithm curve of *logm* versus *s*, the serials of the triangles in order. The green horizontal lines are the dividing lines for the data set *logm*, and the red vertical line is the corresponding dividing line for the triangles. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

of the new cluster is the arithmetic mean of all elements belongs to it. Insert the cluster back to the sorted list according to its value. Here, the distance d between two clusters with values m_1 and m_2, respectively, is defined as

$$d = |\log m_1 - \log m_2|.$$

3. Continue step 2 until there are only two clusters left. Thus, the triangles belongs to the cluster with smaller value are recognized as dense region, and the others as sparse region.

In Fig. 2, we illustrate the clustering result of a CAD mesh, where the sparse region is displayed in green, and the dense region in red. Fig. 3 is the curve of log$m$ versus s, the serials of the triangles in order, where the vertical axis represents log$m$, and the horizontal axis represents the serials of the triangles. Note that, the red vertical line in Fig. 3 is the dividing line of the two clusters.

The above clustering algorithm works well for the most CAD models in our experiments. Moreover, it is very easy for the users to improve the clustering results. In our implementation, the users can adjust the dividing line directly on the logarithm curve (Fig. 3), and the clustering result is updated interactively, till the users get the desired result.

It worth mentioning that, it is possible for some small triangles in a planar patch to be clustered into the dense region, which are easy to be merged in the adjacent planar patches in the following process.

In some CAD mesh models, there are a few very tiny triangles which can worsen the clustering result. Then, as the pre-processing, we delete the data corresponding to the tiny triangles with area below 10^{-5} from the initial clusters.

4. Segmentation of the sparse region

In CAD mesh model, the sparse region is generally composed of three type regions, i.e., planar region, cylindrical region and conical region. In this section, the sparse region is further segmented by Gauss mapping, randomized Hough transformation, and a followed merging operation. For clarity, we list the main steps of the sparse region segmentation algorithm in Algorithm 1, and explain them in detail in the following sections.

Algorithm 1. Mesh segmentation Algorithm by the Randomized Hough Transformation.

1 Recognize the initial planar patches, and map them to the Gauss sphere surface;
2 Determine the most possible plane on the Gauss sphere surface by the randomized Hough transformation, which intersects with the Gauss sphere in a circle;
3 On the Gauss sphere surface, search the data points on the plane (and in the circle), and merge the planar patches of the CAD mesh model, which correspond to these data points;
4 Repeat steps 2 and 3 till there are less than six points left on the Gauss sphere surface, or there are less than six points on the new determined plane.

4.1. Initial planar patch recognition

As the pre-processing, we first compute the normalized normal of each triangular face in the sparse region, and merge the adjacent faces with inner product of their normals above α_p into a single planar patch, which are taken as the units in the Hough transformation and the following merging operation (see Fig. 4). It should be pointed out that, the triangles in the dense region are also merged into their adjacent planar patch, if the inner product of their normalized normals is greater than α_p, and their common edge is not the shortest edge of the needle triangle in the sparse region, with edge ratio greater than 15.0. In our implementation, we take $\alpha_p = (1-10^{-5})$.

Finally, we map the normalized normals of these merged planar patches and non-merged triangular patches into the Gauss sphere. The planar region is mapped to a point on the Gauss sphere surface, and the cylindrical and conical regions are mapped to great and small circles on the Gauss sphere surface, respectively.

4.2. Determination of the plane on the Gauss sphere surface

As stated above, the cylindrical and conical regions are mapped to great and small circles on the Gauss sphere surface, each of which is the intersection between a plane and the Gauss sphere. In this section, the randomized Hough transformation [14] is employed to recognize which points on the Gauss sphere lie in a plane. To perform the randomized Hough transformation, the Cartesian coordinates system ($\mathbf{o}$;x,y,z) is established, taking the center of the Gauss sphere as the origin.

