On Visual Similarity Based 3D Model Retrieval

Ding-Yun Chen, Xiao-Pei Tian, Yu-Te Shen and Ming Ouhyoung

Department of Computer Science and Information Engineering,
National Taiwan University, Taipei, Taiwan
{dynamic, babylon, edwards}@cmlab.csie.ntu.edu.tw, ming@csie.ntu.edu.tw

Abstract

A large number of 3D models are created and available on the Web, since more and more 3D modelling and digitizing tools are developed for ever increasing applications. The techniques for content-based 3D model retrieval then become necessary. In this paper, a visual similarity-based 3D model retrieval system is proposed. This approach measures the similarity among 3D models by visual similarity, and the main idea is that if two 3D models are similar, they also look similar from all viewing angles. Therefore, one hundred orthogonal projections of an object, excluding symmetry, are encoded both by Zernike moments and Fourier descriptors as features for later retrieval. The visual similarity-based approach is robust against similarity transformation, noise, model degeneracy etc., and provides 42%, 94% and 25% better performance (precision-recall evaluation diagram) than three other competing approaches: (1)the spherical harmonics approach developed by Funkhouser et al., (2)the MPEG-7 Shape 3D descriptors, and (3)the MPEG-7 Multiple View Descriptor. The proposed system is on the Web for practical trial use (http://3d.csie.ntu.edu.tw), and the database contains more than 10,000 publicly available 3D models collected from WWW pages. Furthermore, a user friendly interface is provided to retrieve 3D models by drawing 2D shapes. The retrieval is fast enough on a server with Pentium IV 2.4GHz CPU, and it takes about 2 seconds and 0.1 seconds for querying directly by a 3D model and by hand drawn 2D shapes, respectively.

Categories and Subject Descriptors (according to ACM CCS): H.3.1 [Information Storage and Retrieval]: Indexing Methods

1. Introduction

Recently, the development of 3D modeling and digitizing technologies has made the model generating process much easier. Also, through the Internet, users can download a large number of free 3D models from all over the world. This leads to the necessities of a 3D model retrieval system. Although text-based search engines are ubiquitous today, multimedia data, such as 3D models, usually lacks meaningful description for automatic matching. The MPEG group aims to create an MPEG-7 international standard, also known as "Multimedia Content Description Interface", for the description of multimedia data [11]. However, little description is about 3D models. The need of developing efficient techniques for content-based 3D model retrieval is increasing.

To search 3D models that are visually similar to a queried model is the most intuitive way. However, most methods concentrate on the similarity of geometric distributions rather than directly searching for visually similar models. The geometric-based approach is feasible since much appearance for an object is controlled by its geometry. In this paper, however, we present a novel approach that matches 3D models using their visual similarities, which are measured with image differences in light fields. We take this approach to better fit the goal of comparing models that appear to be similar to a human observer. The concept of the visual similarity-based approach is similar to that of the image-driven simplification, proposed by Lindstrom and Turk [14].

The geometry-based approach is broadly classified into two categories: shape-based and topology-based matching. The shape-based approach uses the distribution of vertices or polygons to judge the similarity between 3D models [1, 2, 4, 5, 6, 7]. The challenge of the shape-based approach is how to define shape descriptors, which need to be sensitive, unique, stable, efficient, and robust against similarity transformations of various kinds of 3D models. The topology-based approach utilizes topological structures of 3D models to measure the similarity between them [3]. The difficulties of the topology-based approach include automatic topology extraction from all types of 3D models, and the discrimination between topologies from different categories. Each of the two approaches has its inherent merits and demerits. For example, a topology-based approach leads to high similarity between two identical 3D models with different gestures, whereas a shape-based approach cannot. On the other hand, a shape-based approach results in high similarity between 3D models with different connections among parts, whereas a topology-based approach cannot. For instance, both a finger and the shoulder of a human model are parts of a human body. The topologies are quite different whether the finger does or does not connect to a human body, but the shapes are similar.

Most previous works of 3D model retrieval focused on defining suitable features for the matching process [1~13], and were based on either statistical properties, such as global shape histograms, or the skeletal structures of 3D models. Osada et al. [2] proposed and analyzed a method for computing shape signatures of arbitrary 3D polygonal models. The key idea is to represent the signature of a 3D model as a shape distribution, which is a histogram created from the distance between two random points on a surface for measuring global geometric properties. The approach is simple, fast and robust, and could be applied as a pre-classifier in a complete 3D model retrieval system.

Funkhouser et al. [1] proposed a practical web-based search engine that supports queries based on 3D sketches, 2D sketches, 3D models, and/or text keywords. For 3D shape queries, a new matching algorithm that uses spherical harmonics to compute similarities is developed. It does not require repair of model degeneracy or alignment of orientations. In their system, a multimodal query is applied to increase the retrieval performance by combining features such as text and 3D shapes. It is also fast enough to retrieve from a repository of 20,000 models in less than one second.

Hilaga et al. [3] proposed a technique in which the similarity between polyhedral models is accurately and automatically calculated by comparing the skeletal and topological structure. The skeletal and topological structure decomposes a 3D model to a one-dimensional graph structure. The graph is invariant to similarity transformations, robust against simplification and deformation caused by changing posture of an articulated object, etc. In their experimental results, the average search time from 230 3D models is about 12 seconds with a Pentium II 400MHz processor. Another 3D model retrieval system [10] , having 445 models in the database, is extended from the work of Hilaga et al., and takes about 12 seconds on a server with Pentium IV 2.4 GHz processor.

In this paper, a novel visual similarity-based approach for 3D model retrieval is proposed, and the system is also available on the web for practical trial use. The proposed approach is robust against similarity transformations, noise and model degeneracy, etc. There are more than 10,000 3D models in our database, and a user-friendly interface is provided for 3D model retrieval by drawing 2D shapes, which are taken as one or more projection views.

