Computer-Aided Design 55 (2014) 13-25

Contents lists available at ScienceDirect

Computer-Aided Design

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

Shape recognition of CAD models via iterative slippage analysis*

Bing Yi^a, Zhenyu Liu^{a,*}, Jianrong Tan^a, Fengbei Cheng^a, Guifang Duan^a, Ligang Liu^b

^a State Key Lab of CAD&CG, Zhejiang University, Hangzhou, 310027, China

^b University of Science and Technology of China, Hefei, 230026, China

HIGHLIGHTS

- We present a new shape recognition method by iterative slippage analysis.
- The exact normal is found to be one of the key points for slippage analysis.
- The appropriate region is found to be the other key points for slippage analysis.
- A knowledge guided region growing method is used to get the appropriate region.
- An iterative normal modification method is used to obtain the exact normal.

ARTICLE INFO

Article history: Received 31 July 2012 Accepted 28 April 2014

Keywords: Shape recognition Slippage analysis Basic primitives

ABSTRACT

A new slippage analysis method for recognizing basic primitive surfaces of CAD models is presented in this paper. Obtaining the exact normal and searching the appropriate local region of each point are found to be the key steps for determining the local slippage motion type. First, the tensor voting-based boundary point recognition method is integrated to preprocess the original points. Then, the local slippage analysis method is used to initialize the point type. Furthermore, the appropriate region of each point is acquired by the region growing method. Meanwhile, the middle level information (the basic primitive surface types and the representative parameters) is found, guiding the modification of the normal of each point and the iterative detection of the surface types. Finally, the middle level information-based smooth method is introduced to refine the boundary of each basic primitive surface. The empirical results show that the proposed algorithm is efficient and robust for recognizing primitive shapes from CAD models of mechanical parts.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

With the rapid development of 3D digital data acquisition devices, we can obtain all the point cloud of 3D models easily and rebuild the triangular meshes with high accuracy. This is recently called reverse engineering, which provides a new way to create massive complex 3D models. However, due to the loss of middle level and high level model information (symmetry, parallelism, perpendicularity, etc.), such meshes are too raw to be directly used in the subsequent processes, such as CAD model reconstruction, and convergent-type CAE. With the virtual explosion in the amount of raw data available for designer, the critical problem shifts to obtain the middle level and high level information through this

E-mail address: liuzy@zju.edu.cn (Z. Liu).

http://dx.doi.org/10.1016/j.cad.2014.04.008 0010-4485/© 2014 Elsevier Ltd. All rights reserved. data and adopt it to improve the efficiency for redesigning a new product [1]. The key processes to detect the middle level and high level model information from existing 3D point clouds are mesh segmentation and shape recognition [2–6].

Recently, a lot of researches have been done on shape segmentation and recognition. Most previous approaches are based on the local model information like curvature or boundary detection. However, even with the segmented surface patches, it is also difficult to get the middle level model information, let alone the high level model information. Realizing that a lot of mechanical parts consist of basic primitives, such as plane, sphere, cylinder, cone, extrusion, revolution, helix, nowadays, many researchers have attempted to segment the CAD models into basic primitives [7–13]. The existing primitive shape fitting approaches likely confront the following problems: (1) hardly can recognize all the basic primitives, such as plane, sphere, cylinder, cone, extrusion, revolution, and helix; (2) hardly can obtain the middle level information of the basic primitives, such as the center and radius of a sphere, the normal and position of a plane, etc; (3) sensitivity to numerical





CrossMark

This paper has been recommended for acceptance by Karthik Ramani, PhD.
 * Corresponding author.

noise inherently embedded within the obtained 3D point clouds. Therefore, we propose a new slippage analysis method to robust recognize shape primitives of mechanical parts. We discover that the key processes for slippage shape segmentation are the exact normal and the appropriate region selection of each point. The existences of noise in point cloud obtained by 3D digital data acquisition devices and sharp edges in mechanical objects make it difficult to find the exact normal of each point. Thus, we introduce the tensor voting method to classify the point type into a plane, a sharp edge and a corner. Meanwhile, the detected point type improves the efficiency of the surface type segmenting and recognizing. Furthermore, we use the knowledge-based region growing method to get the exact region for each primitive patch, and then adopt the extended RANSAC method to obtain the middle level information, which iteratively guides the normal modification and the shape recognition.

2. Related work

Mesh segmentation has been extensively studied in the past years. The goal of segmentation is to cluster the mesh model into meaningful parts. However, it is hard to propose a segmentation approach that can segment all kinds of models into appropriate parts, due to the application field difference [14,15]. In computer graphics, the region growing method, the watershed method, the K-means method, the mesh shift method, the shape diameter function method, and the rand walk method have been well studied [16-21]. But these approaches cannot be directly applied to segment mechanical parts. Várady et al. [2] provide a detailed survey in the reconstruction of mechanical parts and underline that the surface type recognition and surface fitting are the specific issues for reconstructing a B-rep model. Agathos et al. [22] present an exhaustive overview of 3D mesh segmentation both on surfacebased methods, which segment the part into basic primitives. and on volume-based methods, which segment the part into different volumes or features. We roughly group the related works into three categories: low level information-based surface segmentation, middle level and high level information-based shape classification, and robust shape recognition.

Low level information-based surface segmentation. Many traditional segmentation methods use the local level information such as the Gaussian curvature or the mean curvature to segment the model. One of the most popular segmentation methods is region growing. This method selects a set of seed points and merges the neighbor points to a patch which has the same local properties, such as principle curvatures. However, it is difficult to find the exact seeds and there will be over segmentation in the transition of two patches [23,24]. Lavoué et al. [25,26] extend this algorithm by using a robust curvature tensor to guide the region growing and introduce a boundary score to rectify the patch boundaries. Kim et al. [27] introduce the tensor voting-based mesh segmentation method. The point type is recognized first by the robust tensor voting theory. Then, the mesh data with additional attribute such as color information is clustered into several patches by the *k*-means algorithm. However, the local information-based region growing method cannot segment all the basic primitives, and it is much difficult to obtain the middle level and high level information of mechanical parts.