Supposing a point $\boldsymbol{p} = (p_x, p_y, p_z)$ is on a plane C, with normalized normal,

$$\boldsymbol{n} = (n_x, n_y, n_z) = (\sin\varphi\cos\theta, \sin\varphi\sin\theta, \cos\varphi), \quad (1)$$

the distance ρ from the origin to the plane C is,

$$\rho = \boldsymbol{p} \cdot \boldsymbol{n} = p_x\sin\varphi\cos\theta + p_y\sin\varphi\sin\theta + p_z\cos\varphi, \quad (2)$$

where φ is the angle between the normal $\boldsymbol{n}$ and the z-axis, and θ is the angle between the x-axis and the projection of $\boldsymbol{n}$ on the $x-y$ plane. Each point in the (φ,θ,ρ)-space corresponds to a plane in the ($\mathbf{o}$; x,y,z) space. Since the Gauss sphere is a unit sphere, the distance from the origin to the plane intersecting with the Gauss sphere is less than 1, i.e., $\rho < 1$. Therefore, in the

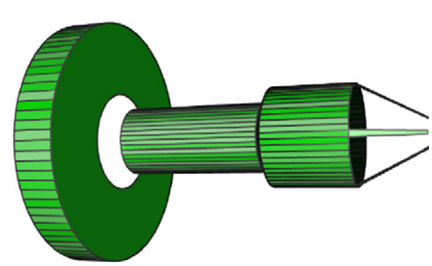

Fig. 4. The initial merged planar patches on the CAD mesh model after pre-processing, which are taken as the units in the Hough transformation and the followed merging operation, together with the non-merged triangular faces in the sparse region.

(φ,θ,ρ)-space, we construct a unit sphere U_h for the randomized Hough transformation.

Next, three different points are chosen randomly from the data points on the Gauss sphere surface. They construct a plane, corresponding to a point (φ,θ,ρ) in the unit sphere U_h.

If two planes with $(\boldsymbol{n}_1,\rho_1)$ and $(\boldsymbol{n}_2,\rho_2)$, which corresponds to the two points $(\varphi_1,\theta_1,\rho_1)$ and $(\varphi_2,\theta_2,\rho_2)$, satisfy the following conditions,

1. the angle between their normals (1) is less than 1°, and
2. $|\rho_1-\rho_2|<0.01$,

the two points are merged into a single point in the (φ,θ,ρ)-space in the following manner. Suppose $(\varphi_1,\theta_1,\rho_1)$ (equivalent to $(\boldsymbol{n}_1,\rho_1)$) is the old point, which is merged by k_h points, and the point $(\varphi_2,\theta_2,\rho_2)$ (equivalent to $(\boldsymbol{n}_2,\rho_2)$) is a new point, which needs to be merged in the old point. Then, the new merged point is the point in the (φ,θ,ρ)-space, which corresponds to

$$\left(\frac{k_h\boldsymbol{n}_1+\boldsymbol{n}_2}{k_h+1},\frac{k_h\rho_1+\rho_2}{k_h+1}\right).$$

Note that, the thresholds in this section are fixed because both the Gauss sphere and U_h are unit spheres.

We repeat the point selection and plane calculation, till over one thousand points are merged into a single point (φ,θ,ρ) in the unit sphere U_h. The point corresponds to a plane C in the $(\mathbf{o};x,y,z)$ space which intersects the Gauss sphere in a circle.

Furthermore, we search the data point set $X=\{(x_i,y_i,z_i), i=1,2,\ldots,k_g\}$ on the Gauss sphere surface, which satisfy

$$|x_i\sin\varphi\cos\theta+y_i\sin\varphi\sin\theta+z_i\cos\varphi-\rho|<0.035, \tag{3}$$

and take them as the points on the plane C (see Fig. 5(a)). If the circle in which the plane C intersects with the Gauss sphere is a great circle, the planar patches corresponding to these data points X_g consist of a cylinder; otherwise, they constitute a cone.

It should be pointed out that, the recognized data points in X_g do not take part in the next step of the randomized Hough transformation. But after a new plane is determined, they are still checked by Eq. (3) to see whether they lie in the plane.

4.3. Merge the planar patches

In the last section, a set of data points X_g in a circle on the Gauss sphere surface are found, which correspond to a set X_m of planar patches in the original CAD mesh model. However, it is possible for the set of planar patches that,

1. the planar patches in X_m constitute several patches which are separated in geometry;
2. the planar patches consists of several cylinder or cone with different but parallel axes;
3. there are true planar patches in X_m.