In general, a retrieval system contains off-line feature extraction and on-line retrieval processes. We introduce the *LightField Descriptor* to represent 3D models, which is detailed in Section 2, as well as the feature extraction. In Section 3, comparing 3D models is represented for the on-line retrieval process. The experimental results and the performance evaluations are shown in Section 4. Section 5 concludes the write up.

2. Feature Extraction for Representing 3D Models

The proposed descriptor used for comparing the similarity among 3D models is extracted from 4D light fields, which are representations of a 3D object. The phrase light field describes the radiometric properties of light in a space and was coined by Gershun [23]. A light field (or plenoptic function) is traditionally used in image-based rendering and is defined as a five dimensional function that represents the radiance at a given 3D point in a given direction [24, 25]. For a 3D model, the representation is the same along a ray, so the dimension of the light field around an object can be reduced to 4D [25, 14]. Each 4D light field of a 3D model is represented by a collection of 2D images, which are rendered from a 2D array of cameras. The camera positions of one light field can be put either on a flat surface [25] or on a sphere [14] in the 3D world. The light field representation has not only been used in image-based rendering, but also in image-driven simplification by Lindstrom and Turk [14], whose approach uses images to decide which portions of a model to simplify.

In this paper, we extract features from the light fields rendered from cameras on a sphere. The main idea of using the approach to get the similarity between two models is introduced in Section 2.1. To reduce size of the features and speed up the matching process, the cameras of the light fields are distributed uniformly and positioned on vertices of a regular dodecahedron. We name the descriptor *LightField Descriptor*, and describe it in Section 2.2. In Section 2.3, we show that one 3D model is represented by a set of *LightField Descriptors* in order to improve the robustness against rotations while comparing between two models. One that is also important is the image metric used, and is detailed in Section 2.4. Finally, the flow chart of extracting the *LightField Descriptors* for a 3D model is summarized in Section 2.5.

2.1. Measuring similarity between two 3D models

The main idea comes from the following statement, "If two 3D models are similar, they also look similar from all viewing angles." Accordingly, the similarity between two 3D models can be measured by summing up the similarity from all corresponding images of a light field. However, what

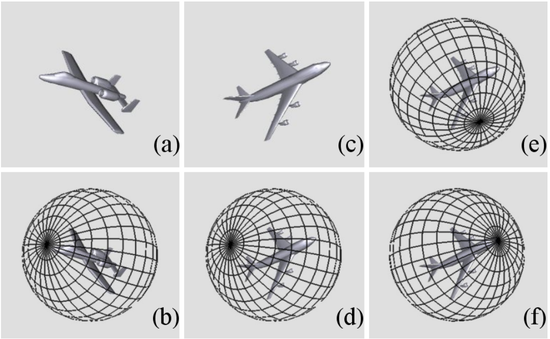

Figure 1: *The main idea measuring similarity between two 3D models*

must be considered is the transformation, including translation, rotation and scaling. The translation and the scaling problems are discussed in Section 2.5 and ignored by our image metric described in Section 2.4. As for rotation, the key to this problem is visual similarity. The camera system surrounding each model is rotated until the highest overall similarity (cross-correlation) between the two models from all viewing angles is reached. Take Figure 1 as an example, where (a) and (c) are two different airplanes with inconsistent rotations. First, for the airplane in Figure 1 (a), we place the cameras of a light field on a sphere, as shown in Figure 1 (b), where cameras are put on the intersection points of the sphere. Then, cameras of this light field can be applied, at the same positions, to the airplane in Figure 1 (c), as shown in Figure 1 (d). By summing up the similarities of all pairs of corresponding images in Figure 1 (b) and (d), the overall similarity between the two 3D models is obtained. Next, the camera system in Figure 1 (d) can be rotated to a different orientation, such as Figure 1 (e), which leads to another similarity value between the two models. After evaluating similarity values, the correct corresponding orientation, in which the two models look most similar from all corresponding viewing angles, can be found, such as Figure 1 (f). The similarity between the two models is defined as summing up the similarity from all corresponding images between Figure 1 (b) and (f).

However, the computation will be very complicated and impractical to a 3D model retrieval system using current processors. Therefore, the camera positions of a light field are distributed uniformly on vertices of a regular dodecahedron, such that reduced camera positions are used for approximation.

2.2. A *LightField Descriptor* for 3D models

To reduce the retrieval time and the size of features, the light field cameras can be put on 20 vertices of a regular dodecahedron. That is, there are 20 different views, which are distributed uniformly, over a 3D model. The 20 views can roughly represent the shape of a 3D model, as been applied similarly in previous works. Huber and Hubert [27] proposed an automatic registration algorithm, which is able to reconstruct real world objects from 15 to 20 various viewpoints from a laser scanner. Lindstrom and Turk [14] also employ the 20 views in comparing 3D models for image-driven simplification.

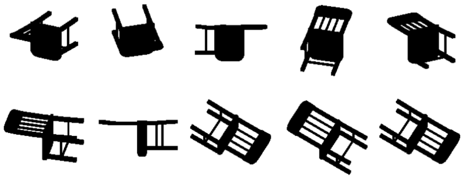

Figure 2: *A typical example of the 10 silhouettes for a 3D model*

Since their applications are different from the retrieval system, the requirements of rendering image and the image metric used are also different. First, lighting is different while rendering images of an object. We turn all lights off, so that the rendered images will be silhouettes only, which enhance the efficiency and the robustness of image metric. Second, orthogonal projection is applied in order to speed up the retrieval process and reduce the size of features. Therefore, ten different silhouettes are produced for a 3D model, since the silhouettes projected from two opposite vertices on the dodecahedron are identical. Figure 2 shows a typical example of the 10 silhouettes of a 3D model. In our implementation, the rendered image size is 256 by 256 pixels. Consequently, the rendering process can filter out high-frequency noise of 3D models, and also make our approach reliable from degeneracy of meshes, such as those missing, wrongly-oriented, intersecting, disjoint and overlapping polygons.