Middle level and high level information-based shape classification. Recently, many researches have realized that shape features play important roles in the segmentation of man-made objects. Cohen-Steiner et al. [28] propose a segmentation algorithm, called variation shape approximation. This algorithm iteratively fit planes and partition triangles to the regions until the convergence. Wu et al. [29] extend this algorithm by using not only planes, but also cylinders, spheres, and rolling ball surfaces for the fittings. Yan et al.

[11,12] propose an iterative method for mesh segmentation by fitting quadric surfaces. However, these methods partition the entire surface into parts approximated by shape proxies; thus the complex surface will also be assigned primitives. Attene et al. [8,9] introduce a hierarchical mesh segmentation method to detect primitive geometries. The algorithm generates a binary tree of clustering and iteratively merges the local neighbor points into one single primitive cluster based on the approximation method. These fitting approaches require users to input the number of regions first, but it is difficult to find the appropriate number of regions before mesh segmentation. Protopsaltis et al. [30] introduce the planar cross section to reconstructing CAD models from the point clouds. However, the cross section should be along a principal axis, and the feature intersections and cross section along a sweep trajectory are ignored. Sellamani et al. [31] propose a robust method to approximate sweep shape by using prominent cross section. Goyal et al. [32] adopt the prominent cross section to extract high level, volumetric information from mesh models. However, it is difficult to reconstruct a mechanical part with multiple sweep components and intersections between them. Furthermore, the manufacture and function features of mechanical parts are composed of basic geometry primitives; thus, the classified sweep components need further process to be converted to semantic manufacture features.

Robust primitive shape recognition. Most of traditional mesh segmentation methods aim to segment the surface into different parts, and some try to classify the mesh by the surface type. However, how to robust recognize the primitive shape from point clouds is challenging. Benjamin et al. [33] propose a heat walk algorithm to segment triangle meshes, which is robust to a variety of noise factors. Fang et al. [34] adopt the heat mean signature to robust segment the surface which satisfies the perceptually consistent mesh segmentation conditions. However, the heat kernel-based method cannot be used to segment the mechanical parts into basic primitives. Golovinskiy et al. [35] propose a system for recognizing objects in 3D point clouds of urban environments. The graphcut algorithm is adopt to segment the point clouds into different patches, and then a trained classifier model is used to recognize the segmented patches. However, it is difficult to propose an appropriate shape features that can recognize all the objects from the point clouds. Lafarge et al. [36] proposed a multi-label Markov Random Field formulation, which is based on the principle curvature, to segment the surface into patches, and then use a primitive-fitting method to classify the basic primitives. However, this method is less competitive for recognizing shape from models, which are strongly corrupted by noise. Décoret et al. [37] extend the standard Hough transform and employ it to identify planes for billboard clouds of triangle meshes. However, this method exhibits poor run-time performance on recognizing basic primitive from large or complex mechanical part because of the high computational demand of the Hough transform. Schnabel et al. [10] present a shape detection method in point cloud based on the RANSAC method. They demonstrate a robust algorithm that used random samples and the middle level information to cluster basic primitive shapes. However, the basic primitives adopted in this method are not suitable for building all the mechanical models. Gelfand et al. [13] propose a slippable motion-based hierarchical clustering method. They introduce a rigid motion to segment mechanical objects into planes, spheres, cylinders, linear extrusion surfaces, surfaces of revolution and helical surfaces. We certify that the exact normal estimation and the choice of neighborhood point set of each point are the key tips for the slippage-based segmentation method theoretically and practically, and employ the RANSAC method to obtain the middle level parameters of each basic primitive, guiding the modification of the normal of each point and the iterative shape recognition of mechanical parts.

Surface type	Recognize method	Geometry property	How to recognize
Plane	Gauss map	Normal distribution	One point
	Slippage	Number of slippage	3
	Curvature	k _{max} , k _{min}	$k_{\text{max}} = k_{\text{min}} = 0$
Sphere	Gauss map	Normal distribution	All the sphere
	Slippage	Number of slippage	3
	Curvature	k _{max} , k _{min}	$k_{\text{max}} = k_{\text{min}} = const$
Cylinder	Gauss map	Normal distribution	A great circle
	Slippage	Number of slippage	2
	Curvature	k _{max} , k _{min}	$k_{\text{max}} = const, k_{\text{min}} = 0$
Cone	Gauss map	Normal distribution	A small circle
	Slippage	Number of slippage	1
	Curvature	k _{max} , k _{min}	$k_{\text{max}} = const, k_{\text{min}} = 0$
Helix	Gauss map Slippage Curvature	Normal distribution Number of slippage k _{max} , k _{min}	Uncertain 1 $k_{ m max} eq 0, k_{ m min} eq 0$
Revolution	Gauss map	Normal distribution	Parallel great circle
	Slippage	Number of slippage	1
	Curvature	k _{max} , k _{min}	$k_{\max} \neq 0, k_{\min} \neq 0$
Extrusion	Gauss map	Normal distribution	A dense circle
	Slippage	Number of slippage	1
	Curvature	k _{max} , k _{min}	$k_{\max} \neq 0, k_{\min} = 0$

 Table 1

 Methods for recognizing the basic primitive.

The remainder of this article is organized as follows. In Section 3, the local slippage analysis-based shape segmentation method is briefly reviewed, and the method for recognizing the basic primitives of mechanical models is described; then the exact normal and the appropriate local region of each point is proved to be the key points for local slippage analysis. Then, Section 4 introduces the iterative shape recognition method in detail, where the middle level information is used to guide the shape recognizion and the normal modification. And the experimental results of recognizing basic primitives from CAD and CAE models are demonstrated in Section 5. Finally, the conclusion and future work are outlined in Section 6.