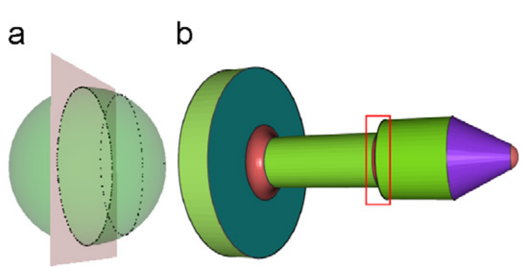

Fig. 5. The data points in a great circle on the Gauss sphere surface (a) correspond to the cylindrical regions (in green) on the CAD mesh model in (b), while the data points in a small circle correspond to the conical regions in purple in (b). (a) The Gauss sphere and a plane which intersects the sphere in a great circle; (b) the segmentation result of the sparse region. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Therefore, we should trace and merge the planar pathes in X_m, and determine the potential true planar patches.

In our implementation, we regard the adjacent planar patches as lying in different cylinders or cones with parallel axes, if the angle between them is larger than the threshold α_{angle}. We also take the planar patch, between which and its adjacent planar patch the area ratio (larger area over small area) is larger than the threshold α_{area}, as the potential true planar patch. In our implementation, we choose $\alpha_{angle}=20^\circ$, and $\alpha_{area}=5$, except in Figs. 8(b) and 11, $\alpha_{area}=2.5$.

Thus, we iteratively merge the adjacent patches into one patch, if they satisfy,

1. the angle between them is less than α_{angle}, and,
2. the ratio between their areas (larger area over smaller area) is less than α_{area}.

At the end, if there are some planar patches left in X_m which are not merged with its adjacent patches. They are processed by the following strategies.

1. If the non-merged planar patch P_s in X_m is smaller than its adjacent merged patch P_a, and the angle between them is smaller than α_{angle}, P_s is merged to P_a;
2. If the non-merged planar patch P_s in X_m is smaller than its two adjacent merged patches P_a and P_b, and the two angles between P_s and P_a, P_s and P_b are both smaller than α_{angle}, the three patches are merged into a whole patch;
3. If the non-merged planar patch P_s in X_m is larger than its adjacent patches, we take P_s as the potential true planar patch.

After the algorithm terminates, the non-merged potential true planar patches becomes the true planar patches.

In this manner, the planar regions, cylindrical regions, and conical regions can be segmented successfully for most CAD mesh models. Fig. 5(b) is the sparse region segmentation result for the CAD model in Fig. 2.

5. Segmentation of the dense region

In the CAD mesh models, the dense region consists of small and relatively regular triangles, which are generated by triangulating the blending surfaces in general. In some CAD models, the dense regions are isolated by the sparse regions, each of which is generated from a single blending surface, and does not require further segmentation. However, in some cases, a single dense region is generated from several connected blending surfaces (such as the model in Fig. 6), which needs further segmentation.

In CAD models, the simple quadratic surfaces is frequently used as the blending surface, such as the spherical surface, torus, etc. Note that the curvature distribution on the same quadratic surface is uniform, we segment the dense region into the sub-regions, each of which is generated from a single blending surface, by performing mean shift clustering operation on the mean curvature field defined on the dense region.

It should be pointed out that, the mean shift method has been employed in segmenting graphical models by performing the mean shift operation on the normal field defined on the mesh faces [11]. However, the mean shift on the normal field is easy to oversegment a whole quadratic region (such as spherical surface, torus, etc.) into multiple regions (Fig. 6(b)), while the mean shift

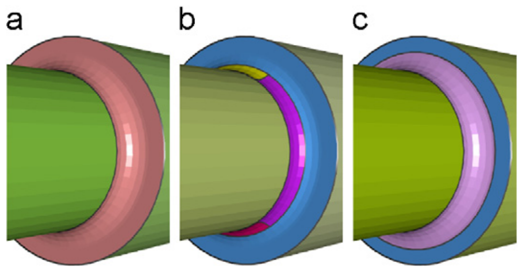

Fig. 6. The comparison between the results of the mean shift on the normal field [11] (b), and on the mean curvature field (c). (a) is the original dense region, which is the zoom of the part in the red box of the model in Fig. 5(b). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

on the mean curvature field can avoid the over-segmentation by choosing suitable parameters (Fig. 6(c)).

