On Volume Distribution Features Based 3D Model Retrieval

Mingyong Pang[1,2], Wenjun Dai[2], Gangshan Wu[2], and Fuyan Zhang[2]

[1] Dep. of Educational Tech., Nanjing Normal Uninversity, Nanjing 210097, China
[2] Dep. of Computer Sci. & Tech., Nanjing University, Nanjing 210093, China
{panion, czdwj}@graphics.nju.edu.cn, {gswu, fyzhang}@nju.edu.cn

Abstract. In this paper, a 3D mesh retrieval method is proposed based on extracting geometric features of models. The method first finds three principal directions for a model by employing the principal component analysis method, and rotates the model to align it in a reference frame. Then, three sets of planes are used to slice the model along to the directions respectively. Subsequently, three character curves of the model can be obtained and be used as descriptor to key the model in 3D mesh model library. By comparing descriptors of two models, our method can compute similarity of models. Experiences show that our method is rapid, stable and robust to deal with various mesh models with arbitrary geometric and topological complexity.

1 Introduction

Prevailing in the communities of CAD, virtual reality and gaming environment, 3D mesh models become more and more a common feature of nowadays multimedia. Efficiently accessing such rich and complex 3D data turns into an essential issue and motivated numerous and extensive research work within the area of content based indexing and retrieval. A key aspect of the issue is how to extract features from models to describe their geometric shapes or topologies.

At present, there are approximately two types of methods for extracting features: geometric or topological. The former describes shape characters according to the distributions of geometric properties, e.g, area, curvature, normal and volume, of models. In general, a normalizing preprocessing is needed to ensure that the extracted features satisfy invariance properties with respect to geometric transforms such as isometries and isotropic scaling[1]. In [2],[3], histograms of volume distributions of models concerning their bounding spheres were used to describe the features of the models. A systematic comparison and analysis about various histogram methods can be found in [2]. In [4], a set of parallel planes were employed to slice a model, and then profile on each cross section of the model is extracted and resampled uniformly to calculate the feature of the cross section. All the features were collected and gave a total description of the model. Results show that expensive cost are involved in the method. In [5], a visual similarity based method was developed on the idea that if two 3D models are similar, they also look similar from all viewing angles. According to

Z. Pan et al. (Eds.): ICAT 2006, LNCS 4282, pp. 928–937, 2006.

the idea, a camera system defined on regular dodecahedron were introduced. The cameras located at vertices of the dodecahedron first took "photographs", i.e. silhouettes, for a model from 20 different directions, and then translated the silhouettes into feature curves. The set of the curves forms a *descriptor* of the model. On the other hand, the topological methods characterize the features of models by analyzing branchedness, connectivity and skeleton. In [6] and [7], the Morse theory and Reeb graph were used to describe the topological characters of models, respectively. Hilaga et al [8] further present a multiresolution Reeb graph algorithm. The algorithm represented the skeletal and topological structure of a 3D shape at various levels of resolution. The similarity calculation between 3D shapes was processed using a coarse-to-fine strategy while preserving the consistency of the graph structures.

Besides the methods mentioned above, there some other analytic methods extracting and indexing models, for example, Fourier analysis [9] and spherical harmonic [10]. In spherical harmonic method, the features of every model were represented by rotation invariant descriptors instead of rotation dependent descriptors that were aligned into a canonical coordinate system defined by the model. For a thorough summary of shape matching and a comparison of techniques, see [11] and [12].

In this paper, we present a volume distribution features based method that is capable of retrieving similar models from mesh model library for a given model. The method consists of two phases: building and retrieving. The building phase extracts feature curves along three principal axes of each model. The curves form a descriptor of the model and be used to organize database of feature data for the model. In the retrieving phase, descriptor of a given key-model is first extracted and then is used to search similar models in the library by comparing the dissimilarity between key descriptor and the descriptors in the database.

The rest of the paper is organized as follows: Section 2 introduces the concept of solid mesh model and the data structure used for representation of models in memory. Section 3 discusses how to extract volume distribution features from a model and Section 4 shows how to compute the similarity of two models from their descriptors. Section 5 gives some experimental results produced with our method followed by conclusions of our work in Section 6.