3. Shape slippage analysis revisited

Most mechanical parts are composed of basic primitives, which can be segmented by applying the local rigid motion type. Gelfand et al. [13] demonstrate slippage signatures to cluster engineering models into basic primitives. We extend the slippage analysisbased method to recognize basic primitives and prove that the exact normal and the appropriate local regions are the key issues for slippage analysis.

3.1. Slippage analysis and shape recognition

3.1.1. Slippage analysis

Let *M* be a set of triangles $\{t_i\}_{i=1}^m$ that constitute a mesh surface. We want to segment it into several middle level surface patches denoted by $S = \{t_i\}_{i=1}^n$, where $\bigcup_{i=1}^k S_i = M$ and $S_i \bigcap S_j = \emptyset$ for any $i \neq j$. And each patch can be described as a basic primitive. We also employ the representative parameters such as sphere center, sphere radius, plane normal, cylinder axis, etc., for the basic primitives.

Gelfand et al. [13] introduce a surface descriptor, which is based on the surface behavior under different kinds of rigid motion. And the rigid motion M(t) can be decomposed into two timevary motions: the rotation part R(t) and the translation part T(t). Applying a differential to the rigid motion equation of each point, the instantaneous velocity of point *x* can be expressed as follows:

(1)

where $r = (r_x, r_y, r_z)$ is a 3×1 vector of rotation around x, y, and z axes and $t = (t_x, t_y, t_z)$ is a translation vector. Then the rigid motion type is defined as follows:

- If r = 0, the motion *M* is a translation with constant velocity along the direction *t*.
- If $r \cdot t = 0$, the motion *M* is a rotation with constant angular velocity.
- If $r \cdot t \neq 0$, the motion *M* is a uniform helical motion.

According to the approximation of the kinematic surface, a slippage signature of each patch is defined to segment the surface mesh. And applying the rigid motion to a small patch, a minimization equation of a set of points is defined to find the slippable motion type.

$$\min_{[r\ t]} \sum_{i=1}^{n} ((r \times p_i + t) \cdot n_i)^2.$$
(2)

Eq. (2) is a least-squares problem, which can be solved as a linear system Cx = 0, where C is a covariance matrix of second partial derivatives of the objective function with respect to the motion parameter. Then the mesh model can be segmented into basic primitives based on the eigen value of matrix C.

3.1.2. Shape recognition

With the slippage motion analysis, the mechanical objects can be segmented into different shape types. The plane and sphere are 3-slippage motion surfaces. The cylinder is the 2-slippage motion surface. The cone, helix, revolution and extrusion are 1-slippage motion surfaces. There may be a non-kinematic surface with 0-slippage motion. However, only with the number of the slippage motion, it is very difficult to recognize the basic primitives. By analyzing the design and manufacture processes of mechanical objects, we classify all the surfaces of mechanical objects into seven types: plane, sphere, cylinder, cone, extrusion, revolution, helix, as shown in Table 1. There are also kinds of freeform surfaces in mechanical objects, which are non-kinematic surfaces. We easily classify those freeform surfaces into non-slippage type (or have not been recognized surface).

In order to recognize all the basic primitives, we introduce two more methods to assist the shape recognition: the Gauss



Fig. 1. Relation of the noise points and the exact points in the local surface (p_i is a point on the ideal surface, and p_{i1} represents the scanned point. *O* defines the original point of the Cartesian coordinate system. t_u , t_v , *n* symbolize the direction of the max curvature, the min curvature and the normal respectively).

map method and the principle curvature. With the k_{max} and k_{min} curvature, we can robustly recognize the plane, sphere, cylinder, cone by the curvature difference. To consolidate the results of surface type segmentation and classification, we also introduce Gauss map method for the normal distribution of the basic primitive is different, when mapping to the unit sphere. The shape recognition method can be seen in Table 1.

3.2. Key issues for slippage analysis

3.2.1. Normal influence on slippage analysis

From Eq. (2), we can find that the point position and point normal are the only two variations of the slippage-based shape recognition. For the effect of noise, it is difficult to get the exact point of the original shape, and it is even more difficult to obtain the exact normal of each point. Lots of researchers have studied the normal estimate method for point cloud in computer graphics. The traditional normal estimation methods are based on plane fitting, quadric surface fitting, triangle-based area weighted average and triangle-based angle weighted average. Grimm et al. [38] propose a point set analysis approach that uses a combination of local surface models and one-rings to produce robust surface normal estimation for non-uniformly sampled, noisy point data. But without the middle level and high level information, it is also hard to get the exact normal of points in a smooth surface, let alone in kinds of sharp corners, sharp edges, which are only C^0 continuous. In order to get the exact point normal, we introduce iterative middle level information-based normal modification method, which will be described in detail in Section 4.2.4. Here we just illustrate the normal effects on slippage shape segmentation.