5.1. Mean curvature computation

We have implemented several well-known discrete mean curvature computation methods, it is found that the method mentioned in [21] (originally presented in [22]) works well in most cases for the computation of the mean curvature on the barycenter of each triangle.

Denote the barycenter of a triangle as $\boldsymbol{P}$. Taking $\boldsymbol{P}$ as the center, we construct a sphere s with radius r. The sphere s is partitioned into two parts, s_{in} and s_{out}, by the surface of the CAD mesh model, where, s_{in} is inside the CAD model, and s_{out} is out of the CAD model. Then, the surface area of s_{in} is

$$A_{in}(r,\boldsymbol{P}) = 2\pi r^2 - \pi H(\boldsymbol{P}) r^3 + O(r^4), \tag{4}$$

where $H(\boldsymbol{P})$ is the mean curvature at the triangle barycenter $\boldsymbol{P}$. By omitting the high degree term, we have

$$H(\boldsymbol{P}) = \frac{1}{\pi r^3}(2\pi r^2 - A_{in}(r,\boldsymbol{P})). \tag{5}$$

In our implementation, the radius r is taken as $\beta_r l_r$, i.e.,

$$r = \beta_r l_r, \tag{6}$$

where l_r is the average length of edges in the current connected group of the dense region, and β_r is a multiple varying in the interval [1.0, 2.0]. In the smooth region, any $1.0 \le \beta_r \le 2.0$ leads to good results. However, in the sharp boundary, where it is only C^0 or C^1 continuous, and the mean curvature has no definition, β_r should be carefully selected to generate desirable segmentation. The value of β_r for each example presented in this paper is listed in Table 2.

To compute the surface area of s_{in}, we first triangulate the whole spherical surface into 80 uniform spherical triangles. Moreover, the spherical triangles intersecting with the boundary of s_{in} are refined thrice. Then, the sum of the spherical triangles inside the surface of the CAD model and intersecting with the boundary of s_{in} is taken as the surface area $A_{in}(r,\boldsymbol{P})$ of s_{in} approximately.

For more details on the computation of mean curvature, please refer to [21,23].

5.2. Mean shift on the mean curvature field

After computing the mean curvature c_i at the center $\boldsymbol{P}_i$ of each triangle in the dense region T, they constitute the mean curvature field $\chi = \{\boldsymbol{x}_i = (\boldsymbol{P}_i, c_i),\ i = 1,2,\ldots,n\}$ in $\mathbb{R}^4$. The mean shift [11] is performed on χ to segment the dense region T into meaningful sub-regions.

Specifically, denoting $\boldsymbol{x} = (\boldsymbol{P}, c)$, the mean shift vector $\boldsymbol{m}(\boldsymbol{x})$ is taken as

$$\boldsymbol{m}(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} \boldsymbol{x}_i g_g\left(\left\|\frac{\boldsymbol{P}-\boldsymbol{P}_i}{h_g}\right\|^2\right) g_c\left(\left\|\frac{c-c_i}{h_c}\right\|^2\right)}{\sum_{i=1}^{n} g_g\left(\left\|\frac{\boldsymbol{P}-\boldsymbol{P}_i}{h_g}\right\|^2\right) g_c\left(\left\|\frac{c-c_i}{h_c}\right\|^2\right)} - \boldsymbol{x}. \tag{7}$$

Here, we adopt the triangular kernel:

$$k_T(x) = \begin{cases} 1-x, & 0 \le x \le 1, \\ 0, & x > 1, \end{cases} \tag{8}$$

and then, $g_g(x)$ and $g_c(x)$ in Eq. (7) are defined as

$$g_g(x) = g_c(x) = -k'_T(x) = \begin{cases} 1, & 0 \le x \le 1, \\ 0, & x > 1. \end{cases} \tag{9}$$

Furthermore, h_g and h_c in Eq. (7) are the bandwidths for $\{\boldsymbol{P}_i\}$ and $\{c_i\}$, respectively. In our implementation, we set

$$h_g = \beta_g l_m, \quad h_c = \beta_c / l_m, \tag{10}$$

where l_m is the longest edge length of the bounding box of the current connected triangles in the dense region, and β_g and β_c are two multiples. They should be carefully chosen for each CAD mesh model for desirable segmentation. We will explain the selection of β_g and β_c in Section 6.

