Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition 40 (2007) 283–295
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

A new 3D model retrieval approach based on the elevation descriptor

Jau-Ling Shih*, Chang-Hsing Lee, Jian Tang Wang

Department of Computer Science and Information Engineering, Chung Hua University, No. 707, Sec. 2, WuFu Rd., Hsinchu, Taiwan, ROC

Received 18 August 2005; received in revised form 11 February 2006; accepted 11 April 2006

Abstract

The advances in 3D data acquisition techniques, graphics hardware, and 3D data modeling and visualizing techniques have led to the proliferation of 3D models. This has made the searching for specific 3D models a vital issue. Techniques for effective and efficient content-based retrieval of 3D models have therefore become an essential research topic. In this paper, a novel feature, called *elevation descriptor*, is proposed for 3D model retrieval. The elevation descriptor is invariant to translation and scaling of 3D models and it is robust for rotation. First, six elevations are obtained to describe the altitude information of a 3D model from six different views. Each elevation is represented by a gray-level image which is decomposed into several concentric circles. The elevation descriptor is obtained by taking the difference between the altitude sums of two successive concentric circles. An efficient similarity matching method is used to find the best match for an input model. Experimental results show that the proposed method is superior to other descriptors, including spherical harmonics, the MPEG-7 3D shape spectrum descriptor, and D2.

Keywords: 3D Model retrieval; Elevation descriptor

1. Introduction

The development of image, video, and 3D model archives has made multimedia retrieval become a popular research topic. Most of the commercial multimedia retrieval systems employ keyword search to assist users to find desired multimedia data. To facilitate search accuracy, the managers of the multimedia database must empirically annotate well-chosen keywords for all multimedia data. If the database is very large, the task is laborious and time consuming. Moreover, the appropriate keywords differ from person to person. In general, the simplest approach is to extract keywords from filenames, captions, or context (e.g., Google). However, this approach fails when the filenames are not well annotated (e.g., "c0033.jpg") or unspecified filenames are defined (e.g., "jeffrey.gif" or "circle.bmp"). Thus, the demand for an automatic and efficient content-based multimedia retrieval system has become a crucial issue.

With the proliferation of computer graphics and computer animations, 3D models are as plentiful as images and video. The primary challenge to a content-based 3D model retrieval system [1] is to extract proper features for discriminating the diverse shapes of 3D models for efficiently indexing similar ones. The 3D model retrieval methods can be roughly classified into three categories: low-level feature-based methods, high-level structure-based methods, and view-based methods. The low-level feature-based methods try to represent the shape of 3D models by their geometric and topological properties. The features can be a single vector consisting of a fixed number of feature values or distributions of a set of feature values. The high-level structure-based methods try to decompose a 3D model into a set of key parts and capture the geometric relationships of the key parts. The view-based methods project the shape of a 3D model on a number of 2D projections from different views.

The rest of the paper is organized as follows: the related work is described in Section 2. In Section 3, the proposed elevation descriptors (EDs) are introduced. Section 4 gives the experimental results to show the effectiveness of the proposed ED. Finally, conclusions are given in Section 5.

* Corresponding author. Tel.: +886 3 5186407; fax: +886 3 5186416.
E-mail address: sjl@chu.edu.tw (J.-L. Shih).

0031-3203/$30.00
doi:10.1016/j.patcog.2006.04.034

2. Related work

In this section, some related work for 3D model retrieval is described. The 3D model retrieval methods are classified into three categories: low-level feature-based methods, high-level structure-based methods, and view-based methods.

2.1. Low-level feature-based methods

In 3D model retrieval systems, low-level feature descriptors are usually extracted to describe the geometric properties [2,3], spatial properties [4–9], and shape distributions [10–16] of 3D models. The similarity between two 3D models can be measured by comparing their features.

Zhang and Chen [2] proposed methods to efficiently calculate features such as area, volume, moments, and Fourier transform coefficients from mesh representation of 3D models. Paquet et al. [3] employed moments to describe symmetries of 3D objects, cord-based descriptors to represent shape information in fine details, and wavelet transform descriptors to describe the density distribution through a volume.

Vranic et al. [4] performed Fourier transform on the sphere with spherical harmonics (SH) to get the feature vectors. This method requires pose normalization to be rotation invariant. A modified rotation invariant shape descriptor based on the SH without pose normalization has been proposed by Funkhouser et al. [5,6]. First, a 3D model is decomposed into a collection of spherical functions by intersecting the model with concentric spheres of different radii. Each spherical function is decomposed into a set of harmonics of different frequencies. The sum of norms of each frequency component at each radius forms the shape descriptor. The reason for the descriptor being rotation invariant is that rotating a spherical function does not change the energies in each frequency component. Novotni and Klein [7] used 3D Zernike moments for 3D shape retrieval. It is naturally an extension of SH-based descriptors. The 3D Zernike moments is a 2D histogram indexed by radius and frequency. The benefits of the 3D Zernike moments are that they are rotation invariant and less sensitive to geometric and topological artifacts. Yu et al. [8] generated a surface penetration map in which the number of surfaces that the ray emitted from the center of the sphere penetrates is counted. Fourier transform of the map are used for retrieval or comparison purpose. Ankerst et al. [9] proposed shape histograms to characterize the area of intersection of a 3D model with a collection of concentric shells and sectors. Quadratic form distance measure is employed to compute the distance between the histogram bins.

Osada et al. [10] tried to represent each 3D model by the probability distributions of geometric properties computed from a set of randomly selected points located on the surface of the model. These geometric properties, including distance, angle, area, and volume, are employed to describe the shape distribution. Among these distributions, the most effective is D2, which measures the distribution of distances between any two randomly selected points. Ip et al. [11,12] refined the D2 descriptor by classifying the D2 distance into three categories: IN distance if the line segment connecting the two points lies completely inside the model, OUT distance if the line segment lies completely outside the model, and MIXED distance if the line segment passes both inside and outside the model. The dissimilarity measure is a weighted distance of D2, IN, OUT, and MIXED distributions. However, it is difficult to do the classification task, if a 3D model is represented by polygon meshes. Ohbuchi et al. [13,14] combined the absolute angle-distance (AAD) histogram with the D2 descriptor for 3D model retrieval. AAD measures the distribution of angles between the normal vectors of two surfaces on which the two randomly selected points locate. In their experimental results, AAD outperforms D2 at the expense of about 1.5 times computational cost. In typical mesh-based representation of 3D models, many polygonal meshes are required to finely represent the complex components of a 3D model. As a result, an area-weighted defect will occur since the random sampling of surface points is greatly affected by the complex components. Therefore, Shih et al. [15] proposed a new descriptor called grid D2 (GD2) to alleviate this problem. In GD2, a 3D model is first decomposed into a voxel grid. Rather than on random points, the random sampling operation is performed on voxels within which some polygonal surfaces are located. The shape spectrum descriptor (SSD) [16] is adopted in the MPEG-7 standard for 3D model retrieval. SSD represents the histogram of curvatures of all points on the 3D surface. The advantages of SSD are that it can match two 3D models without first aligning the 3D objects, and that it is robust to the tessellation of the 3D polygonal model.