Since the cameras are placed on the vertices of a fixed regular dodecahedron, we need to rotate the camera system 60 times (to be explained below), so that the cameras can be switched onto different vertices, while measuring the similarity between descriptors of two 3D models. The dissimilarity, D_A, between two 3D models is defined as:

$$D_A = \min_i \sum_{k=1}^{10} d(I_{1k}, I_{2k}) , \qquad i = 1..60 \qquad (1)$$

where d denotes the dissimilarity between two images, defined in Section 3.1, and i denotes different rotations between camera positions of two 3D models. For a regular dodecahedron, each of the 20 vertices is connected by 3 edges, which results in 60 different rotations for one camera system (mirror mapping is not available). I_{1k} and I_{2k} are corresponding images under i-th rotation.

Here is a typical example to explain our approach. There are two 3D models, a *pig* and a *cow*, in Figure 3 (a), both

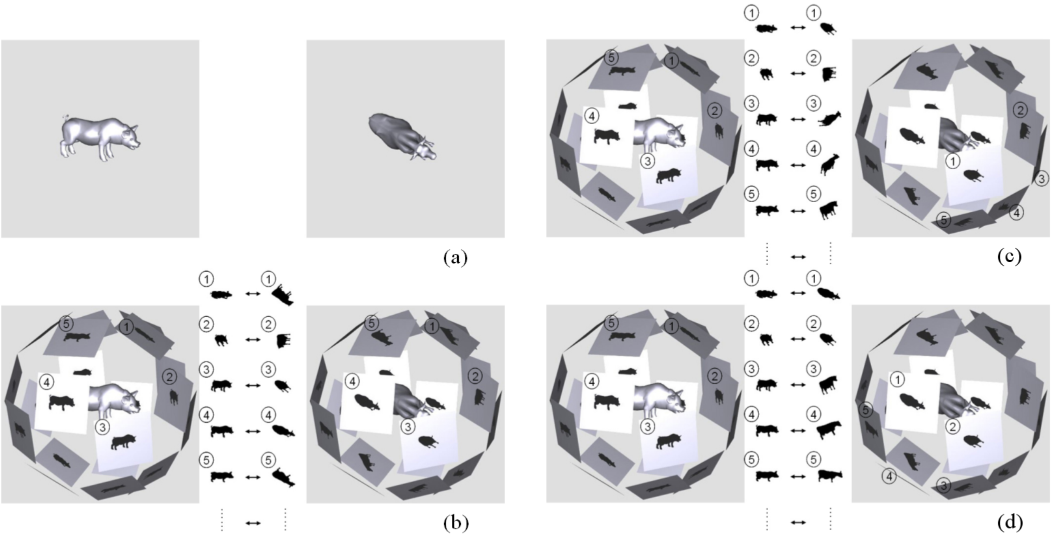

Figure 3: *Comparing LightField Descriptors between two 3D models*

rotated randomly. First, 20 images are rendered from vertices of a dodecahedron for both the 3D model. As shown in Figure 3 (b), we compare all the corresponding 2D images from the same viewing angles, such as, the order 1∼5 between *pig* and *cow* model. Thus we get a similarity value under this rotation of camera system. Then, we map the order 1∼5 differently as in Figure 3 (d), and get another similarity value. After repeating this process, we find a rotation of camera positions with the best similarity (cross-correlation being highest), as shown in Figure 3 (d). Therefore, the similarity between the two models is the summation of the similarities among all the corresponding images.

Consequently, the *LightField Descriptor* is defined as the basis representation of a 3D model, and is defined as features of 10 images rendered from vertices of dodecahedron over a hemisphere. A *LightField Descriptor* somehow eliminates the rotation problem, but this is not exact enough. Therefore, a set of light fields is applied to improve the robustness.

2.3. A set of *LightField Descriptors* for a 3D model

To be robust against rotations among 3D models, a set of *LightField Descriptors* is applied to each 3D model. If there are N *LightField Descriptors*, which are created from different camera system orientations for both 3D models, there are $(N \times (N-1)+1) \times 60$ different rotations between the two models. Therefore, the dissimilarity, D_B, between two 3D models is then defined as:

$$D_B = \min D_A(L_j, L_k), \qquad j, k = 1..N \tag{2}$$

where D_A is defined in Equation (1), and L_j and L_k are light field descriptors of two models, respectively.

The relationship of the N light fields needs to be carefully set to ensure that all the cameras are distributed uniformly and able to cover different viewing angles to solve the rotation problem effectively. The approach of generating the evenly distributing camera positions of the N light fields comes from the idea in relaxation of random points proposed by Turk [15]. The process can be pre-processed, and then all 3D models use the same distributed light fields to generate corresponding descriptors. In our implementation, we set $N = 10$, as shown in Figure 4, that is, the similarity between two 3D models is obtained from the best one of 5,460 different rotations. Therefore, the average maximum error of rotation angle between two 3D models is about 3.4 degree in longitude and latitude. That is:

$$\frac{180^\circ}{x} \times \frac{360^\circ}{x} = 5460 \;\Rightarrow\; x \cong 3.4^\circ \tag{3}$$

which is small enough for our 3D model retrieval system according to our experimental results. Of course, the number

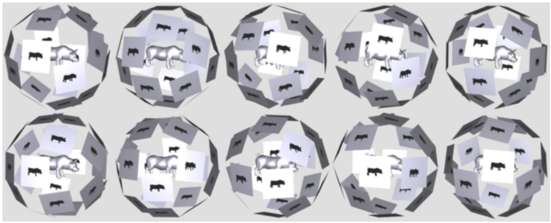

Figure 4: *A set of LightField Descriptors for a 3D model*

N can be bigger than 10, and we will evaluate in the future the saturation effect when *N* becomes bigger.