The mean shift algorithm is composed of the following steps:

1. Initialize $\boldsymbol{y}_i^{[0]}$ with $\boldsymbol{x}_i$, $i = 1,2,\ldots,n$;
2. Compute

$$\boldsymbol{y}^{[j+1]} = \boldsymbol{y}^{[j]} + \boldsymbol{m}(\boldsymbol{y}^{[j]}), \quad j = 0,1,2,\ldots \tag{11}$$

till it converges to several points $\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_k$.

Finally, all of the points $\boldsymbol{x}_i$ which have the same convergence point $\boldsymbol{g}_j$ are taken as one cluster, and the triangles they correspond to are segmented as one or more regions according to their connectivity. Fig. 6(c) is the segmentation result by performing the mean shift operation on the mean curvature field.

As mentioned in Ref. [11], to speed up the mean shift procedure, we use a k-D tree to store the geometry information $\{\boldsymbol{P}_i, i = 1,2,\ldots,n\}$, and check the mean curvature difference between two triangles only when their distance $\|\boldsymbol{P}_i - \boldsymbol{P}_j\|$ is not more than h_g.

6. Implementation and results

The CAD mesh segmentation algorithm proposed in the paper has been implemented in Visual C++ 2008 environment. And lots of CAD mesh models have been tested with it. All examples in this paper run on a PC with 2.53 GHz CPU (Intel Core Duo) and 4 G RAM, under Windows Vista 64-bit system.

Fig. 7 shows the segmentation results for two CAD models where the two original CAD mesh models are shown in Fig. 1,

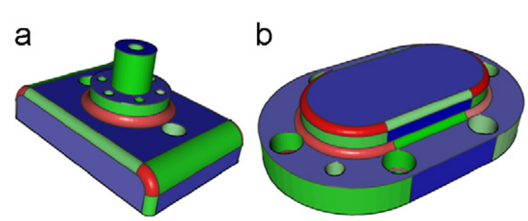

Fig. 7. Segmentation results of two CAD mesh models with a bit noise in the cylindrical regions. The original triangular mesh models are presented in Fig. 1, respectively.

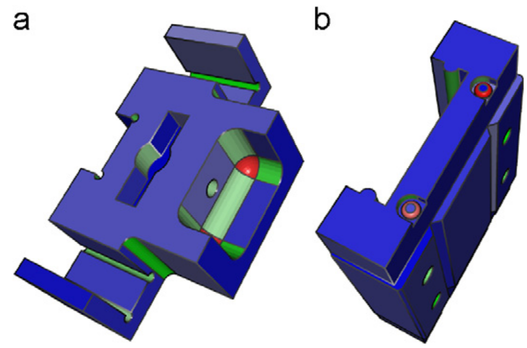

Fig. 8. Segmentation results of two CAD mesh models. The sparse and dense region clustering result in (b) is adjusted manually.

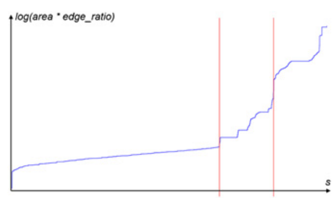

Fig. 9. Logarithm curve of log$m$ to the serials of the ordered triangles for the mesh model in Fig. 8(b). The right red vertical line is the dividing line automatically generated by the clustering algorithm, and the left red line is the dividing line selected manually. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

respectively. Although there are a bit noise in the cylindrical regions (Fig. 1), they are recognized successfully by the randomized Hough transformation (Fig. 7).

Fig. 8 shows the segmentation results for the other two CAD mesh models. For all of the four CAD mesh models in Figs. 7 and 8, the single connected dense region on them is generated from a single blending surface, therefore, the dense regions on them do not need to be further segmented after the sparse and dense region clustering.