2.2. High-level structure-based methods

The low-level feature-based methods discussed above only take the geometric or topological properties of 3D models into consideration. On the other hand, high-level structure-based methods describe the relationship between model components. Hilaga et al. [17] used multi-resolution Reeb graphs (MRG) to describe the skeleton structure of a 3D model. Mathematically, the Reeb graph is defined as the quotient space of a shape and a quotient function. The Reeb graph used by Hilaga et al. is based on a quotient function defined by an integral geodesic distance. Bespalov et al. [18] applied the Reeb graph for description of solid models. One major advantage of using the Reeb graph to measure the distance between two 3D models is that it is robust to 3D shape deformation. However, computation of the Reeb graph is time consuming and very sensitive to the fine components of 3D models.

2.3. View-based methods

The main idea of view-based methods is to represent a 3D model using a number of binary images. Therefore, a set of

2D features can be used to index similar 3D models. Each binary image is obtained from the boundary contour of the 3D model from different views. Several methods provide a 2D query interface to facilitate view-based retrieval of 3D models [5,19]. Super and Lu [20] exploited 2D silhouette contours for 3D object recognition. Curvature and contour scale space are extracted to represent each silhouette. Chen et al. [21] introduced the lightfield descriptor to represent 3D models. The lightfield descriptor is computed from 10 silhouettes. Each silhouette is represented by a 2D binary image. Zernike moments and Fourier descriptors are used to describe each binary image. Since a 3D model may be rotated or deformed, the number of 2D silhouettes must be large enough to represent a 3D model. On the other hand, the retrieval time increases as the number of silhouettes increases.

In fact, 2D silhouettes represented by binary images do not describe the altitude information of the 3D model from different views well. Therefore, a new descriptor, called the ED, is proposed for 3D model retrieval. Six elevations are obtained to represent a 3D model. Each elevation is represented by a 2D gray-level image which describes the altitude information of a 3D model from different views. In addition, an effective way for extracting features from each gray-level image is employed in order to make them less sensitive to rotations. In the following section, the proposed method is described in detail.

3. The proposed elevation descriptor for 3D model retrieval

In this section, the proposed ED is described. Since the features are extracted from six elevations representing 2D projections from different views, a similarity matching method is used to find the best match for an input model as efficiently as possible.

3.1. Elevation representation

Initially, the tightest bounding box circumscribing the 3D model is constructed (see Fig. 1(a)). The bounding box is then decomposed into a $2L \times 2L \times 2L$ voxel grid (see Fig. 1(b)). A voxel located at (m, n, h) is regarded as an opaque voxel, notated as $Voxel(m, n, h) = 1$, if there is a polygonal surface located within this voxel; otherwise, the voxel is regarded as a transparent voxel, notated as $Voxel(m, n, h) = 0$. Based on the decomposition process, the area-weighted defect is greatly reduced since each opaque voxel is weighted equally irrespective of the number of points located within this voxel. Secondly, the model's mass center is moved to location (L, L, L) and the average distance from all opaque voxels to the mass center is linearly scaled to be $L/2$ so that the ED is invariant to translation and scaling, as shown in Fig. 1(c). In this paper, L is set to be 32, which provides adequate resolution for discriminating objects and the fine-detail noise in complex components of a 3D model can be filtered out.

Next, six elevations are extracted to indicate the altitude information of 2D projections from six different views: front, top, right, rear, bottom, and left. Each elevation is represented by a gray-level image in which the gray values denote the altitude information. Let the front, top, right, rear, bottom, and left elevations be notated successively as $\boldsymbol{E}_k$, $k = 1, 2, \ldots, 6$. The gray value of each pixel on these elevations is defined as

$$f_1(m, n) = \mathbf{max}\{(65 - h)Voxel(m, n, h) | 1 \leqslant h \leqslant 64\}, \quad \text{for } 1 \leqslant m,\ n \leqslant 64,$$

$$f_2(m, h) = \mathbf{max}\{(65 - n)Voxel(m, n, h) | 1 \leqslant n \leqslant 64\}, \quad \text{for } 1 \leqslant m,\ h \leqslant 64,$$

$$f_3(n, h) = \mathbf{max}\{mVoxel(m, n, h) | 1 \leqslant m \leqslant 64\}, \quad \text{for } 1 \leqslant n,\ h \leqslant 64,$$

$$f_4(m, n) = \mathbf{max}\{hVoxel(m, n, h) | 1 \leqslant h \leqslant 64\}, \quad \text{for } 1 \leqslant m,\ n \leqslant 64,$$

$$f_5(m, h) = \mathbf{max}\{nVoxel(m, n, h) | 1 \leqslant n \leqslant 64\}, \quad \text{for } 1 \leqslant m,\ h \leqslant 64,$$

$$f_6(n, h) = \mathbf{max}\{(65 - m)Voxel(m, n, h) | 1 \leqslant m \leqslant 64\}, \quad \text{for } 1 \leqslant n,\ h \leqslant 64.$$

Fig. 2 shows the six elevations of three example 3D models. From these figures, we can see that the two 3D jeep models exhibit similar elevations, whereas the jeep and ship models differ.

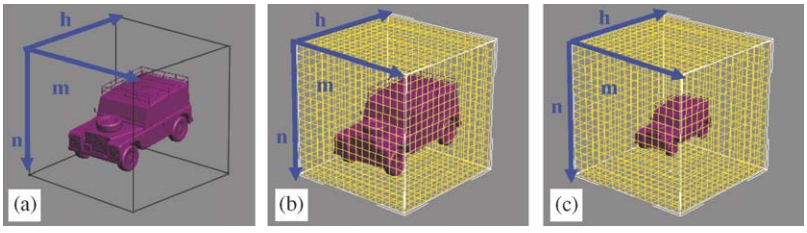

Fig. 1. Original and decomposed 3D jeep model. (a) The 3D jeep model circumscribed by a bounding box. (b) The bounding box of the 3D jeep model is decomposed into a $2L \times 2L \times 2L$ voxel grid. (c) The normalized 3D jeep model.