2.4. Image metric

An image metric is a function measuring the distance between two images. Recently, Content-Based Image Retrieval (CBIR) has become a popular research, and different image metrics have been proposed [19~22]. Many approaches of image metrics are robust against transformations such as translation, rotation, scaling, and image distortion.

One of the important features of images is the shape descriptor, which can be broadly classified into region-based and contour-based descriptor. The use of a combination of different shape descriptors has been proposed recently in order to improve the retrieval performance [21, 22]. In this paper, we adopt an integrated approach proposed by Zhang and Lu [21], which combines a region shape descriptor (Zernike moments descriptor) and a contour shape descriptor (Fourier descriptor). The Zernike moment descriptor is also used in MPEG-7, which is named *RegionShape* descriptor [11]. Different shape signatures have been used to derive Fourier descriptor, and the retrieval using Fourier Descriptors derived from the centroid distance has significantly higher performance than those of the other methods. These were also compared by Zhang and Lu [20]. The centroid distance function is expressed by the distance to boundary points from the centroid of the shape. The boundary points of a shape are extracted through a contour tracing algorithm, proposed by Pavlidis [31]. Figure 5 shows a typical example of the centroid distance. Figure 5 (a) shows a 2D shape rendered from a viewpoint of a 3D model, and the contour tracing result is shown in Figure 5 (b). Figure 5 (c) shows the centroid distance of (a).

Sometimes, however, a 3D model might be rendered into several separated 2D shapes, as shown in Figure 6 (a). When this situation occurs, the following two stages are applied to connect. First, we apply Erosion operation [32] from one to several times to connect the separated parts, as shown in Figure 6 (b). Second, a thinning algorithm [32] is applied in order to connect the separated parts, as shown in Figure 6 (c). Note that the pixels of rendered 2D shape cannot be removed during the thinning algorithm. The separated parts are then connected, and the high-frequency noise will be filtered out by the Fourier descriptor. But if there are still separated parts after running the Erosion operation for several times, a bounding box algorithm will replace the first stage above.

There are 35 coefficients for Zernike moment descriptor and 10 coefficients for Fourier descriptor. Each coefficient is quantized to 8 bits in order to reduce the size of descriptors and accelerate the retrieval process. Consequently, the approach is robust against translation, rotation, scaling and image distortion, and is very efficient.

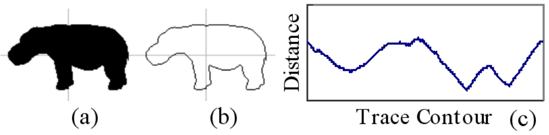

Figure 5: *A typical example of the centroid distance for a 2D shape*

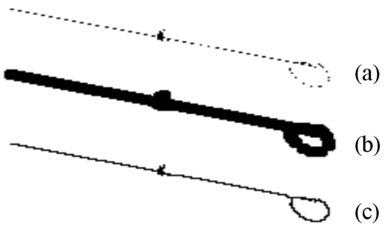

Figure 6: *Connection of different parts of 2D shapes*

2.5. Steps of extracting *LightField Descriptors* for a 3D model

The steps of extracting the *LightField Descriptors* for a 3D model are summarized in the following.

(1) Translation and scaling are applied first to ensure that 3D model is entirely contained in rendered images. The input 3D model is translated from the center of the model to the origin of world coordinate system. The axis is then scaled such that the maximum length is 1.

(2) Render images from the camera positions of light fields, as described in Section 2.3.

(3) For a *LightField Descriptor*, 10 images are represented for 20 viewpoints, and are in a pre-defined order for storage. For a 3D model, 10 descriptors are created, so that totally 100 images should be rendered.

(4) Descriptors for a 3D model are extracted from the 100 images, as in Section 2.4.

3. Retrieval of 3D Models Using *LightField Descriptors*

In the off-line process mentioned in last section, the *LightField Descriptors* of each 3D model in the database are calculated and stored for 3D model retrieval. This section details the on-line retrieving process, which compares the descriptors of the queried one with all the other 3D models in the database. Comparing the *LightField Descriptors* within two models is described in Section 3.1. Those who are greatly dissimilar to the queried model will be rejected early in the process, detailed in Section 3.2, which accelerates the retrieval with a large database. Practically, when a user wants to retrieve 3D models, he/she can upload a 3D model as a query key. However, early experiences of Funkhouser et al. [1] suggest that even a simple gesture interface, such as Teddy system [26], is still too hard for novice and casual users to learn quickly. They proposed that drawing 2D shapes with a painting program to retrieve 3D models is intuitive for users. In this paper, a user-friendly drawing interface for 3D model retrieval is also provided. The approach of comparing 2D shapes with 3D models is described in Section 3.3.

3.1. Similarity between *LightField Descriptors* of two 3D models

The retrieving process can be referred as calculating the similarity one by one between the queried one and each of the models in the database and showing those similar to the queried one. The similarity between two models is defined as summing up the similarity from all the corresponding images, as described in Section 2.3. The comparison of two descriptors is as Equation (2). When comparing the dissimilarity, d, of corresponding images, we use simple $L1$ distance to measure:

$$d\,(J\,,K)\;=\;\sum_{i}|C_{1i}-C_{2i}| \qquad (4)$$

where C_1 and C_2 denote coefficients of two images, and i denotes the index of their coefficients. There are 45 coefficients for each image, each quantized to 8 bits. To simplify the computation, a table is created and stored for the value of $L1$ distance from 0 to 255. Thus, a table-look-up method is used to speed up the retrieval process.