In all of the examples in this paper, except the example in Fig. 8(b), the dividing lines (Fig. 3) for the sparse and dense region clustering are determined automatically. In the example in Fig. 8(b), because the sparse triangles in the small features have similar areas with the dense ones, and the clustering algorithm cannot distinguish them well, the dividing line should be adjusted manually to generate desirable segmentation result. Fig. 9 demonstrates the logarithm curve and the automatically generated and manually selected dividing lines. This example shows that, even the agglomerative hierarchical clustering algorithm fails to generate a good dividing line, it is still easy for the user to adjust it on the logarithm curve, and the clustering result can be updated in real-time.

In some special cases, especially when the CAD model is composed mainly of cylindrical, conical and planar regions, the mesh segmentation algorithm by the randomized Hough transformation (Algorithm 1) can directly partition the whole CAD mesh model well, without the sparse and dense region clustering. The examples demonstrated in Fig. 10 is generated just by Algorithm 1, without the sparse and dense region clustering. Note the very narrow conical regions on the two models in Fig. 10(a and b), which are hard to be recognized as ruled regions by the heuristic method presented in Ref. [3].

Fig. 11 illustrates a segmentation result, where the dense region is further segmented by the mean shift on the mean curvature field. Fig. 11(a) is the original CAD mesh. Fig. 11(b) is the segmentation result, a connected dense region is partitioned into three sub-regions, demonstrated in Fig. 11(c).

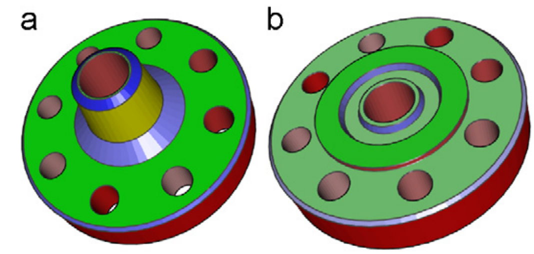

Fig. 10. Two segmentation results just by Algorithm 1. Note that the very narrow conical regions on the two models in (a) and (b) are segmented successfully by Algorithm 1.

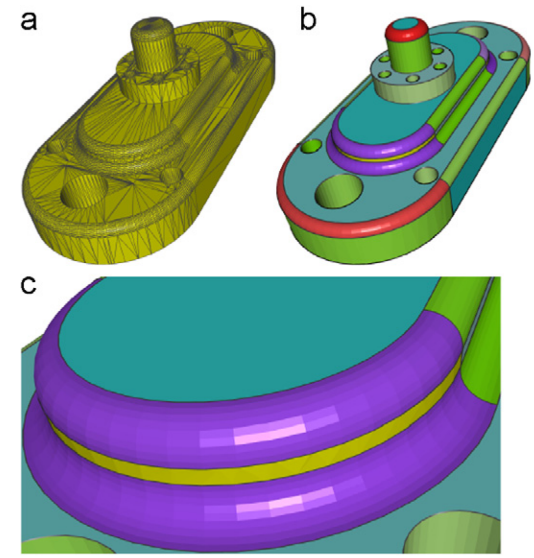

Fig. 11. The mean shift operation is performed on the mean curvature field to segment a dense region into sub-regions. (a) The original CAD mesh model; (b) the segmentation result; (c) zoom of the segmented dense region by the mean shift.

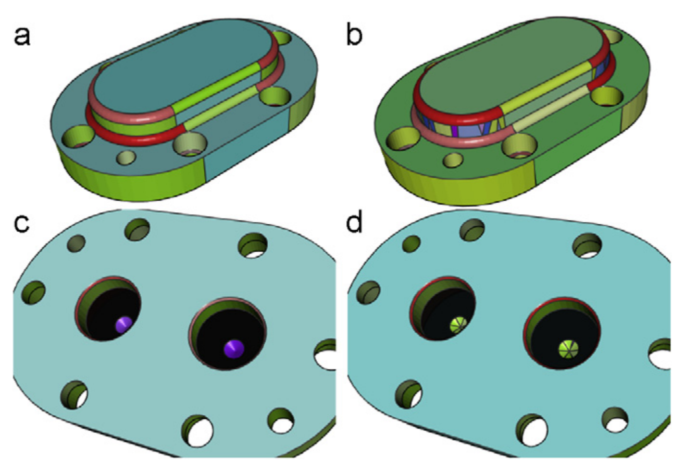

Fig. 12. Comparison between our method and the method in [3] on a CAD mesh model. (a,c) Two views of the segmentation results by our method. (b,d) Two possible segmentation results by the method in [3].