2 Solid Mesh Model and Basic Data Structure

In this paper, the model, called *solid mesh model*, prefers to the closed 3D manifold mesh, which satisfies following conditions: a) each face of model is a simple planar convex polygon; b) each edge in the model must be exactly shared by two faces; c) topology around every vertex of the model is homeomorphous to a disc, i.e. manifold property.

In order to conveniently extract cross section of a model sliced by a plane, it is very important to select an appropriate and versatile data structure to represent model in computer. We use halfedge-based data structure [13] to store model

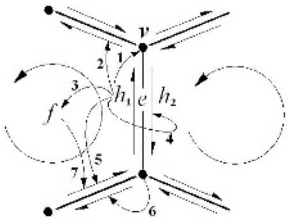


Fig. 1. Halfedge data structure

data in computer memory. One of advantages of the halfedge structure is that it provides fast, constant-time access to the one-ring neighbors of each vertex.

The connectivity information stored in the halfedge structure in this paper is illustrated in Fig.1, where 1 to 5 are pointers pointing to the vertex, next halfedge, face, previous halfedge and opposite halfedge of the halfedge respectively, and 6 and 7 are the pointers pointing the first halfedge of a vertex or a face, respectively.

3 Extracting Volume Distribution Features

3.1 Aligning Model on Principal Directions

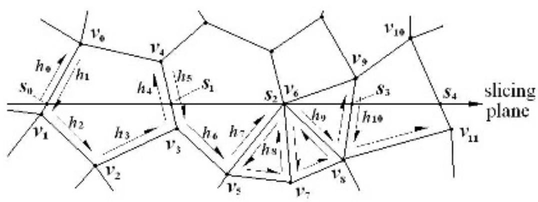
In order to compute the simularity between two models by comparing their volume distribution features, our approach must first seek the three orthogonal principal axes for the model. In the paper, we use the well-known Principal Component Analysis (PCA) method to find the principal axes of the model. PCA is a common method in the area of signal analysis, which translates an initial data set into the space of principal components by performing an orthogonal transformation on coordinates of the vertices of the model and then evaluates the principal axes directions by minimizing the correlativity of a single sample of data, see reference [14] for details .

Once the principal directions are worked out, the model can be rotated such as its principal directions align to the axes of fixed frame system.

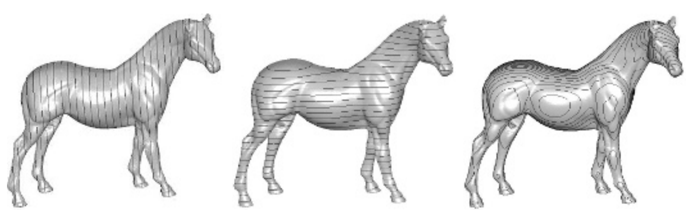
3.2 Slicing Model and Picking Up Section Polygons

Without loss of generality, we only discuss the case that a set of equidistant and horizontal planes, i.e. parallel to xy-plane of the frame system, is used to slice the model. Obviously, the position of each plane can uniquely be described by a pure scalar z_k, the intercept of the plane on z-axis. Here, we name z-axis *slicing axis* and denote the plane or *slicing plane* by $P(z_k)$. According to the difference of topological complexity of the model, the cross section of the sliced model on slicing plane may consist of a series of nested and nonintersect planar simple polygons, called *section polygon group* or SPG, in the paper (Fig.3).