3.2. Retrieval of 3D models from database with large number of models

For a 3D model retrieval system, a database with a large number of models should be considered. For example, there are over 10,000 3D models in our database. To efficiently retrieve 3D models from an enormous database, an iterative early-jump-out method is applied in our 3D model retrieval system. First, when comparing the queried model with all the others, only parts of the images and coefficients are used. This can remove almost half of the models. The threshold of removing models is set as the mean of the similarity rather than the median, since the calculation of the mean is simpler. Then, compare the queried model to the remainder models using more images and coefficients. Repeat the above steps in several times. All iterations are detailed as follows.

(1) In the initial stage, all 3D models in the database are compared with the queried one. Two *LightField Descriptors* of the queried model are compared with ten of those in the database. Three images of each light field are compared, and each image is compared using 8 coefficients of Zernike moment. Each coefficient is quantized to 4 bits.

(2) In the second stage, five *LightField Descriptors* of the queried model are compared to ten of the others in the database. Five images of each light field are compared, and each image is compared using 16 coefficients of Zernike moment. Each coefficient is quantized to 4 bits.

(3) Thirdly, seven *LightField Descriptors* of queried model are compared with ten of the others in the database. The other is the same as the second stage, while another five images of each light field are compared.

(4) The fourth stage is the same as full comparison, but only the Zernike moment coefficients, quantized to 4 bits, are used. In addition, the top 16 of the 5,460 rotations are recorded between the queried one and others.

(5) Each coefficient of Zernike moment is quantized to 8 bits, and the retrieval uses the top 16 rotations, recorded from the 4^{th} stage, rather than 5,460 rotations.

(6) In the last stage, each coefficient of Fourier descriptor is added to the retrieval.

The approach speeds up the retrieval process by early rejection of non-relevant models. The query performance and robustness of each step will be evaluated in the future.

3.3. A user friendly interface to retrieve 3D models from 2D shapes

Creating a queried model for retrieval is not easy and fast for general users. Thus, a user-friendly interface, a painting program, is provided in our system. Furthermore, users can again utilize the retrieved model to find more specific 3D models, since 2D shapes carry less information than a 3D model. To sum up, our 3D model retrieval system is easy for a novice user.

Recognizing 3D objects from single 2D shape is an interesting and difficult problem, and has been researched long time ago. Dudani et al. [16] identified aircrafts with moment invariants derived from the boundary of 2D shapes. They captured 2D images of 3D objects from every 5 degrees of azimuth and roll angle. 3,000 images for 6 aircrafts are used for comparison with an input 2D shape. A 3D aircraft recognition algorithm of Wallace and Wintz [17] used 143 projections to represent an aircraft over the hemisphere, and performed the recognition using normalized Fourier descriptors of the 2D shape boundary. Cyr and Kimia [18] proposed 3D object recognition by generating "equivalent view" from different positions on the equator for 65 3D objects. They recognized an unknown 2D shape by comparing all views of 3D objects using shock matching, which takes about 5 minutes. Recently, Funkhouser et al. [1] proposed a search engine for 3D models, which also provides a 2D drawing interface for 3D model retrieval. The boundary contours are rendered from 13 viewpoints for each 3D model, and then additional 13 shape descriptors are created. In our 3D model retrieval system, it is intuitive and direct to compare 2D shapes with 3D models, since the descriptors for 3D models are composed of features of 100 2D shapes over the hemisphere, as

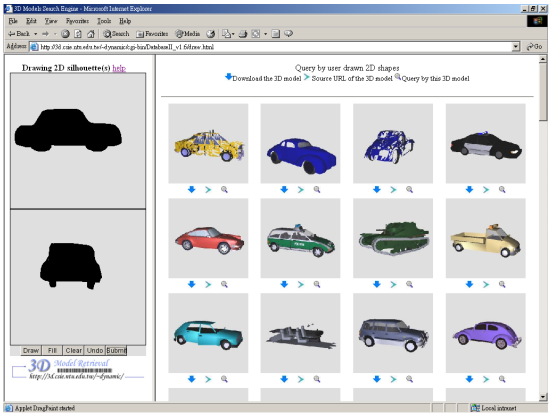
3D Models Search Engine - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://3d.csie.ntu.edu.tw/~dynamic/cgi-bin/DatabaseII_v1.6/draw.html
Go
Drawing 2D silhouette(s) help
Query by user drawn 2D shapes
Download the 3D model
Source URL of the 3D model
Query by this 3D model
Draw Fill Clear Undo Submit
3D Model Retrieval
http://3d.csie.ntu.edu.tw/~dynamic/
Applet DragPaint started
Local intranet

Figure 7: *Retrieval results from user drawn 2D shapes*

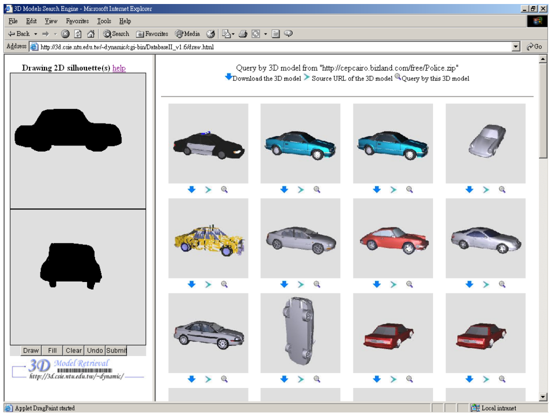
3D Models Search Engine - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://3d.csie.ntu.edu.tw/~dynamic/cgi-bin/DatabaseII_v1.6/draw.html
Go
Drawing 2D silhouette(s) help
Query by 3D model from "http://cepcairo.bizland.com/free/Police.zip"
Download the 3D model
Source URL of the 3D model
Query by this 3D model
Draw Fill Clear Undo Submit
3D Model Retrieval
http://3d.csie.ntu.edu.tw/~dynamic/
Applet DragPaint started
Local intranet

Figure 8: *Retrieval results from interactively searching of selecting a 3D model from Figure 7*

described in Section 2.3. The image metric we used is defined in Section 2.4.