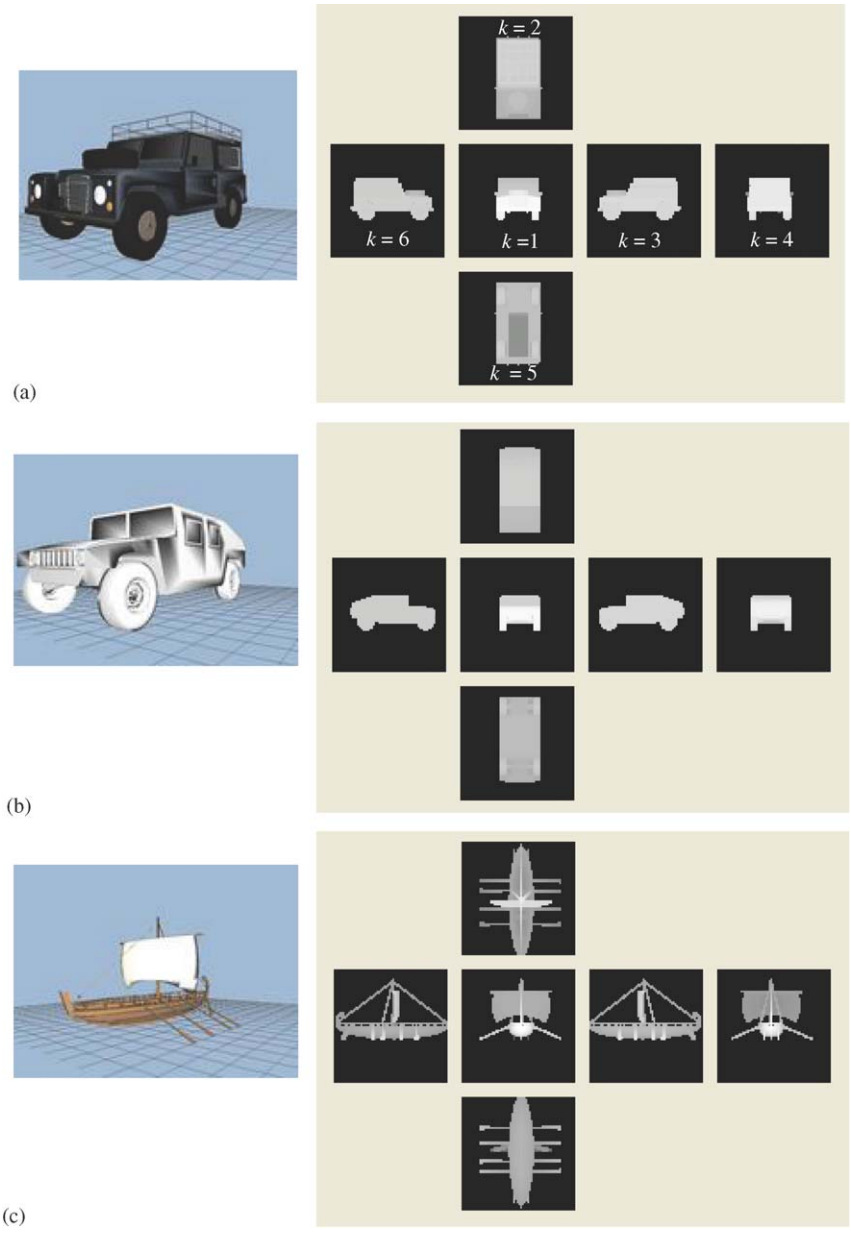

Fig. 2. 3D models and their six elevations including front ($k = 1$), top ($k = 2$), right ($k = 3$), rear ($k = 4$), bottom ($k = 5$), and left ($k = 6$) elevations. (a) 3D jeep model and its six elevations. (b) Another 3D jeep model and its six elevations. (c) 3D ship model and its six elevations.

3.2. Feature extraction

To extract the ED from these six elevations, each elevation is decomposed into L concentric circles around the center point (see Fig. 3). The region within the jth concentric circle is denoted as C_j:

$$C_j = \left\{ (r, c) | \sqrt{(r - L)^2 + (c - L)^2} < j \right\},$$

for $j = 1, 2, \ldots, 32$. For the kth elevation, the sum of gray values of all pixels located within the jth circle, C_j, is defined as

$$g_k(j) = \sum_{(r,c) \in C_j} f_k(r, c),$$

where $j = 1, 2, \ldots, 32$. Let $g_k(0) = 0$, the difference between the sums of gray values within two successive concentric circles is then derived:

$$d_k(j) = g_k(j) - g_k(j - 1),$$

for $j = 1, 2, \ldots, 32$. Furthermore, every $d_k(j)$ value is normalized by using the following equation:

$$x_k(j) = \frac{d_k(j)}{\sum_{k=1}^{6} D(k)},$$

Fig. 3. Top elevation of the 3D jeep model shown in Fig. 2(a) segmented by several concentric circles.

where $D(k)$ is the sum of all $d_k(j)$ values for the kth elevation:

$$D(k) = \sum_{j=1}^{32} d_k(j).$$

The ED $\mathbf{x}$ is defined as

$$\mathbf{x} = [(\mathbf{x}_1)^{\mathrm{T}}, (\mathbf{x}_2)^{\mathrm{T}}, \ldots, (\mathbf{x}_6)^{\mathrm{T}}]^{\mathrm{T}},$$

where

$$\mathbf{x}_k = [x_k(1), x_k(2), \ldots, x_k(32)]^{\mathrm{T}}.$$

Fig. 4 shows the EDs for the three 3D models shown in Fig. 2. It is evident that these two jeep models exhibit similar EDs whereas the jeep and ship models have totally different ones.

In general, the ED is less sensitive to rotation if a 3D model is rotated by a small degree. Assume a 3D model is rotated by a small degree θ (see Fig. 5), the increment/decrement Δn of the altitude value of a voxel located at radius j is