For a set of slicing planes, our approach maintains a list for each plane $P(z_k)$ recording all the edges intersected with the plane, and denotes set of the edges by $E(z_k)$. A single-pass visiting over the edges of the model can finish the construction of the lists. As an example, we give a process of computing SPG from $E(z_k)$. We first get an arbitrary edge, h_0^k, from $E(z_k)$. Following the definition of $E(z_k)$, h_0^k must intersect with $P(z_k)$. According to the relations between the ends, v_i and v_j, of h_0^k and z_k, we can simply work out the intersection, s_0, of $P(z_k)$ and h_0^k. Generally, let us suppose that v_i is on the upper side of $P(z_k)$ and v_j under $P(z_k)$, so the halfedge h_0^k points downwards v_j from v_i. In Fig.2(a), the halfedge h_1 is just an example for h_0^k. Further, let us denote by h_{next}^k the next halfedge of h_0^k. If both ends of h_{next}^k are under $P(z_k)$, we replace h_0^k by h_{next}^k and repeat the process above until the target end of h_{next}^k is above the plane. At the moment, we denote the opposite halfedge of h_{next}^k by h_1^k, and h_1^k is the second halfedge pointing downwards and intersected with the slicing plane. So the second intersection, s_1, can be evaluated. Next, see h_1^k as h_0^k above, and repeat the aforesaid process, and a series of intersections, $s_0, s_1, \ldots$, can be obtained. Once a intersection point is repeated in the series, the process is stopped and a section polygon is formed.



(a) Generating section polygon



(b) Section lines on horse model along three orthogonal directions

Fig. 2. Extracting section polygons from polygonal model

During the process above, once a halfedge intersected with the slicing plane is visited, its corresponding edge is immediately removed from $E(z_k)$. If the $E(z_k)$ is not empty when a section polygon is created, one can repeat the process to find another section polygon until $E(z_k)$ becomes empty. Collecting the created polygons, SPG for the $P(z_k)$ can be obtained on the fly.

A special case needs to be seriously dealt with is that one end or both ends of a halfedge are on the plane, e.g, the halfedges taking v_6 as their one end in

Fig.2(a). A recommended method as following: if the beginning end of halfedge is on $P(z_k)$, treat it as under the plane and oppositely, see the target end of halfedge as above the plane. Fig.2(b) shows an example of horse sliced by three sets of slicing planes from three orthogonal directions.

3.3 Evaluating Areas of SPGs

Since edges belong to each face in the model is arranged counter-clockwise viewed from outer of the model, the polygons of each SPG have directions themselves, which are defined by the order of the series of the intersections (Fig.2(a)). When looking down the SPGs from the positive direction of the reference system, one can find that when one "walks" along the boundary of the polygon, the inner of polygon is always on his left(Fig.3).

For an arbitrary planar polygon $s_0, s_1, \ldots, s_n$ and $s_i = (x_i, y_i)(i = 0, 1, \ldots, n)$, the signed area of the polygon can be calculated by following formula

$$S = \frac{1}{2}\left\{\begin{vmatrix} x_0 & y_0 \\ x_1 & y_1 \end{vmatrix} + \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \cdots + \begin{vmatrix} x_n & y_n \\ x_0 & y_0 \end{vmatrix}\right\} . \quad (1)$$

When direction of polygon is counter-clockwise, the sign of the area of the polygon is positive, otherwise negative. Thus, the area of a SPG equals to the absolute value of algebraic area sum of its section polygons.

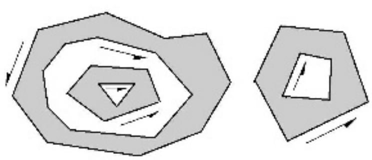


Fig. 3. Section polygon group

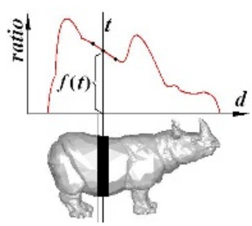


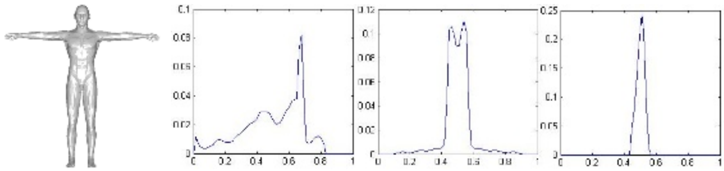
Fig. 4. Defining volume distribution curve