4. Experimental Results

In Section 4.1, the proposed 3D model retrieval system is demonstrated. The performance and robustness of the approach are evaluated in Section 4.2 and 4.3, respectively.

4.1. The proposed 3D model retrieval system

The 3D model retrieval system is on the following web site for practical trial use: *http://3d.csie.ntu.edu.tw*. There are 10,911 3D models in our database now, all free downloaded via the Internet. Users can query with a 3D model or drawing 2D shapes, and then search interactively and iteratively for more specific 3D models using the first retrieved results. Models are available for downloading from the hyperlink of their original downloaded path listed in the retrieval results. Figure 7 shows a typical example of a query with 2D drawing shapes, and Figure 8 shows the interactive search by selecting a 3D model from Figure 7.

The system consists of off-line feature extraction in preprocessing and on-line retrieval processes. In the off-line process, the features are extracted in a PC with a Pentium III 800MHz CPU and GeForce2 MX video card. On the average, each 3D model with 7,540 polygons takes 6.1 seconds to extract features, detailed in Table 1. Furthermore, the average time of rendering and feature extraction for a 2D shape takes 0.06 seconds. Extracting features are suitable for both 3D model and 2D shape matching. No extra effort should be done for 2D shapes. In the on-line process, the retrieval is done in a PC with two Pentium IV 2.4GHz CPUs. Only one CPU is used for the query at one time, and the retrieval takes 2 and 0.1 seconds with a 3D model and two 2D shapes as the query keys, respectively.

4.2. Performance Evaluation

Traditionally, the diagram of "Precision" vs "Recall" is a common way of evaluating performance in documental and visual information retrieval. Recall measures the ability of the system to retrieve all models that are relevant. Precision measures that the ability of the system to retrieve only models that are relevant. They are defined as:

$$Recall = \frac{relevant\ \ correctly\ \ retrieved}{all\ \ relevant}$$

$$Precise = \frac{relevant\ \ correctly\ \ retrieved}{all\ \ retrieved}$$

In general, the recall and precision diagram requires a ground truth database to assess the relevance of models with a set of significant queries. Test sets are usually large, but only a small fraction of the relevant models are included [30]. Therefore, a test database with 1,833 3D models is used for evaluation. The test database contains free 3D models from 3DCafe [34], downloaded in Dec. 2001, but removes several models with failed formats in decoding. One student independent of this research, regarded as a human evaluator, classified the models according to functional similarities. The test database was clustered into 47 classes including 549 3D

	Average	Standard Deviation	Minimum	Maximum
Vertex	4941.6	13582.8	4	262882
Polygon	7540.7	21003.8	2	519813
Time	6.11 sec	4.38 sec	2.32 sec	48.93 sec

Table 1: *Vertex and polygon number of the 10,911 3D models and the feature extraction time from a PC with a Pentium III 800 MHz CPU*

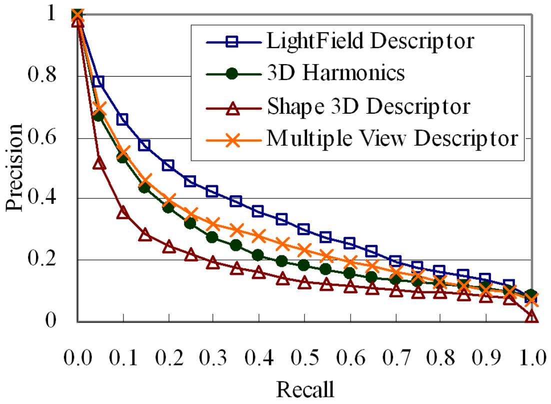

Figure 9: *Performance evaluation of our approach,* LightField Descriptor*, and those of others.*

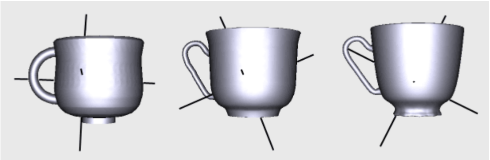

Figure 10: *Three similar cups with their principal axes, orienting the models in different directions. Retrieval results of querying are done by the model in (a). The first number in bracket shows the queried number by our method, and the second number shows the Multiple View Descriptor.*

models mainly for vehicle and household items (such as categories of airplane, car, chair, table, etc.), and all the other 1,284 models are classified as "miscellaneous".

To compare the performance with others systems, three major previous works are implemented as follows:

(1) 3D Harmonics: This approach is proposed by Funkhouser et al. [1], and outperforms many other approaches, such as Moments [12], Extended Gaussian Images [8], Shape Histograms [9] and D2 Shape Distributions [2], which are evaluated in their paper. The source code of SpharmonicKit 2.5 [35], also used in their implementation, is used for computing the spherical harmonics.

(2) Shape 3D Descriptor: The approach is used in MPEG-7 international standard [11], and represents a 3D model with curvature histograms.

(3) Multiple View Descriptor: This method aligns 3D objects with Principal Component Analysis (PCA) [33], and then compares images from the primary, secondary and tertiary viewing directions of principal axes. Descriptor of the viewing directions is also recorded in MPEG-7 international standard [11], but does not limit the usage of image metrics. To get better performance, integration with different image metrics described in Section 2.4 are used. Furthermore, for calculating PCA correctly from vertices, each 3D model is resampled first to ensure that vertices are distributed evenly on the surface.

Figure 9 shows the comparison of the retrieval performance of our approach, *LightField Descriptors*, with those of the others. Each curve plots the graph of "recall and precision" averaged over all 549 classified models in the test database. Obviously, *LightField Descriptor* performs better than the others. The precision values are 42%, 94% and 25% higher than those of 3D Harmonics, Shape 3D Descriptor and Multiple View Descriptor, respectively, after comparing and averaging over all the "recall" axis.