$$\Delta n = j \tan \theta.$$

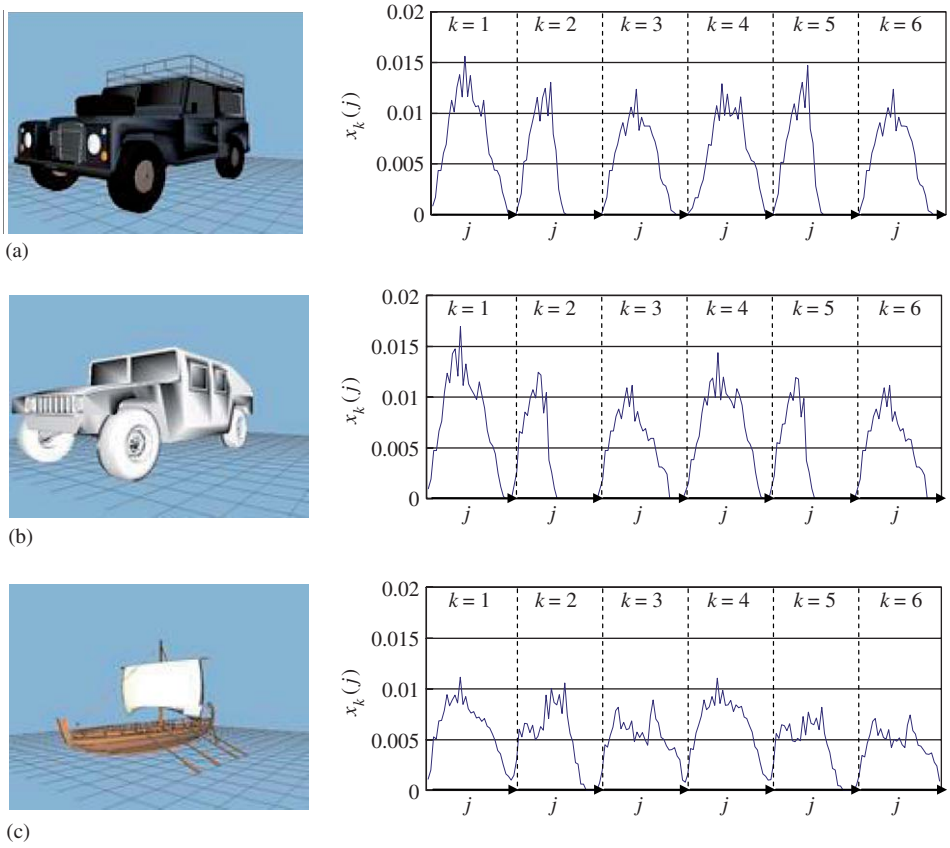

Fig. 4. Elevation descriptors for the three models shown in Fig. 2. The vertical axis represents $x_k(j)$. The horizontal axis represents j $(1 \leqslant j \leqslant 32)$ for each $k = 1, 2, \ldots, 6$ successively. (a) The elevation descriptors for the jeep model shown in Fig. 2(a). (b) The elevation descriptors for another jeep model shown in Fig. 2(b). (c) The elevation descriptors for the ship model shown in Fig. 2(c).

However, the altitude value of a voxel located on the other side will decrease/increase the same value Δn. On average, the sum of the gray values on the jth concentric circle of the rotated elevation is similar to the original one (see Fig. 6).

3.3. Similarity computations

Since each 3D model is represented by six elevations, it requires 720 (6!) elevation matching operations to compute the similarity between two models. To reduce the matching time, an efficient similarity computation is provided to find the best match for a given query model.

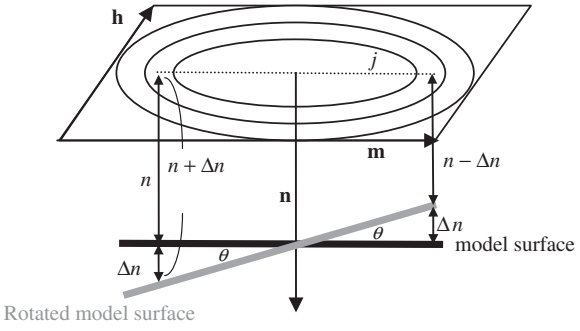

Fig. 5. 3D model rotated by degree θ.

The matching operations can be greatly reduced if the relative positions of the elevations are taken into account. In practice, the front elevation $\boldsymbol{E}_1$ and the rear elevation $\boldsymbol{E}_4$ locate on opposite sides. Similarly, the right elevation $\boldsymbol{E}_3$ and the left elevation $\boldsymbol{E}_6$ as well as the top elevation $\boldsymbol{E}_2$ and the bottom elevation $\boldsymbol{E}_5$ also locate on opposite sides. Let the six elevations of the query model q and the matching model s be, respectively, defined as $\boldsymbol{E}_k^q$ and $\boldsymbol{E}_k^s$, for $k=1, 2, \ldots, 6$. The six elevations of a query model q can be divided into three pairs: $(\boldsymbol{E}_1^q, \boldsymbol{E}_4^q)$, $(\boldsymbol{E}_2^q, \boldsymbol{E}_5^q)$, and $(\boldsymbol{E}_3^q, \boldsymbol{E}_6^q)$. Similarly, the six elevations of a matching model s can be divided into three pairs: $(\boldsymbol{E}_1^s, \boldsymbol{E}_4^s)$, $(\boldsymbol{E}_2^s, \boldsymbol{E}_5^s)$, and $(\boldsymbol{E}_3^s, \boldsymbol{E}_6^s)$. To calculate the difference between q and s, if $\boldsymbol{E}_1^q$ matches $\boldsymbol{E}_i^s$, $\boldsymbol{E}_4^q$ must match $\boldsymbol{E}_{[(i+2) \bmod 6]+1}^s$ according to the topological relationship between $\boldsymbol{E}_1^q$ and $\boldsymbol{E}_4^q$. Similarly, if $\boldsymbol{E}_2^q$ matches $\boldsymbol{E}_i^s$, $\boldsymbol{E}_5^q$ must match $\boldsymbol{E}_{[(i+2) \bmod 6]+1}^s$, and if $\boldsymbol{E}_3^q$ matches $\boldsymbol{E}_i^s$, $\boldsymbol{E}_6^q$ must match $\boldsymbol{E}_{[(i+2) \bmod 6]+1}^s$. In summary, the number of elevation matching operations that need to be performed is $3! \times 2^3 = 48$, instead of 720 matching operations. Table 1 lists these 48 matching operations. In this table, for ith permutation p_i, $\boldsymbol{E}_k^q$ will match $\boldsymbol{E}_{p_i(k)}^s$, $1 \leqslant k \leqslant 6$.

Let $\mathbf{x} = [(\mathbf{x}_1)^{\mathrm{T}}, (\mathbf{x}_2)^{\mathrm{T}}, \ldots, (\mathbf{x}_6)^{\mathrm{T}}]^{\mathrm{T}}$ and $\mathbf{y} = [(\mathbf{y}_1)^{\mathrm{T}}, (\mathbf{y}_2)^{\mathrm{T}}, \ldots, (\mathbf{y}_6)^{\mathrm{T}}]^{\mathrm{T}}$ denote the EDs of q and s, respectively. For the matching operation corresponding to the ith permutation