In Fig. 12, both our method and the method in [3] are employed to segment a CAD mesh model, where (a) and (c) are the segmentation results by our method from two different views. As a comparison, (b) and (d) are two possible results by the method in [3]. In [3] (Section 6.2.2 (B), in step 1 (i)), there is a threshold which controls the maximum distance from the constrained plane to the normals of triangles in cylindrical regions.

Table 1
Statistics for the CAD mesh segmentation examples.

Fig.	f	f_d	r_s	r_d	T_c	T_s (s)
5	3466	2878	5	4	0.025 s	0.057
8(a)	1124	302	50	2	0.006 s	0.067
8(b)	3612	2380	64	4	0.005 s	0.069
7(a)	4098	2376	34	11	0.026 s	0.759
7(b)	6576	3752	37	12	0.039 s	1.492
10	2912	0	24	0	N/A	1.227
11	6798	4708	37	5	0.049 s	0.447

Table 2
Parameters and times used to segment dense regions.

Fig.	n_f	β_r	β_g	β_c	T_u (s)	T_m (s)
6(c)	737	2.0	0.3	4.0	4.630	0.221
11(c)	1094	0.75	0.3	3.0	2.451	0.362

When this threshold is set to 0.003, their method produces the result in Fig. 12(b), where the cylindrical region are over-segmented. When the threshold is set to 0.03, their method generates the result in Fig. 12(d), where the conical regions are over-segmented. We tried to adjust the threshold in [3], but at least one part of the model is over-segmented. This example shows that our method is more robust than the method in [3].

Moreover, Table 1 presents the statistical data for the CAD mesh segmentation examples shown in this paper. In the table, f is the number of faces in the CAD mesh model. f_d is the number of triangles in the dense regions, r_s is the number of sparse regions after sparse region segmentation, and r_d is the number of dense regions isolated by the sparse regions. Finally, T_c is the time spent on the sparse and dense region clustering; T_s is the time spent on the sparse region segmentation. It can be seen that these two steps cost a little time.

Table 2 shows the parameters and time used to segment the dense regions. Here n_f is the number of triangles in the dense region illustrated in the corresponding figures. β_r is the multiple in determining the radius of the sphere for computing the mean curvature (Eq. (6)). β_g and β_c are the multiples for calculating the bandwidths h_g and h_c in the mean shift operation (Eq. (10)). T_u is the time spent on mean curvature computation, and T_m is the time spent on mean shift clustering.

As stated above, β_r usually varies from 1.0 to 2.0 depending on the smoothness of the region. We choose $\beta_r = 0.75$ in Fig. 11(b) to exclude the influence of the neighboring holes. The general rule for the selection of β_c is that, if the dense region is over-segmented into too many regions, β_c should be increased; if it is segmented into too few regions, β_c should be decreased. On the other hand, it usually leads to good results by setting $\beta_g = 0.3$, if the region is circular.

7. Conclusions

In this paper, we develop a CAD mesh segmentation algorithm. The triangles in the CAD mesh model is first partitioned into sparse and dense regions by the agglomerative hierarchical clustering algorithm. Furthermore, the sparse region is segmented into planar regions, cylindrical regions, and conical regions by the randomized Hough transformation and the followed merging operation. On the other hand, the dense region is partitioned into meaningful sub-regions, by performing the mean shift on the mean curvature field. The segmented CAD mesh model can be used in feature recognition and downstream CAD/CAM applications, such as automated process planning and sheet metal tool design, high-quality finite element analysis (FEA) mesh generation, and redesign/reuse of feature shapes for product design, etc.

Acknowledgments

We thank Prof. Hiromasa Suzuki for giving us the model in Fig. 2. This paper is supported by Natural Science Foundation of China (nos. 60970150, 60736019, 60933008), and Zhejiang Provincial Natural Science Foundation of China (no. Y1090416).