3.4 Building Features Curves of Volume Distributions

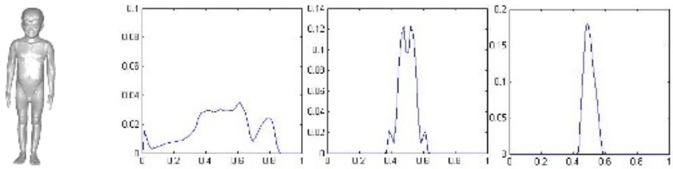
The geometric features of a model can be described by three curves of volume distributions defined on its principal axes, where each curve is defined as $ratio = f(d)$, here, d is discretely defined on the principal axes. In Fig.4, the value $ratio = f(d)$ at $d = t$ is the ratio of volume of shading area in the whole volume of the model. Here, the width of the shading part is the distance between two adjacent slicing planes.

Suppose the number of slicing planes is $N+1$, and the most upper and lower planes are $P(z_{max})$ and $P(z_{min})$ respectively. Obviously, the distance between two adjacent planes is $(z_{max} - z_{min})/N$. So, the slicing planes from bottom to top can be decided as

$$P(z_i) = P\left(z_{min} + i \times \frac{z_{max} - z_{min}}{N}\right), (i = 0, 1, \ldots, N) . \quad (2)$$



(a) Feature curves of man model



(b) Feature curves of child model

Fig. 5. Two character models and corresponding feature curves

According to Section 3.3, suppose the areas of SPGs on the planes are S_0, $S_1, \ldots, S_N$, respectively, the volume of the model can be approximately evaluated by

$$V \approx \frac{z_{max} - z_{min}}{N} \sum_{i=0}^{N-1} \frac{S_i + \sqrt{S_i S_{i+1}} + S_{i+1}}{3} \ . \tag{3}$$

When N is big enough, the areas of two adjacent SPGs is close to identical. In this case, we use $(S_i + S_{i+1})/2$ to replace S_i and S_{i+1} in formula (3). So (3) can be simplified as

$$V \approx \frac{z_{max} - z_{min}}{N} \sum_{i=0}^{N-1} \frac{S_i + S_{i+1}}{2} \ . \tag{4}$$

where, volume of the model between the ith and $(i+1)$th slicing planes is about

$$V_i \approx \frac{z_{max} - z_{min}}{N} \frac{S_i + S_{i+1}}{2} \ . \tag{5}$$

Notice that (4) and (5) avoid the expensive square-root operator in (3). Let $d_i = (z_i + z_{i+1})/2$, then

$$ratio_i = f(d_i) = \frac{V_i}{V} = \frac{S_i + S_{i+1}}{\sum_{t=0}^{N-1}(S_t + S_{t+1})} \ . \tag{6}$$

From the discrete feature points $(d_i, f(d_i))$, we can obtain the volume distribution curves by interpolating or fitting methods. Here, the set $\{f(d_0), f(d_1), \ldots, f(d_{N-1})\}$ is named a *feature* of the model. Fig.5 illustrates an example of volume distribution feature curves of two character models.

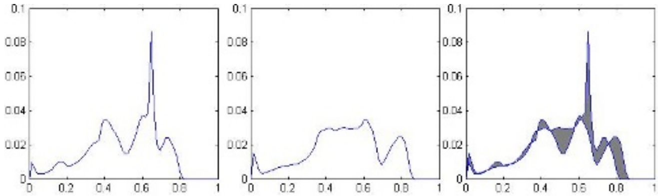


Fig. 6. Evaluating dissimilarity of two feature curves

4 Evaluating Similarity

Suppose that feature sets of two models M^0 and M^1 are $R^0 = \{f^0_X, f^0_Y, f^0_Z\}$ and $R^1 = \{f^1_X, f^1_Y, f^1_Z\}$ respectively, where $f^k_i = \{g^k_{i0}, g^k_{i1}, \ldots, g^k_{iN-1}\}, (k = 0, 1; i = X, Y, Z)$ are features. Here, we use the dissimilarity between M^0 and M^1 to measure the similarity of them. The dissimilarity is specialization of Minkovski distance in the discrete case, which is defined as

$$Diff^{init}(M^0, M^1) = \left(\sum_{i=X,Y,Z} \sum_{j=0}^{N-1} \left(g^0_{ij} - g^1_{ij}\right)^2 \Delta_i \right)^{\frac{1}{2}} . \tag{7}$$

where, $\Delta_i (i = X, Y, Z)$ is the distance between two adjacent slicing planes. To simplify computing, we normalize the model on each direction of slicing axes. Consequently, we have simplified version of the dissimilarity

$$Diff(M^0, M^1) = \sum_{i=X,Y,Z} \sum_{j=0}^{N-1} \left(g^0_{ij} - g^1_{ij} \right)^2 . \tag{8}$$