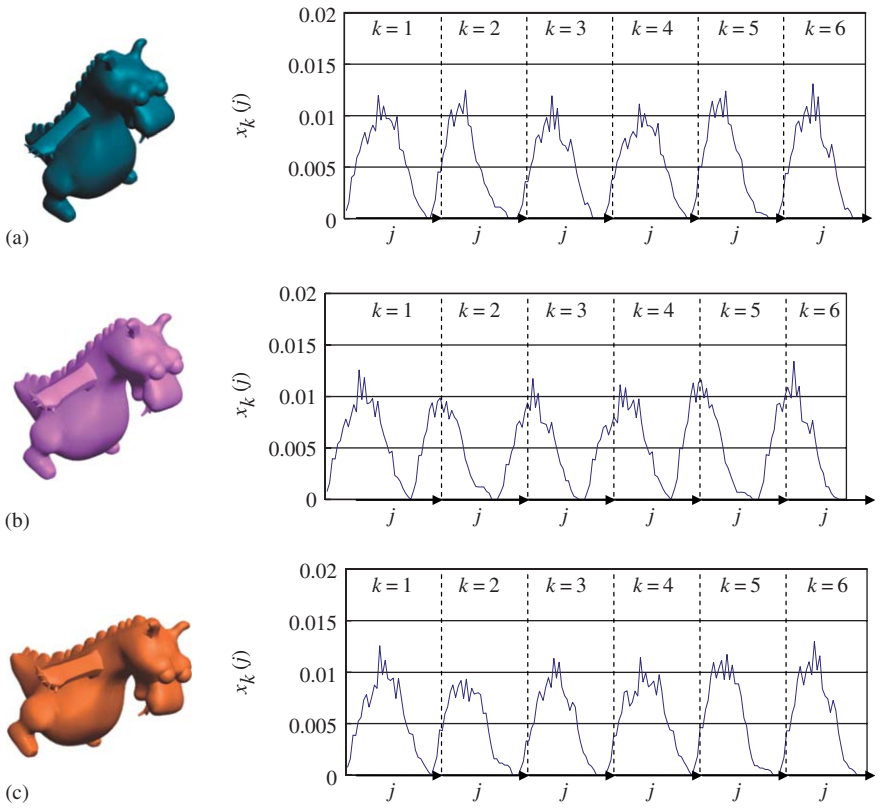

Fig. 6. Elevation descriptors for the rotated models. (a) The elevation descriptors for the dragon model. (b) The elevation descriptors for the rotated dragon model with rotation degrees $\theta_m = 10°$ and $\theta_n = 10°$. (c) The elevation descriptors for another rotated dragon with rotation degrees with $\theta_m = 20°$ and $\theta_n = 20°$.

Table 1
Forty-eight permutations for elevation matching between a query model and a matching model

k	1	2	3	4	5	6	k	1	2	3	4	5	6	k	1	2	3	4	5	6
$p_1(k)$	1	2	3	4	5	6	$p_{17}(k)$	2	1	3	5	4	6	$p_{33}(k)$	3	2	1	6	5	4
$p_2(k)$	1	3	2	4	6	5	$p_{18}(k)$	2	3	1	5	6	4	$p_{34}(k)$	3	1	2	6	4	5
$p_3(k)$	1	5	3	4	2	6	$p_{19}(k)$	2	4	3	5	1	6	$p_{35}(k)$	3	5	1	6	2	4
$p_4(k)$	1	3	5	4	6	2	$p_{20}(k)$	2	3	4	5	6	1	$p_{36}(k)$	3	1	5	6	4	2
$p_5(k)$	1	2	6	4	5	3	$p_{21}(k)$	2	1	6	5	4	3	$p_{37}(k)$	3	2	4	6	5	1
$p_6(k)$	1	6	2	4	3	5	$p_{22}(k)$	2	6	1	5	3	4	$p_{38}(k)$	3	4	2	6	1	5
$p_7(k)$	1	5	6	4	2	3	$p_{23}(k)$	2	4	6	5	1	3	$p_{39}(k)$	3	5	4	6	2	1
$p_8(k)$	1	6	5	4	3	2	$p_{24}(k)$	2	6	4	5	3	1	$p_{40}(k)$	3	4	5	6	1	2
$p_9(k)$	4	2	3	1	5	6	$p_{25}(k)$	5	1	3	2	4	6	$p_{41}(k)$	6	2	1	3	5	4
$p_{10}(k)$	4	3	2	1	6	5	$p_{26}(k)$	5	3	1	2	6	4	$p_{42}(k)$	6	1	2	3	4	5
$p_{11}(k)$	4	5	3	1	2	6	$p_{27}(k)$	5	4	3	2	1	6	$p_{43}(k)$	6	5	1	3	2	4
$p_{12}(k)$	4	3	5	1	6	2	$p_{28}(k)$	5	3	4	2	6	1	$p_{44}(k)$	6	1	5	3	4	2
$p_{13}(k)$	4	2	6	1	5	3	$p_{29}(k)$	5	1	6	2	4	3	$p_{45}(k)$	6	2	4	3	5	1
$p_{14}(k)$	4	6	2	1	3	5	$p_{30}(k)$	5	6	1	2	3	4	$p_{46}(k)$	6	4	2	3	1	5
$p_{15}(k)$	4	5	6	1	2	3	$p_{31}(k)$	5	4	6	2	1	3	$p_{47}(k)$	6	5	4	3	2	1
$p_{16}(k)$	4	6	5	1	3	2	$p_{32}(k)$	5	6	4	2	3	1	$p_{48}(k)$	6	4	5	3	1	2

p_i, $1 \leqslant i \leqslant 48$, the distance between $\mathbf{x}$ and $\mathbf{y}$ is defined as

$$Dis^i_{q,s} = \sum_{k=1}^{6} \|\mathbf{x}_k - \mathbf{y}_{p_i(k)}\|_1 = \sum_{k=1}^{6}\sum_{r=1}^{32} |x_k(r) - y_{p_i(k)}(r)|,$$

where $p_i(k)$ denotes the kth value for the ith permutation, $1 \leqslant k \leqslant 6$. The distance between the query model q and the matching model s is defined as

$$Dis_{q,s} = \min_{1 \leqslant i \leqslant 48} Dis^i_{q,s}.$$

Then, the similarity measure between q and s is defined as the inverse of the distance

$$Sim_{q,s} = \frac{1}{Dis_{q,s}}.$$

Note that the larger the similarity value, the more similar a matching model is to the query. Therefore, the retrieved models similar to a query can be determined by taking those with larger similarity values.

4. Experimental results

To demonstrate the effectiveness of the proposed ED for different 3D models, experiments have been conducted on two test databases. Three other features, including SH [5], the MPEG-7 3D SSD [16], and D2 [10], are implemented to compare the retrieval results. The performance is measured by *recall* and *precision*. The recall value, *Re*, and the precision value, *Pr*, are defined by the following equations:

$$Re = N/T,$$

and

$$Pr = N/K,$$

where N is the number of relevant models retrieved, T is the total number of relevant models in the database, and K is the total number of retrieved models.

4.1. Experiment on Database 1

Database 1 is established to test the performance of invariance to deformations. To derive Database 1, 20 models are selected as the seed models. Then, each seed model is deformed by 14 kinds of transformations, including 4 geometric deformations, 2 scalings, 3 rotations, and 5 various resolutions (see Fig. 7). Thus, there are in total 300 models in Database 1.

Fig. 8 shows some 3D models and their deformed models representing geometric deformation, rotation, scaling, and various resolution as well as the corresponding EDs. The similarities between these pairs of models are 0.9702, 0.9304, 0.9999, and 0.9691, respectively. We can see that even though the model is deformed, the similarity value is still large enough.

In the previous section, it is shown that the ED of a rotated 3D model is similar to the original if the rotation degree is small. In our simulation results, the ED is also robust if a 3D model is rotated by a large degree. Table 2 shows the distance between a rotated query model and all seed models in Database 1. The query model q is a dragon, the seed model of class 3 on Database 1 (see Fig. 7). The query dragon model is rotated about the m-axis and n-axis by different degrees θ_m and θ_n : 20°, 40°, 60°, and 80°. We can see that even though the dragon model is rotated by various degrees, the one with the smallest distance is still the original dragon model. That is, the proposed ED is robust to rotations.