In the local surface of a point p_i , the noise can be decomposed into a portion along the direction of the normal and the directions of the principle curvatures (as shown in Fig. 1).

$$p_{i1} = p_i + \alpha \cdot t_u + \beta \cdot t_v + \delta \cdot n \tag{3}$$

where p_i is a point on the ideal surface, and p_{i1} represents the scanned point. *O* defines the original point of the Cartesian coordinate system. t_u , t_v , n symbolize the direction of the max curvature, the min curvature and the normal respectively. If the scanned points are dense enough, the component of the noise in the tangent plane can be ignored for it has little influence on the slippage shape segmentation. So, Eq. (3) can be expressed as follows:

Substitute this into Eq. (2), the result is shown as follows.

$$(r \times p_{i1} + t) \cdot n_i = [r \times (p_i + \delta n) + t] \cdot n_i$$

$$= (r \times p_i + r \times \delta n + t) \cdot n_i$$

$$= r \times p_i \cdot n_i + r \times \delta n \cdot n_i + t \cdot n_i$$

$$= r \times p_i \cdot n_i + \delta r \times n \cdot n_i + t \cdot n_i$$

$$= r \times p_i \cdot n_i + t \cdot n_i$$

$$= (r \times p_i + t) \cdot n_i.$$
(5)

Eq. (5) shows that if we can get the exact normal of each point, and the scanned data is dense enough, the noise will have little effect on the shape recognition. Though the scanned data are with much noise and are raw, we can also refine the shape recognizing result by iteratively modifying the approximated normal of each point. Fig. 2 shows the normal influence on the slippage shape recognition of fandisk model. We use only 1-ring neighborhood of the points to approximate the normal and to compute the slippage motion type of each point. And for the sharp corner and sharp edge points, we split them into several points according to the number of the neighbor surface. Each point is added a Gaussian noise (σ is the ratio of the distance between the point and the original point to the max model length) along the normal direction. Fig. 2(b) and (c) show the original normal-based initial slippage shape recognition result and the triangle area weighted normalbased one respectively.

3.2.2. Neighborhood choice influence on slippage analysis

The slippage shape detection method is based on finding a point set that minimizes the motion along the normal for each point. Getting the exact neighbor point set of each point is found to be the other key process for slippage shape recognition. When a small local neighbor is selected for each point, it would be incorrectly recognized as high slippage shape type. If a large scale neighbor point set is chosen, it would be incorrectly recognized as low slippage shape type. Gelfand et al. [13] also find the neighborhood influence on the slippage shape recognition. They choose the nearest 30 vertices to initialize the slippage motion type of each point, and then define a similarity score to merge small patches into large patches based on the number of the slippage motion and the motion eigenvectors. If the chosen point set is not suitable for segmenting the surface correctly, they choose a larger neighborhood and try the segmentation method again.

To get the exact point region for slippage shape recognition, we employ an adaptive region growing method guided by the middle level information, which would be elaborated in Section 4.2.3. Here, we just illustrate the influence of neighbor point choosing in detecting the basic primitive by region growing. Fig. 3(a) shows the shape recognizing result of fandisk model by the proposed method using a small neighbor (1-ring neighbor) for each point. It segments most of the cylinder into 3-slippage shape. With the neighbor size growing, more and more cylinder points are correctly recognized as 2-slippage shape type. However, it cannot recognize all the slippage shape type by reason of the cylinder radius non-uniformity and the cylinder surface incompletion. Keeping on increasing the point's neighbor size, it would recognize most cylinder points, but incorrectly recognizes high slippage shape as low slippage shape, which are obviously shown in Fig. 3(c) and (d).

4. Shape recognition of CAD models

In this section, we describe the implementation details of the presented algorithm. Our algorithm aims to recognize the basic primitives of mechanical parts by extending the slippage motionbased shape detection method. The shape recognition algorithm consists of three main steps: (1) preprocessing and initialization;



(a) Slippage shape recognition result of fandisk model with 1-ring neighborhood of each point to compute the slippage type.



 $\sigma = 0.01$ $\sigma = 0.025$ $\sigma = 0.05$ (b) Slippage shape recognition result of fandisk with the original normal and with Gaussian noise along the normal.





(c) Slippage shape recognition result of fandisk with area weighted estimated normal and with Gaussian noise along the normal.

Fig. 2. Points normal influence of the slippage shape type (red is the 3-slippage motion surface, yellow is the 2-slippage motion surface, green is the one motion surface, blue is the non-kinematic surface). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



Fig. 3. Points neighbor influence of the slippage shape recognition result of fandisk (red is the 3-slippage motion surface, yellow is the 2-slippage motion surface, green is the 1-slippage motion surface, blue is the non-kinematic surface). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



Fig. 4. Flowchart of the iterative slippage shape recognition framework.

(2) iterative shape recognition; and (3) the surface boundary rectification. The flowchart of our algorithm is shown in Fig. 4.

4.1. Preprocessing and initialization

4.1.1. Sharp feature detection

As we have formulated in Section 3, the key processes for slippage motion detecting are to get the exact normal and to find the exact neighbor size of each point. However, it is difficult to find the exact normal of each point in the sharp edge of mechanical objects. We adopt the feature preserved mesh smooth method [39] to smooth the original model, which is strongly corrupted by noise, and then employ a robust tensor voting-based feature detecting method to classify all the points into a corner, a sharp edge, and a plane. More detail about tensor voting-based feature detecting method could refer to the related papers [40,41]. For the sharp edge and corner points are C^0 continuous, we set them to be non-kinematic surface points. Then, we only need to detect the slippage motion type of plane points.

4.1.2. Local slippage analysis

As shown in Fig. 3, the result of the slippage-based shape segmentation method is sensitive to the chosen neighbor size of each point. A small size will make the local point to get more slippage motions, which can be decreased by growing the chosen neighbor to an appropriate size. On the contrary, with a larger neighbor size, the slippage motion would be reduced, which is dilemma to be corrected to the right slippage motion type. So, we choose the 1-ring



Fig. 5. Middle level information of basic primitives (the point and the normal for the plane; the center point and the sphere radius for the sphere; the axis direction, the axis position and the radius for the cylinder; the apex point, the axis direction and the cone-apex angle for the cone; the extrude direction, and the base curve for the extrusion; the rotation axis, the axis position and the base curve for the revolution; the rotation direction and the axis position for the helix).

neighbor of each point and adopt the triangle-based area weighted average-based normal estimation method to initialize the surface type. The original shape recognized point type is also used to avoid choosing the corner edge and sharp edge, which would mess the slippage shape recognition.