However, in our implementation of 3D Harmonics, the precision is not as good as that indicated in the original paper [1], shown in Table 2. Evaluating by different test database is one possible reason, and another one may lie in a small amount of different details between our implementation and original paper, even if we try to implement the same as the original paper. The test database used in the original paper is also purchased by us, and will be evaluated in the future. As for PCA applied to Multiple View Descriptor, Funkhouser et al. [1] found that principal axes are not good while aligning orientations of different models within the same class, and also demonstrated this problem using 3 mugs. Retrieval with similar examples in our test database is shown in Figure 10. Clearly, our approach works well against this particular problem of PCA.

4.3. Robustness evaluation

All the classified 3D models in the test database are applied to the following evaluation in order to assess the robustness. Each transformed 3D model is then used for queries from the test database. The average recall and precision of all 549 classified models are used for the evaluation. The robustness is evaluated by the following transformation:

(1) Similarity transformation: For each 3D model, seven random numbers are applied to x-, y-, and z-axis rotations (from 0 to 360 degree), x-, y- and z-axis translations (-10∼+10 times of the length of the model's bounding box), and scaling (a factor of -10∼+10).

(2) Noise: Each vertex of 3D model is applied three ran-

Recall	0.2	0.3	0.4	0.5	0.6	0.7
Our approach	0.51	0.42	0.36	0.30	0.25	0.20
3D Harmonics	0.37	0.27	0.22	0.18	0.16	0.14
3D Harmonics with different test database [1]	0.41	0.33	0.26	0.20	0.17	0.14

Table 2: *Precision of 3D Harmonics in the original paper for comparison.*

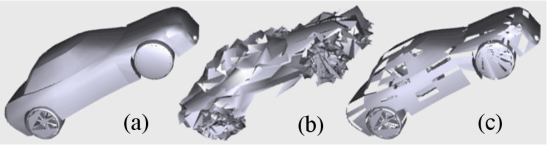

Figure 11: *Robustness evaluation of (b) noise and (c) decimation from (a) original 3D model*

dom number to x-, y- and z-axis translation (-3%∼+3% times of the length of the model's bounding box). Figure 11 (b) shows a typical example of the effect.

(3) Decimation: For each 3D model, randomly select 20% polygons to be deleted. Figure 11 (c) shows a typical example of the effect.

Experimental result of the robustness evaluation is shown in Figure 12. Clearly, our approach is robust against similarity transformation, noise and decimation.

5. Conclusion and Future Works

In this paper, a 3D model retrieval system is proposed based on visual similarity. The new metric based on a set of *Light-Field Descriptors* is proposed for matching among 3D models. The visual similarity-based approach is robust against translation, rotation, scaling, noise, decimation and model degeneracy etc. A practical retrieval system that includes more than 10,000 3D models is available on the web for expert and novice users, and the retrieval can be done in less than 2 seconds on a server with Pentium IV 2.4 GHz CPU. A friendly user interface is also provided to query by drawing 2D shapes. The experimental results demonstrate that our approach outperforms 3D Harmonics, MPEG-7 Shape 3D Descriptor and Multiple View Descriptor.

In future work, several investigations are described as follows. First, other image metric for 2D shapes matching may be evaluated and included to improve the performance. In addition, the image metric for color and texture [11] can also be included to retrieval 3D model using more visual features. Second, different approaches ("cocktail" approach) can be combined to improve the overall performance. Third, the mechanism of training data or active learning [12, 13] may be used to adjust the weighting among different features. Finally, partial matching from several objects takes a long time to compute in general, and is also an important and difficult research direction in the future work [28, 29].

Acknowledgements

We would like to thank Miss Wan-Chi Luo, a graduate student, to help us manually classify the 1833 3DCafe objects into 47 classes plus another class "miscellaneous". We also appreciate Jeng-Sheng Yeh to set up the web server for our 3D model retrieval system. This project is partially funded by the National Science Council (NSC, Taiwan) under the grant number NSC91-2622-E-002-040 and by the Ministry of Education under the grant number 89-E-FA06-2-4-8.

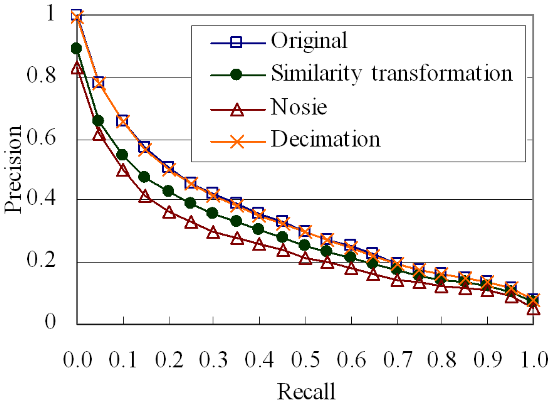

Figure 12: *Robustness evaluation of similarity transformation, noise and decimation*

References

1. T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin and D. Jacobs, "A Search Engine for 3D Models", *ACM Transactions on Graphics*, **22**(1):83-105, Jan. 2003.

2. R. Osada, T. Funkhouser, B. Chazelle and D. Dobkin, "Shape Distributions", *ACM Transactions on Graphics*, **21**(4):807-832, Oct. 2002.

3. M. Hilaga, Y. Shinagawa, T. Kohmura and T. L. Kunii, "Topology Matching for Fully Automatic Similarity Estimation of 3D Shapes", *Proc. of ACM SIGGRAPH*, 203-212, Los Angeles, USA, Aug. 2001.

4. E. Paquet, M. Rioux, A. Murching, T. Naveen and A. Tabatabai, "Description of Shape Information for 2-D and 3-D Objects", *Signal Processing: Image Communication*, **16**:103-122, Sept. 2000.

5. R. Ohbuchi, T. Otagiri, M. Ibato and T. Takei, "Shape-Similarity Search of Three-Dimensional Models Using Parameterized Statistics", *Proc. of 10th Pacific Graphics*, 265-273, Beijing, China, Oct. 2002.