In our experiments, each model in Database 1 is presented as a query. Table 3 shows the average recall values for all query models using the proposed ED, SH, 3D SSD,

Fig. 7. Database 1: (a) 20 seed models, (b) 15 deformed models for the cat class.

and D2. From Table 3, we can see that the ED outperforms other descriptors. The detailed comparison of the average recall value for each class is shown in Fig. 9. The ED has the best performance for most classes. To see what kind of deformations will dramatically affect the retrieval result, a detailed performance comparison for each kind of deformation is shown in Fig. 10. From this figure, we can see that the ED always get the best performance.

4.2. Experiment on Database 2

The second database, Database 2, is derived from the Princeton Shape Benchmark database [22] which contains 1814 models (161 classes) and is used for evaluating shape-based retrieval and analysis algorithms. Note that in this database each class contains a different number of models. The 22 classes that have the largest number of models are selected as queries. Each of them contains at least 15 models. These 22 classes are shown in Fig. 11.

The performance is also measured by *recall* and *precision*. Since the number of models in each class is different, the recall value and the precision value for the *j*th query model within the *i*th class are defined as

$$Re_i^j = N_i^j / T_i,$$

and

$$Pr_i^j = N_i^j / K,$$

where N_i^j is the number of relevant models retrieved, T_i is the total number of relevant models in the database, and K is the total number of retrieved models. The average recall

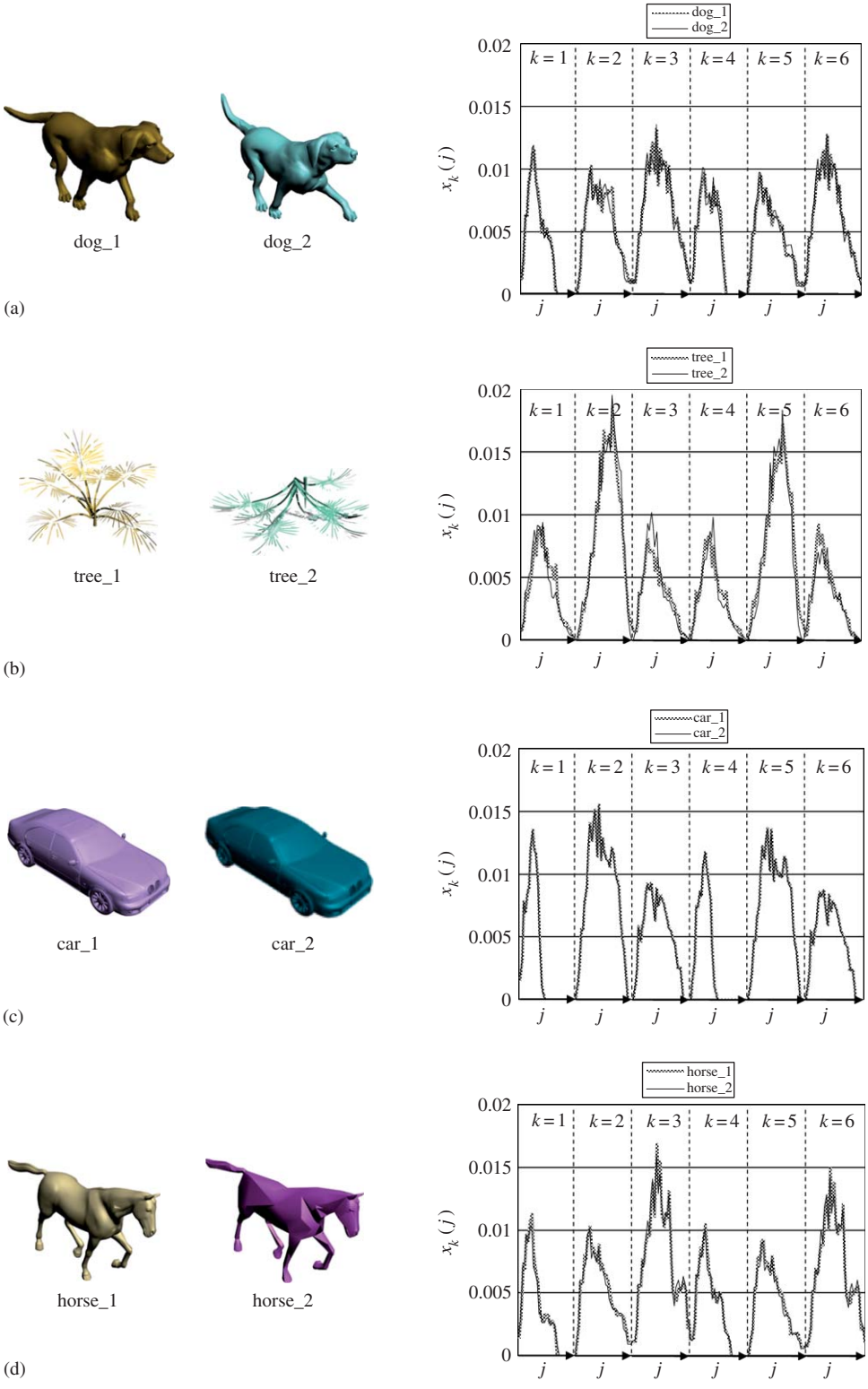

Fig. 8. Deformed 3D models and their corresponding elevation descriptors: (a) geometric deformation, (b) rotation, (c) scaling, (d) various resolution.

and precision are defined by the following equations:

$$Re = \frac{1}{T_S} \sum_{i=1}^{22} \sum_{j=1}^{T_i} Re_i^j,$$

and

$$Pr = \frac{1}{T_S} \sum_{i=1}^{22} \sum_{j=1}^{T_i} Pr_i^j,$$

Table 2
Distance between a rotated query model and all 3D seed models in Database 1