4.2. Iterative shape recognition

As we have mentioned in Section 3.2, it is difficult to recognize basic primitives by the local neighbor and inaccurate normal-based slippage analysis method; therefore, we employ the iterative method to recognize basic primitives exactly and robustly. The pipeline of the proposed iterative shape recognition method is: first, the surface recognition method is adopted to recognize the basic primitive for each patch. Then, the knowledgebased region growing method is used to grow each recognized patch step by step. Furthermore, the extended RANSAC method is introduced to obtain the middle level information for each patch, and then the middle level information is used to modify the normal of each point. We iteratively used this method to recognize the basic primitives until the iterative number or the normal tolerant is achieved.

4.2.1. Surface type recognition

We use the methods shown in Table 1 to recognize the shape type of each patch. Compared with slippage shape detection, curvature and the Gaussian image normal distribution are more difficult and less robust when recognizing the surface types, so we use the local slippage motion detecting method as the basic method. The local curvature and the normal distribution difference assist in recognizing the surface types. And the information of the curvature also helps to guide knowledge-based region growing.

4.2.2. Knowledge guided region growing

We adopt the adaptive region growing method to get the exact neighbor size for the shape type recognition. We grow each recognized primitive shape patches by adding in the neighbor points. First, the sharp edge and corner point in the neighbor points are ignored. Then, we merge the growing neighbor points, which have the same slippage type with the original point set. And we eliminate the points whose principle curvature is greater than the surface type threshold. Finally, we only retain the points which fit the surface's middle level information within the threshold (the thresholds of normal and *d* parameter for the plane; the thresholds of radius and normal for the cylinder; the thresholds of radius and normal for the sphere; the thresholds of cone-apex angle and normal for the cone; the threshold of the slippage motion direction for extrusion, revolution and helix).

4.2.3. Middle level information detection of basic primitives

With the slippage motion type, the normal distribution, and point's curvature, we can segment the surface into basic primitives theoretically. Because of the noise of the scanned data, the surface will be segmented into lots of small patches. So we introduce the middle level information detecting method to find the basic parameters of each surface and use this information to guide the normal modification.

We extend the RANSAC-based shape detection method to get the middle level information of the basic primitives [10]. We compute the middle level information of the plane, sphere, cylinder, and cone surface similar to the method proposed by [10], and add the detection method for extrusion, revolution and helix, which is shown in Fig. 5. The details of shape parameters detection method for extrusion, revolution and helix are shown as follows:

• Extrusion: extrusion is a complex surface, which can be defined by the extrude direction *l*, and the base curve *b* (as shown in Fig. 5(e)). And the slippage motion-based mesh segmentation method would detect the translation or rotation direction easily for 1-slippage surfaces. So, the extrusion direction can be extracted by the slippage motion detect matrix, and the eigen vector of the small eigen value defines the slippage motion direction. For the extrusion is a translation slippage surface,



Fig. 6. Middle level information-based normal modification: (a) axonometric drawing of a plane; (b) axonometric drawing of a sphere; (c) axonometric drawing of a cylinder; (d) axonometric drawing of a cone; (e) front view of a plane; (f) front view of a sphere; (g) front view of a cylinder; (h) front view of a cone (the blue arrow is the exact local normal; the black arrow is the estimated normal; the pink arrow is the modified normal for every iteration). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the second three parts of the eigen vector define the extrusion direction. Then the surface is projected onto the plane whose normal is parallel with the translation direction. Finally, the extrusion base is obtained by fitting the curve in the plane.

- Revolution: the rotation axis, the axis position and the base curve *b* fully define a revolution surface (as shown in Fig. 5(f)). We acquire the rotation direction as mentioned above. For it is a rotation slippage surface, the axis direction is the first three parts of the eigen vector. Then we project all the points to the plane whose normal is parallel with the rotation axis. By computing the center of the point in the projection plane, we get the position of the axis. And with a plane through the axis, the intersection points compose the base curve.
- Helix: as for the helix surface, the most important middle level information is the translation and rotation direction. For the translation direction is perpendicular to the rotation direction, we easily compute the rotation direction and the axis position as mentioned above.

4.2.4. Middle level information-based normal modification

Because of the influence of noise, it is difficult to get the exact normal for each point in the scanned point cloud. So, we employ the iterative normal modification method, which is guided by the surface type and the middle level information. Most mechanical parts are composed of plane, sphere, cylinder and cone, which are high slippage motion shapes and are more sensitive to the point normal. A few objects contain extrusion, revolution and helix, which are 1-slippage shape, and can be classified by the kinematic motion type and the motion direction easily. Then, our method iteratively modifies the estimated normal of the basic shape type such as the plane, sphere, cylinder, cone (as shown in Fig. 6). The details of the normal modification are shown as follows:

• Plane: in the middle level information detecting method which is discussed above, we get the local normal of the point set, as it is shown in Fig. 6(a)(e) with a blue arrow. Then we iteratively modify the normal: $n_{new} = n_{old} + \lambda(n_p - n_{old})$, where n_p is

the detected plane normal of the local point set, and λ is the parameter of the normal modification. We only need to set the number of iteration and the modification parameter.

- Sphere: as for the sphere, we detect the sphere center by the local neighbor point set. Then the normal of each point can be expressed as: $n_c = \frac{p_i c}{\|p_i c\|}$. So we adjust the point normal as the plane. Fig. 6(b)(f) show the normal modification. The blue arrow is the exact local normal; the black arrow is the estimated normal; the pink arrow is the modified normal for every iteration. And *c* is the sphere center.
- Cylinder: for each point in the cylinder, the normal is perpendicular to the axis. Together with the middle level information such as the axis direction and position for the local point set, we get the exact normal by projecting the point and the axis position into the plane which pasts the point and parallels to the axis (as shown in Fig. 6(g)). The exact normal is in the same direction as the vector from the center c to this point. Then we adjust the normal as mentioned above.
- Cone: to get the exact normal of each point in the cone, we use the cone-apex position and the cone axis. Projecting the points into the plane defined by the cone axis and the point, and lining the point and the cone-apex, the normal is in the direction perpendicular to the line (as shown in Fig. 6(h)). Then we adjust the normal as mentioned above.