Since the principal directions of M^0 and M^1 may be exactly reverse when PCA method is used, we have to consider the two possible cases and take the best matching as the similarity of the two models. This is, in (8), let

$$\sum_{j=0}^{N-1} (g^0_{ij} - g^1_{ij})^2 = min\left(\sum_{j=0}^{N-1} \left(g^0_{ij} - g^1_{ij}\right)^2, \sum_{j=0}^{N-1} \left(g^0_{ij} - g^1_{i,N-j-1}\right)^2 \right) . \tag{9}$$

5 Results

We have implemented a prototype retrieval system using C^{++} on 2.8GHz P4 PC with 512M memory. The system consists of two modules. The building module extracts features from model library and organizes them into a feature database. At same time, it also provides user a way of selecting the number of slicing planes used to extract features. The matching module has capability of offering some key models complying with keywords input by user and searches similar models by matching the features of a selected key model.

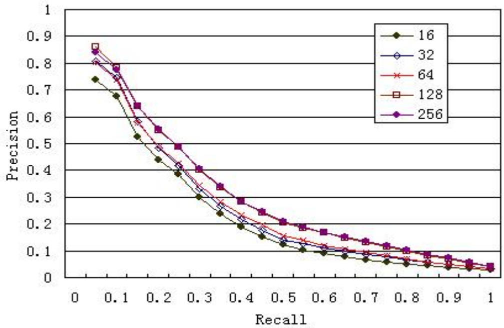


Fig. 7. Evaluation diagram of "precision vs recall" of our method

Table 1. Time costs of extracting and comparing descriptors

Levels of slicing	Extracting(ms)	Comparing(ms)	descriptor size(k)
16	0.143	0.0083	1.05
32	0.156	0.0110	1.94
64	0.192	0.0159	3.72
128	0.258	0.0322	7.28
256	0.393	0.0893	14.4

In our experiences, we use the same mesh model set as used in [16]. The set includes 1834 different models manually categorized into 90 subclasses. Since more the number of slicing planes more accuracy of the retrieval will be obtained, and at same time more the calculating costs will be involved, we have to compromise the aspects. We use 16, 32, 64, 128 and 256 planes to slice the models respectively and the related time costs and sizes of feature data are illustrated in Tab.1 and Fig.7. In the table, the *extracting time* is the average time of extracting 1834 descriptors, including the time of reading models and writing key data. the *comparing time* is average time of matching a key descriptor with the descriptors of 1834 models. Fig.7 shows the relation between precision and recall of our method when different number of slicing planes is used. Traditionally, the diagram of "precision vs recall" is a common way of evaluating performance in text and visual information retrieval. Recall measures the ability of the system to retrieve all models that are relevant and precision measures that the ability of the system to retrieve only models that are relevant. The figure illustrates that our method has near performances when 128 and 256 slicing planes are used. So, 128 planes is the better selection than 256 planes with represent to the tradeoff.