Rotated degree		The class number of matching model s																			
θ_m	θ_n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0	55	32	0	42	36	44	26	41	42	40	44	61	30	31	32	29	28	30	35	33
0	20	57	33	6	43	34	45	25	45	47	44	43	63	33	31	29	30	28	30	33	35
0	40	58	35	9	44	35	48	24	44	48	45	45	66	34	34	29	29	30	29	34	36
0	60	56	35	12	45	39	50	23	41	46	43	49	67	33	38	33	29	33	31	36	35
0	80	55	32	12	45	40	47	23	41	44	41	51	65	31	38	34	28	32	32	37	33
20	0	58	34	8	41	36	46	29	44	47	44	45	64	33	33	32	31	28	31	36	37
20	20	59	34	10	41	36	46	29	46	49	45	45	64	36	33	33	31	27	31	35	38
20	40	60	35	12	43	35	48	27	47	50	46	48	66	37	35	33	30	27	30	35	40
20	60	59	37	13	44	38	51	25	46	50	47	49	68	36	38	33	32	30	31	36	38
20	80	56	34	14	43	40	49	26	43	47	44	52	65	34	39	35	30	32	32	38	34
40	0	63	36	14	40	36	47	32	46	51	47	49	64	36	38	36	34	28	35	37	42
40	20	63	35	14	40	35	46	34	48	53	49	48	63	38	35	35	36	27	35	37	43
40	40	66	37	17	41	34	47	34	51	55	51	49	64	41	37	35	35	26	35	36	46
40	60	65	38	17	42	35	49	31	50	55	52	50	66	40	37	35	37	27	35	36	44
40	80	62	36	17	41	38	48	30	47	52	48	53	66	37	40	37	34	30	36	39	41
60	0	61	34	11	40	35	46	31	45	50	47	47	64	35	36	33	33	27	32	37	40
60	20	66	38	17	39	38	49	36	49	56	54	51	65	40	39	37	38	29	37	39	47
60	40	69	40	21	39	34	49	39	51	56	55	50	64	41	42	36	39	27	37	37	50
60	60	69	41	22	40	33	49	40	53	58	56	51	64	42	42	37	40	27	37	38	51
60	80	64	38	16	39	34	47	34	49	53	50	48	64	38	36	36	36	28	34	37	44
80	0	57	33	5	41	35	45	28	41	44	43	44	63	30	32	31	31	29	31	35	36
80	20	64	39	15	39	38	49	37	45	52	51	49	65	37	36	37	36	30	37	38	44
80	40	69	41	20	39	34	49	41	51	56	54	50	63	41	42	37	40	26	38	37	49
80	60	67	41	22	37	32	46	43	52	57	55	48	60	43	42	38	39	26	34	36	50
80	80	64	39	18	37	32	45	39	49	53	51	45	61	38	37	37	36	26	33	36	46

The rotated query model q is derived from the seed model of class 3 in Database 1 with rotation about the m-axis and n-axis by degrees θ_m and θ_n to be $20°, 40°, 60°$ and $80°$. The values shown in this table are multiplied by 10 000.

Table 3
Comparison of the retrieval results of the proposed ED with other descriptors on Database 1

	Re $(K = 15)$	Re $(K = 30)$
ED	0.9722	0.9869
SH	0.9391	0.9700
SSD	0.8840	0.9358
D2	0.8733	0.9222

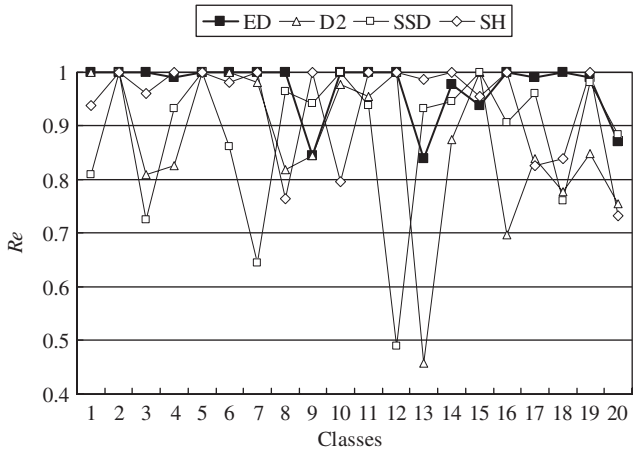

Fig. 9. Comparison of the average recall (Re) for each class on Database 1 ($K = 15$).

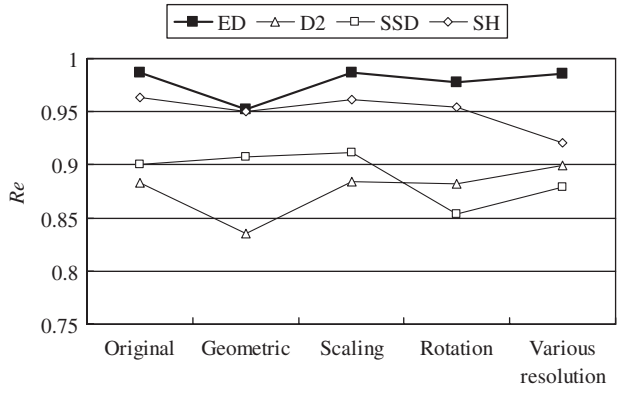

Fig. 10. Recall (Re) for various deformations in Database 1 ($K = 15$).

where $T_S = T_1 + T_2 + \cdots + T_{22}$. The overall performance of the proposed method is still better than others (see Table 4 and Fig. 12). Table 5 compares the average query time. We can see that the query time when using ED is slightly larger than when using SH, but the retrieval accuracy is much better than SH.

5. Conclusions

In this paper, a novel descriptor, called ED, for 3D model retrieval is proposed. First, a 3D model is represented with

Fig. 11. Query models in Database 2 derived from the Princeton Shape Benchmark database. (a) The 22 query classes in Database 2. (b) All models belong to the tank class.

Table 4
Overall performance of the proposed ED method and other methods on Database 2

	$Re\ (K = T_i)$	$Re\ (K = 2T_i)$	$Re\ (K = 3T_i)$	$Re\ (K = 4T_i)$
ED	0.3370	0.4364	0.4945	0.5371
SH	0.2451	0.3159	0.3607	0.3931
SSD	0.2010	0.2395	0.2705	0.2977
D2	0.1745	0.2216	0.2596	0.2902

six gray-level images which describe the altitude information of 2D projections from six different views including front, left, right, rear, top and bottom. Each gray-level image, called an elevation, is then decomposed into a set of concentric circles. The sum of the altitude information within each concentric circle is calculated. The elevation descriptor is obtained from the difference of the altitude sums between two successive concentric circles. Since there are six elevations, an efficient similarity matching method is provided to find the best match for a given query model without exhaustively matching all possible 720(6!) elevation permutations. The experimental results show that for most types of

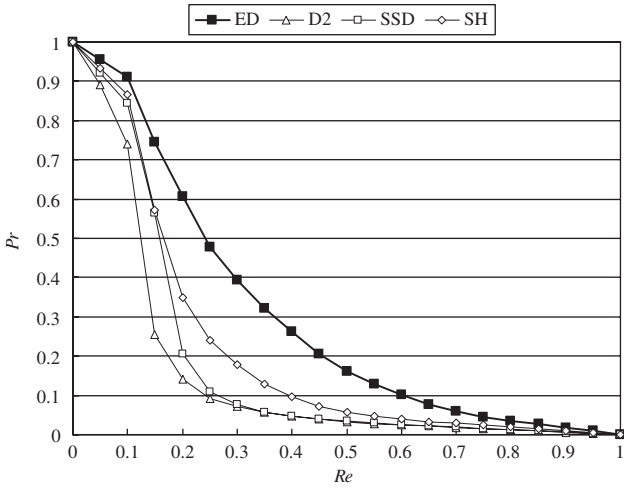

Fig. 12. Precision vs. recall curves on Database 2.

3D models, the proposed ED outperforms other descriptors including SH, the MPEG-7 3D shape spectrum descriptor (SSD), and D2.