4.3. Surface boundary rectification

Our shape recognition method extracts basic primitives and gives good qualitative results in terms of nature and disposition of the segmented regions. Nevertheless, our method, like most of the existing methods, does not extract perfect boundaries, due to the disturbance of the noise. But we have got the middle level information of all the primitive patches, so we propose a middle level information-based boundary rectification method.

For there may exist fragment segments and non-smoothing boundary in the original segmented patches, we compute the total





Fig. 7. Fandisk surfaces type recognition: (a) shape diameter function-based mesh segmentation; (b) Gelfand's slippage shape segmentation; (c) back view of our method; (d) right view of random walk mesh segmentation of fandisk model with automatic selecting seeds; (e) right view of hierarchical fitting primitives segmentation of fandisk model with 22 clustering; (f) right view of our method; (g) front view of random walk-based mesh segmentation; (h) front view of hierarchical fitting primitives segmentation; (i) front view of our method.

normal deviation with the middle level information of all the patches in the neighbor of each fragment segment, which contains less than a small value triangle. Then we merge the fragment segments into the neighbor patch which has the minima normal deviation. Also, we extract the entire boundary for each patch, and obtain the 1-ring neighbor. Then we set the point, which is the minimization of the total normal deviation of each point in the boundary, to be the refined boundary point. For the noise influence, there maybe also jag boundary in the segment result. We grow the minimization boundary points by the nearest points detecting method.

5. Experimental results

The shape recognition algorithm proposed in this paper has been implemented in Visual C++ 2008 environment. And all examples run on a PC with 2.93 GHz CPU (IntelCoreDuo) and 2G RAM, under Windows XP 32-bit system. Results show that our algorithm not only works well for the scanned mesh model of mechanical parts, but also performs well in recognizing the surface type of the finite element analysis (FEA) result of mechanical objects. We set

Table 2
Computing time of shape recognition

Mesh	Number of vertices	Number of triangles	Time (s)
Fandisk	6 475	12946	18.047
Block	34 172	68 352	211.406
Shaft	22821	47 942	205.531
Roll	191 109	382242	1046.268
Carter	192729	385 482	1125.684
Crack	200 334	400 645	1257.286
Top gearbox	26778	53620	82.906
Bottom gearbox	35 437	70966	107.016
Cutter head	32 0 1 0	64 152	41.063
Bunny	72 027	144046	705.482
Armadillo	172 974	345 944	948.406
Bust	255 358	510712	1429.056

the plane normal deviation threshold $\varepsilon_{pn} = 0.02$, the cylinder normal deviation ε_{cn} , the sphere normal deviation threshold ε_{sn} , and the cone normal deviation threshold $\varepsilon_{con} = 0.05$. The normal modification parameter $\lambda = 0.1$. Figs. 7–14 show the results of basic primitive recognition, and Table 2 shows the computational cost of our method.



Fig. 8. Block surfaces type recognition: (a) random walk mesh segmentation with automatic selecting seeds; (b) *k*-means segmentation method with 16 clustering; (c) hierarchical fitting primitives segmentation method with 16 clustering; (d) our model surface type recognition method.



Fig. 9. Shaft of compressor surfaces type recognition: (a) front view of hierarchical fitting primitives segmentation method with 20 clustering; (b) front view of *k*-means segmentation method; (d) back view of hierarchical fitting primitives method; (e) back view of *k*-means method; (f) back view of our method.



Fig. 10. Primitive shape recognition of complex models. (Left: rolling stage, middle: carter, right: crack).



Fig. 11. Cutter head of shield machine's finite elements analysis result surface type recognition. (Left: the mesh model for the finite elements analysis, middle: the stress analysis result, right: the surface type recognition result of our method).



Fig. 12. Top gearbox of gear reducer's finite elements analysis result surface type recognition. (Left: the mesh model for the finite elements analysis, middle: the stress analysis result, right: the surface type recognition result of our method).



Fig. 13. Bottom gearbox of gear reducer's finite elements analysis result surface type recognition. (Left: the mesh model for the finite elements analysis, middle: the stress analysis result, right: the surface type recognition result of our method).



Fig. 14. Shape recognition of artificial models. (Left: bunny, middle: armadillo, right: bust).

5.1. Shape recognition of CAD models

Fig. 7 shows the surface type recognition result of fandisk by our method. The result is compared with shape diameter functionbased mesh partition method, random walk-based mesh segmentation method, slippage shape segmentation method, and fitting primitives-based hierarchical segmentation method. Here we set the iterations of normal modification to be 10, for there are scanned data meshes, which have noise. And we set the max region growing neighbor size to be 6, for that the sampled points of the fandisk model are not too much. Fig. 7(a)-(c) show the result of shape diameter function-based mesh segmentation method, Gelfand's slippage shape segmentation method, and our surface type recognition method respectively. We successfully recognize the surface type in the sharp edge. We also recognize the plane in the left part of the model, and recognize the plane and cone from the right patches. Fig. 7(d),(g),(e),(h),(f),(i) show the result of random walk-based mesh segmentation, fitting primitives-based hierarchical segmentation method and our method respectively. We set the cluster number of the fitting primitives-based hierarchical segmentation method to be 22, whose segmentation result is mostly similar to ours, and automatically select 22 seeds for random walk-based mesh segmentation. Fig. 7(d)-(f) show that our method can exactly recognize the cylinder in the middle of the model, and also can recognize the plane in the bottom between two cylinder parts. Fig. 7(g)-(i) show our method can recognize small part of cylinder surface type, but the fitting primitives-based hierarchical segmentation method will segment it into two parts. The recognized result in the right part of the model shows that our method can recognize the cone and plane exactly, where the fitting primitives-based hierarchical segmentation method segment the three patches into one patch.