References

[1] Attene M, Katz S, Mortara M, Patane G, Spagnuolo M, Tal A. Mesh segmentation—a comparative study. In: IEEE international conference on shape modeling and applications, SMI 2006. IEEE; 2006. p. 7.
[2] Shamir A. A survey on mesh segmentation techniques, In: Computer graphics forum, vol. 27, Wiley Online Library. p. 1539–56.
[3] Sunil V, Pande S. Automatic recognition of features from freeform surface CAD models. Computer-Aided Design 2008;40:502–17.
[4] Gao S, Zhao W, Lin H, Yang F, Chen X. Feature suppression based CAD mesh model simplification. Computer-Aided Design 2010;42:1178–88.
[5] Lattin JM, Carroll JD, Green PE. Analyzing multivariate data. 1st ed. Pacific Grove, CA: Duxbury; 2003.
[6] Xu L, Oja E, Kultanen P. A new curve detection method: randomized Hough transform (RHT). Pattern Recognition Letters 1990;11:331–8.
[7] Fukunaga K, Hostetler L. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory 1975;21:32–40.
[8] Cheng Y. Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence 1995:790–9.
[9] Comaniciu D, Meer P. Mean shift: a robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 2002;24:603–19.
[10] Lavoué G, Dupont F, Baskurt A. A new cad mesh segmentation method, based on curvature tensor analysis. Computer Aided Design 2005;37:975–87.
[11] Yamauchi H, Lee S, Lee Y, Ohtake Y, Belyaev A, Seidel H. Feature sensitive mesh segmentation with mean shift. In: International conference on shape modeling and applications. IEEE; 2005. p. 236–43.
[12] Zhang X, Li G, Xiong Y, He F. 3D mesh segmentation using mean-shifted curvature. Advances in Geometric Modeling and Processing 2008:465–74.
[13] Duda R, Hart P. Use of the Hough transformation to detect lines and curves in pictures. Communications of the ACM 1972;15:11–5.
[14] Borrmann D, Elseberg J, Lingemann, K, Nüchter A. A data structure for the 3d Hough transform for plane detection. In: Proceedings of the 7th IFAC symposium on intelligent autonomous vehicles (IAV '10), Lecce, Italy, 2010.
[15] Mangan A, Whitaker R. Partitioning 3D surface meshes using watershed segmentation. IEEE Transactions on Visualization and Computer Graphics 1999;5:308–21.
[16] Attene M, Falcidieno B, Spagnuolo M. Hierarchical mesh segmentation based on fitting primitives. The Visual Computer 2006;22:181–93.
[17] Lai Y, Hu S, Martin R, Rosin P. Fast mesh segmentation using random walks. In: Proceedings of the 2008 ACM symposium on solid and physical modeling, ACM; 2008. p. 183–91.
[18] Lai Y, Hu S, Martin R, Rosin P. Rapid and effective segmentation of 3D models using random walks. Computer Aided Geometric Design 2009;26:665–79.
[19] Grady L. Random walks for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 2006:1768–83.
[20] Skraba P, Ovsjanikov M, Chazal F, Guibas L. Persistence-based segmentation of deformable shapes. In: IEEE computer society conference on computer vision and pattern recognition workshops (CVPRW). IEEE; 2010. p. 45–52.
[21] Pottmann H, Wallner J, Yang Y, Lai Y, Hu S. Principal curvatures from the integral invariant viewpoint. Computer Aided Geometric Design 2007;24: 428–42.
[22] Cazals F, Chazal F, Lewiner T. Molecular shape analysis based upon the Morse–Smale complex and the Connolly function. In: Proceedings of the nineteenth annual symposium on computational geometry. ACM; 2003. p. 351–60.
[23] Pottmann H, Wallner J, Huang Q, Yang Y. Integral invariants for robust geometry processing. Computer Aided Geometric Design 2009;26:37–60.

Please cite this article as: Xiao D, et al. CAD mesh model segmentation by clustering. Computers and Graphics (2011), doi:10.1016/j.cag.2011.03.020