Table 5
Comparison of average query time of the proposed ED method and other methods in Database 2

	Query time (s)
ED	3.148
SH	2.946
SSD	1.656
D2	1.661

6. Summary

The development of image, video, and 3D model archives has made multimedia retrieval become a popular research topic. The need for an automatic and user-friendly content-based multimedia retrieval system has consequently become urgent. With the proliferation of computer graphics and computer animations, 3D models are becoming as ubiquitous as images and video. Thus, it is necessary to develop an automatic and efficient retrieval system for 3D models. The primary challenge for a content-based 3D model retrieval system is to extract the proper features for representing the diverse shapes of 3D models for efficient indexing of similar 3D models.

In this paper, we will propose a novel feature, called elevation descriptor, for 3D model retrieval. The elevation descriptor is invariant to translation and scaling of 3D models and it is robust for rotation. First, a 3D model is represented by six gray-level images which describe the altitude information of a 3D model from six different views including front, left, right, rear, top and bottom. Each gray-level image, called an elevation, is decomposed into several concentric circles. The sum of the altitude information within each concentric circle is then calculated. To be less sensitive to rotations, the elevation descriptor is obtained by taking the difference between the altitude sums of two successive concentric circles. Since there are six elevations for each 3D model, an efficient similarity matching method is provided to find the best match for an input model.

To demonstrate the effectiveness of the proposed elevation descriptor for different 3D models, experiments are conducted on two test databases. Three other features, including spherical harmonics, the MPEG-7 3D shape spectrum descriptor (SSD), and D2, are implemented to compare their retrieval results. Database 1 is established to test the performance for invariance to deformations. The elevation descriptor gets the best performance for most kinds of deformations. The second database, Database 2, derived from the Princeton Shape Benchmark database, contains 1814 models (161 classes) which are used for evaluating shape-based retrieval and analysis algorithms. The experimental results show that for most types of 3D models, the proposed elevation descriptor outperforms other descriptors.

Acknowledgments

This research was supported in part by the National Science Council, Taiwan under Contract NSC 94-2213-E-216-018 and Chung Hua University, Taiwan under Contract CHU-94-TR-002.

References

[1] J.W.H. Tangelder, R.C. Veltkamp, A survey of content based 3D shape retrieval methods, Proceedings of the Shape Modeling Applications, 2004, pp. 145–156.
[2] C. Zhang, T. Chen, Efficient feature extraction for 2D/3D objects in mesh representation, Proceedings of IEEE International Conference on Image Processing (ICIP), Thessaloniki, Greece, 2001, pp. 935–938.
[3] E. Paquet, A. Murching, T. Naveen, A. Tabatabai, M. Rioux, Description of shape information for 2-D and 3-D objects, Signal Process.: Image Commun. 16 (2000) 103–122.
[4] D.V. Vranic, D. Saupe, J. Richter, Tools for 3D-object retrieval: Karhunen–Loeve transform and spherical harmonics, Proceedings of the IEEE Workshop on Multimedia Signal Processing, 2001, pp. 293–298.
[5] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, D. Jacobs, A search engine for 3D models, ACM Trans. Graphics 22 (1) (2003) 83–105.
[6] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz, Rotation invariant spherical harmonic representation of 3D shape descriptors, Symposium on Geometry Processing, 2003.
[7] M. Novotni, R. Klein, Shape retrieval using 3D Zernike descriptors, Comput. Aided Des. 36 (2004) 1047–1062.
[8] M. Yu, I. Atmosukarto, W.K. Leow, Z. Huang, R. Xu, 3D model retrieval with morphing-based geometric and topological feature maps, Proceedings of Computer Vision and Pattern Recognition, 2003, pp. 656–661.
[9] M. Ankerst, G. Kastenmuller, H.P. Kriegel, T. Seidl, 3D shape histograms for similarity search and classification in spatial databases, Symposium on Large Spatial Databases, 1999, pp. 207–226.
[10] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Shape distributions, ACM Trans. Graphics 21 (4) (2002) 807–832.
[11] C.Y. Ip, D. Lapadat, L. Sieger, W.C. Regli, Using shape distributions to compare solid models, Proceedings of Solid Modeling, 2002, pp. 273–280.
[12] C.Y. Ip, L. Sieger, W.C. Regli, A. Shokoufandeh, Automated learning of model classifications, Proceedings of Solid Modeling, 2003, pp. 322–327.
[13] R. Ohbuchi, T. Minamitani, T. Takei, Shape-similarity search of 3D models by using enhanced shape functions, Proceedings of Theory and Practice of Computer Graphics, 2003, pp. 97–104.
[14] R. Ohbuchi, T. Takei, Shape-similarity comparison of 3D models using alpha shapes, Proceedings of Pacific Conference on Computer Graphics and Applications, 2003.
[15] J.-L. Shih, C.-H. Lee, J.T. Wang, 3D object retrieval system based on grid D2, Electron. Lett. 41 (4) (2005) 23–24.
[16] MPEG Video Group, MPEG-7 Visual part of eXperimetation Model Version 9.0, Doc. ISO/IEC JTC1/SC29/WG11/N3914, Pisa, January 2001.
[17] M. Hilaga, Y. Shinagawa, T. Kohmura, T.L. Kunii, Topology matching for fully automatic similarity estimation of 3D shapes, Proceedings of SIGGRAPH, 2001, pp. 203–212.
[18] D. Bespalov, A. Shokoufandeh, W.C. Regli, Reeb graph based shape retrieval for CAD, Proceedings of ASME Design Engineering Technical Conferences, 2003.

[19] J. Loffler, Content-based retrieval of 3D models in distributed web databases by visual shape information, Proceedings of Information Visualization, 2000, pp. 82–87.

[20] B.J. Super, H. Lu, Evaluation of a hypothesizer for silhouette-based 3-D object recognition, Pattern Recognition 36 (2003) 69–78.

[21] D.Y. Chen, X.P. Tian, Y.T. Shen, M. Ouhyoung, On visual similarity based 3D model retrieval, Comput. Graphics Forum 22 (3) (2003) 223–232.

[22] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The Princeton shape benchmark, Proceedings of the Shape Modeling Applications, 2004, pp. 167–178.

About the Author—JAU-LING SHIH was born on December 13, 1969 in Tainan, Taiwan. She received the B.S. degree in Electrical Engineering from National Sun Yat-Sen University, Kaohsiung, Taiwan in 1992, the M.S. degree in Electrical Engineering from National Cheng Kung University, Tainan, Taiwan in 1994, and the Ph.D. degree in Computer and Information Science from National Chiao Tung University, Hsinchu, Taiwan in 2002. She is currently an Associate Professor in the Department of Computer Science and Information Engineering, Chung Hua University, Hsinchu, Taiwan. Her main research interests include image processing and image retrieval.

About the Author—CHANG-HSING LEE was born on July 24, 1968 in Tainan, Taiwan. He received the B.S. and Ph.D. degrees in Computer and Information Science from National Chiao Tung University, Hsinchu, Taiwan in 1991 and 1995, respectively. He is currently an Assistant Professor in the Department of Computer Science and Information Engineering, Chung Hua University, Hsinchu, Taiwan. His main research interests include multimedia data compression, digital signal processing, and digital watermarking.