Figs. 8 and 9 give more comparisons of recognition result on block mesh and shaft mesh of compressor by the random walkbased mesh segmentation method, the fitting primitives-based hierarchical segmentation method, the k-means method and our method. Fig. 8 shows that the k-means method just segments model into distance-based nearby patches, and never considers the surface type. The random walk-based segmentation method improperly segments the models into parts without interactive selecting of seeds. The fitting primitives-based hierarchical segmentation method and our method will segment model surface into basic primitives. And our method can even recognize all the basic primitives and will automatically recognize all the surfaces without inputting the number of clustering. Fig. 9 shows a segmentation result of the complex shaft of a compressor. Our approach can recognize all the basic primitive surface types though the nearby surfaces are smooth transition. Our method would recognize the freeform surfaces into non-slippage shape types.

Fig. 10 shows the shape recognition results of complex mechanical models. Our method successfully recognizes the cylinders and planes for roll models. The cones, planes, and cylinders of the carte model are also recognized by our method. Our method is also adopted to recognize the cylinders, planes, and cones for crack models. The results show that our method is appropriate for automatically recognizing basic primitives for complex mechanical parts.

5.2. Shape recognition of CAE models

Figs. 11–13 show more surface type recognition results of FEA mesh model using our method. We set the normal modification iterations to be 5, for there are mesh data with little noise. And we set the max neighbor size of region growing to be 8, for mesh points are sparse in low curvature but dense in high curvature. Figs. 11(a), 12(a), 13(a) show the original mesh of cutter head of shield machine, top and bottom gearbox of gear reducer, respectively. Fig. 11(b) gives the stress analysis result of the cutter head of the shield machine with Abaqus software. Figs. 12(b), 13(b) show the stress analysis results of the top and bottom gearbox of gear reducer with Ansys software. Figs. 11(c), 12(c), 13(c) give the recognition result of our method, which proves that our method is robust and exact in recognizing the basic primitives. The obtained surface type and the middle level information will progress the analysis result model merging and a couple model building with different physical fields.

5.3. Shape recognition of artificial models

Fig. 14 shows the shape recognition results of artificial models using our method. The bunny and armadillo models are inappropriately fitted into parts, for that the artificial models are composed of complex shape parts, not the basic primitives. The shape recognition result of bust model shows that our method recognizes the rotation surface on the bottom part models exactly, and also inappropriate fits the upper parts model into primitives.

6. Conclusion and future work

We present a new shape recognition approach for the CAD model based on slippage analysis. By analyzing the slippage motion-based segmentation method, we prove that the normal estimating and the neighbor point set choosing of each point are the principle steps for slippage motion detection theoretically and practically. For the normal of each point in point cloud is difficult to estimate exactly, we define the sharp points of mechanical parts to be non-slippage points, avoiding the disturbing of mesh segmentation. Then, with the middle level parameter of basic primitives, we iteratively adjust the point normal and obtain the appropriate regions of each point by region growing. Finally, we detect all the basic primitives exactly. The empirical results show that the proposed algorithm is efficient and robust for shape recognition of the CAD models and CAE meshes of mechanical parts.

With the output of the shape recognition method, there maybe provides an available way to convert mesh surfaces to B-rep even CSG models for CAD/CAM/CAE modeling. With the middle level information and recognized basic surface types, it will be convenient to capture the global structures of mechanical parts, such as symmetries, parallelism, perpendicularity, coplanarity, coaxial, etc. However, there are also freeform surfaces in complex mechanical parts; we simply recognize them as non-slippage surfaces. To recognize and classify the freeform surface is also worthy to study in the future.

Acknowledgments

The support of National Natural Science Foundation of China (51075357, 61070071), National Basic Research 973 Program of China (2011CB706503), Science Fund for Creative Research Groups of National Natural Science Foundation of China (51221004) in carrying out this research is gratefully acknowledged.