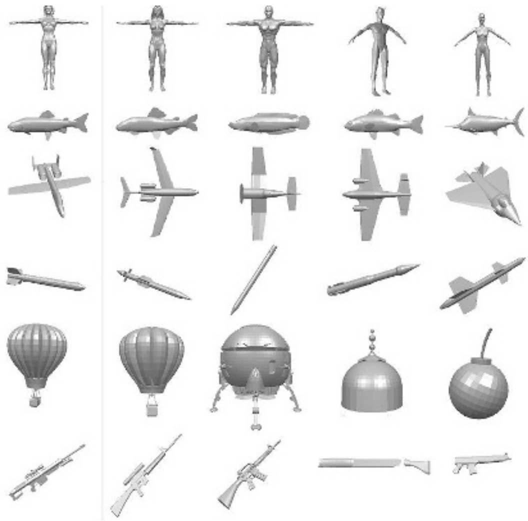


Fig. 8. Retrieval results of our method from an unorganized library consisted of 10991 models, where leftmost column are key models and top 4 matchings are placed on their right

Fig.8 shows some retrieval results of our method from an unorganized model library consisted of 10991 models of size 3.2G. The leftmost column are key models and their top 4 matchings are placed on the right.

6 Conclusion

In the paper, we present a solution of retrieving models from 3D data library based on extracting volume distribution features and implement a prototype system of the solution. Experiential results shows that our method is well of precision-recall diagram and both the aspects of constructing key database and matching similar models are rapid and efficient. Our method can deal with various solid mesh models with arbitrary geometry and topologies. Especially, it can identify the cases that the traditional methods are difficult to handle, e.g, silhouette based method can't distinguish the hollow sphere from solid sphere.

References

1. Zaharia T, Preteux F. Shape-based retrieval of 3D mesh models. IEEE International Conference of Multimedia and Expo'2002 (ICME '02), 1: 437–440.
2. Ankerst M, Kastenmuller G, Kriegel H. 3D shape histograms for similarity search and classification in spatial databases. Proc. of 6th International Symposium on Large Spatial Databases, Hong Kong, China, 1999, 207–226.
3. Osada R, Funkhouser T, Chazelle B, et al. Shape distributions. ACM Transactions on Graphics, 2002, 21(4): 807–832.

4. Pu J, Liu Y, Gu Y, et al. 3D model retrieval based on 2D slice similarity measurements, Proc. of the 2nd International Symposium on 3D Data Processing, Visualization and Transmission, Thessaloniki, Greece, 2004, 95–101.
5. Chen D, Tian X, Shen Y, et al. On visual similarity based 3D model retrieval, Computer Graphics Forum, 2003, 22(3): 223–232.
6. Shinagawa Y, Kunii T. Constructing a Reeb graph automatically from cross section. IEEE Computer Graphics & Applications, 1991, 11(6): 44–51.
7. Xiao Y, Werghi N, Siebert P. A topological approach for segmenting human body shape. 12th International Conference on Image Analysis and Processing, Mantova, Italy, 2003, 82–93.
8. Hilaga M, Shinagawa Y, Kohmura T, et al. Topology matching for fully automaticSimilarity estimation of 3D shapes, ACM SIGGRAPH'2001, 2001, 203–212.
9. Kazhdan M, Funkhouser T, Rusinkiewicz S. Rotation invariant spherical harmonic repressentation of 3D shape descriptors. ACM SIGGRAPH'2003, 2003, 156–164.
10. Vranic D V, Saupe D. Description of 3D-shape using a complex function on the sphere. Proc. Of the IEEE International Conferece on Multimedia and Expo(ICME2002), Lausanne, Switzerland, 2002, 177–180.
11. Iyer N, Jayanti S, Lou K, Kalyanaraman Y, and Ramani K. Three dimensional shape searching: State-of-the-art review and future trends. Computer-Aided Design, April 2005, 37: 509–530.
12. Tangelder J, Veltkamp R. A survey of content based 3D shape retrieval methods. in International Conference on Shape Modeling and Applications 2004, pp145–156
13. Weiler K. Edge-based data structure for solid modeling in curved- surface environments, IEEE Computer Graphics & Application, 1985, 5(1): 21–40.
14. Jolliffe I T. Principal component analysis. New York, Springer, 2002.
15. Puzicha J, Rubner Y, Tomasi C, et al. Empirical evaluation of dissimilarity measures for color and texture. IEEE International Conference on Computer Vision, 1999, 1165–1173.
16. Min P, Kazhdan M, Funkhouser T. A comparison of text and shape matching for retrieval of online 3D models. Proc. European Conference on Digital Libraries, Bath, UK, 2004, 209–220.