References

- Chandrasegaran S, Ramani K, Sriram R, Horváth I, Bernard A, Harik R, et al. The evolution, challenges, and future of knowledge representation in product design systems. Comput Aided Des 2013;45(2):204–28.
- [2] Várady T, Martin R, Cox J. Reverse engineering of geometric modelsan introduction. Comput Aided Des 1997;29(4):255–68.
- [3] Iyer N, Jayanti S, Lou K, Kalyanaraman Y, Ramani K. Three-dimensional shape searching: state-of-the-art review and future trends. Comput Aided Des 2005; 37(5):509–30.
- [4] Ke Y, Fan S, Zhu W, Li A, Liu F, Shi X. Feature-based reverse modeling strategies. Comput Aided Des 2006;38(5):485–506.
- [5] Chang K, Chen C. 3D shape engineering and design parameterization. Comput Aided Des Appl 2011;5(8):681–92.
- [6] Béniére R, Subsol G, Gesquiére G, Breton F, Puecha W. A comprehensive process of reverse engineering from 3D meshes to CAD models. Comput Aided Des 2013;45(11):1382–93.
- [7] Marshall D, Lukacs G, Martin R. Robust segmentation of primitives from range data in the presence of geometric degeneracy. IEEE Trans Pattern Anal Mach Intell 2001;23(3):304–14.
- [8] Attene M, Patané G. Hierarchical structure recovery of point-sampled surfaces. Comput Graph Forum 2010;29(6):1905–20.
- [9] Attene M, Falcidieno B, Spagnuolo Michela. Hierarchical mesh segmentation based on fitting primitives. Vis Comput 2006;22(3):181–93.
- [10] Schnabel R, Wahl R, Klein R. Efficient RANSAC for point-cloud shape detection. Comput Graph Forum 2007;26(2):214–26.
- [11] Yan D, Liu Y, Wang W. Quadric surface extraction by variational shape approximation. In: 4th international conference on geometric modeling. 2006. p. 73-86.
- [12] Yan D, Wang W, Liu Y, Yang Z. Variational mesh segmentation via quadric surface fitting. Comput Aided Des 2012;44(11):1072–82.
- [13] Gelfand N, Guibas L. Shape segmentation using local slippage analysis. In: Eurographics symposium on geometric processing. 2004. p. 219–28.
- [14] Shamir A. A survey on mesh segmentation techniques. Comput Graph Forum 2008;27(6):1539–56.
- [15] Chen X, Golovinskiy A, Funkhouser T. A benchmark for 3D mesh segmentation. ACM Trans Graph 2009;28(3):73:1–73:12.
- [16] Vieira M, Shimada K. Surface mesh segmentation and smooth surface extraction through region growing. Comput Aided Geom Design 2005;22(8): 771–92.
- [17] Jagannathan A, Miller E. Three-dimensional surface mesh segmentation using curvedness-based region growing approach. IEEE Trans Pattern Anal Mach Intell 2007;29(12):2195–204.
- [18] Mangan A, Whitaker R. Partitioning 3D surface meshes using watershed segmentation. IEEE Trans Vis Comput Graphics 1999;5(4):308–21.
- [19] Yamauchi H, Lee S, Lee Y, Ohtake Y, Belyaev A, Seidel H. Feature sensitive mesh segmentation with mean shift. In: Shape modeling international. 2005. p. 236–43.
- [20] Shapira L, Shamir A, Cohen-Or D. Consistent mesh partitioning and skeletonisation using the shape diameter function. Vis Comput 2008;24(4):249–59.
- [21] Lai Y, Hu S, Martin R, Rosin P. Fast mesh segmentation using random walks. In: Proceedings of ACM symposium on solid and physical modeling. 2008. p. 183–91.
- [22] Agathos A, Pratikakis I, Perantonis S, Sapidis N, Azariadis P. 3d mesh segmentation methodologies for cad applications. Computer Aided Design and Applications 2007;4(6):827–41.
- [23] Zhang Y, Paik J, Koschan A, Abidi M. A simple and efficient algorithm for part decomposition of 3d triangulated models based on curvature analysis. In: Proceedings of the international conference on image processing. 2002. p. 273–76.
- [24] Zuckerberger E, Tal A, Shlafman S. Polyhedral surface decomposition with applications. Comput Graph 2002;26(5):733–43.
- [25] Lavoué G, Dupont F, Baskurt A. A new cad mesh segmentation method based on curvature tensor analysis. Comput Aided Des 2005;37(10):975–87.
- [26] Lavoué G, Dupont F, Baskurt A. Curvature tensor based triangle mesh segmentation with boundary rectification. In: Computer graphics international. 2004. p. 10–25.
- [27] Kim H, Choi H, Lee K. Feature detection of triangular meshes based on tensor voting theory. Comput Aided Des 2009;41(1):47–58.
- [28] Cohen-Steiner D, Alliez P, Desbrun M. Variational shape approximation. ACM Trans Graph 2004;23(3):905–14.
- [29] Wu J, Kobbelt L. Structure recovery via hybrid variational surface approximation. In: Proceedomgs of eurographics. 2005. p. 277–84.
- [30] Protopsaltis A, Fudos I. A feature-based approach to re-engineering CAD models from cross sections. Computer Aided Design and Application 2010; 7(5):739–57.

- [31] Sellamani S, Muthuganapathy R, Kalyanaraman Y, Murugappan S, Goyal M, Ramani K, et al. PCS: prominent cross-sections for mesh models. Computer Aided Design and Application 2010;7(4):601–20.
- [32] Goyal M, Murugappan S, Piya C, Benjamin W, Fang Y, Liu M, et al. Towards locally and globally shape-aware reverse 3D modeling. Comput Aided Des 2012;44(6):537–53.
- [33] Benjamin W, Polk A, Vishwanathan S, Ramani K, Walk Heat. Robust salient segmentation of non-rigid shapes. Comput Graph Forum 2011;30(7): 2097–106.
 [34] Fang Y, Sun M, Kim M, Ramani K. Heat-mapping: a robust approach toward
- [34] Fang Y, Sun M, Kim M, Ramani K. Heat-mapping: a robust approach toward perceptually consistent mesh segmentation. In: IEEE conference on computer vision and pattern recognition. 2011. p. 2145–52.
- [35] Golovinskiy A, Kim V, Funkhouser T. Shape-based recognition of 3D point clouds in urban environments. In: IEEE International conference on computer vision. 2009. p. 2154–61.
- [36] Lafarge F, Keriven R, Bredif M. Insertion of 3D-primitives in mesh-based representations: towards compact models preserving the details. IEEE Trans Image Process 2010;19(7):1683–94.
- [37] Décoret X, Durand F, Sillíon F, Dorsey J. Billboard clouds for extreme model simplification. In: Proceedings of the ACM siggraph. 2003. p. 689–96.
- [38] Grimm C, Smart W. Shape classification and normal estimation for nonuniformly sampled, noisy point data. Comput Graph 2011;35(4):904–15.
- [39] Jones T, Durand F, Desbrun M. Non-iterative, feature-preserving mesh smoothing. ACM Trans Graph 2003;22(3):943–9.
- [40] Medioni G, Tang Ch, Lee M. Tensor voting: theory and applications. URL: http://citeseerx.ist.psu.edu/view doc/summary?doi=10.1.1.41.9125 (accessed 26.07.12).
- [41] Tong W, Tang Ch. Robust estimation of adaptive tensors of curvature by tensor voting. IEEE Trans Pattern Anal Mach Intell 2005;27(3):1–16.