6. D. V. Vranic and D. Saupe, "Description of 3D-Shape using a Complex Function on the Sphere", *Proc. of IEEE International Conference on Multimedia and Expo (ICME)*, 177-180, Lausanne, Switzerland, Aug. 2002.

7. I. Kolonias, D. Tzovaras, S. Malassiotis and M. G. Strintzis, "Fast Content-Based Search of VRML Models based on Shape Descriptions", *Proc. of Interna-*

tional Conference on Image Processing (ICIP), 133-136, Thessaloniki, Greece, Oct. 2001.

8. B. K.P. Horn, "Extended Gaussian Images", *Proceedings of the IEEE*, **72**(12):1671-1686, 1984.
9. M. Ankerst, G. Kastenmuller, H.-P. Kriegel and T. Seidl, "3D Shape Histograms for Similarity Search and Classification in Spatial Databases", *Proc. of 6th International Symposium on Advances in Spatial Databases (SSD)*, Hong Kong, China, 207-228, 1999.
10. D.-Y. Chen and M. Ouhyoung, "A 3D Object Retrieval System Based on Multi-Resolution Reeb Graph", *Proc. of Computer Graphics Workshop*, 16, Tainan, Taiwan, June 2002.
11. S. Jeannin, L. Cieplinski, J. R. Ohm and M. Kim, *MPEG-7 Visual part of eXperimentation Model Version 7.0, ISO/IEC JTC1/SC29/WG11/N3521*, Beijing, China, July 2000.
12. M. Elad, A. Tal, and S. Ar. "Content Based Retrieval of VRML Objects - An Iterative and Interactive Approach", *Proc. of 6th Eurographics Workshop on Multimedia*, 97-108, Manchester UK, Sept. 2001.
13. C. Zhang and T. Chen, "An Active Learning Framework for Content-Based Information Retrieval", *IEEE Transactions on Multimedia Special Issue on Multimedia Database*, **4**(2):260-268, June 2002.
14. P. Lindstrom and G. Turk, "Image-Driven Simplification ", *ACM Transactions on Graphics*, **19**(3):204-241, July 2000.
15. G. Turk, "Generating Textures on Arbitrary Surfaces Using Reaction-Diffusion", *Computer Graphics (Proc. of ACM SIGGRAPH)*, **25**(4):289-298, July 1991.
16. S. A. Dudani, K. J. Breeding and R. B. McGhee, "Aircraft Identification by Moment Invariants", *IEEE Transactions on Computers*, **C-26**(1):39-46, Jan. 1977.
17. T. P. Wallace and P. A. Wintz, "An Efficient Three-Dimensional Aircraft Recognition Algorithm Using Normalized Fourier Descriptors", *Computer Graphics and Image Processing*, **13**:99-126, 1980.
18. C. M. Cyr, B. B. Kimia, "3D Object Recognition Using Shape Similarity-Based Aspect Graph", *Proc. of International Conference on Computer Vision (ICCV)*, 254-261, Vancouver, Canada, July 2001.
19. C. E. Jacobs, A. Finkelstein, D. H. Salesin, "Fast Multiresolution Image Querying", *Proc. of ACM SIGGRAPH*, 277-286, Los Angeles, USA, Aug. 1995.
20. D. S. Zhang and G. Lu. "A comparative Study of Fourier Descriptors for Shape Representation and Retrieval". *Proc. of 5th Asian Conference on Computer Vision (ACCV)*, 652-657, Melbourne, Australia, Jan. 2002.
21. D. S. Zhang and G. Lu. "An Integrated Approach to Shape Based Image Retrieval". *Proc. of 5th Asian Conference on Computer Vision (ACCV)*, 652-657, Melbourne, Australia, Jan. 2002.
22. D. Heesch and S. Ruger, "Combining Features for Content-Based Sketch Retrieval - A Comparative Evaluation of Retrieval Performance", *Proc. of 24th BCS-IRSG European Colloquium on IR Research Glasgow (LNCS 2291)*, UK, Mar. 2002
23. A. Gershun, "The Light Field", Moscow, 1936. Translated by P. Moon and G. Timoshenko in *Journal of Mathematics and Physics*, **18**:51-151, MIT, 1939.
24. L. McMillan and G. Bishop, "Plenoptic Modeling: An Image-Based Rendering System", *Proc. of ACM SIGGRAPH*, 39-46, Los Angeles, USA, Aug. 1995.
25. M. Levoy and P. Hanrahan, "Light Field Rendering", *Proc. of ACM SIGGRAPH*, 31-42, New Orleans, USA, Aug. 1996.
26. T. Igarashi, S. Matsuoka and H. Tanaka, "Teddy: A Sketching Interface for 3D Freeform Design", *Proc. of ACM SIGGRAPH*, 409-416, Los Angeles, USA, Aug. 1999.
27. F. Huber and M. Hebert, "Fully Automatic Registration of Multiple 3D Data Sets", *Proc. of IEEE Workshop on Computer Vision Beyond the Visible Spectrum: Methods and Applications (CVBVS)*, Kauai, Hawaii, USA, Dec. 2001.
28. A. E. Johnson and M. Hebert, "Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **21**(5):433-449, May 1999.
29. S. M. Yamany and A. A. Farag, "Surfacing Signatures: An Orientation Independent Free-Form Surface Representation Scheme for the Purpose of Objects Registration and Matching", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(8):1105-1120, Aug. 2002.
30. A. E. Bimbo, *Visual Information Retrieval*, Morgan Kaufmann Publishers, Inc., 1999.
31. T. Pavlidis, *Algorithms for Graphics and Image Processing*, Computer Science Press, 1982.
32. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Addison-Wesley Pub. Co., 1992.
33. I. T. Jolliffe, *Principal Component Analysis*, 2nd edition, Springer, 2002.
34. 3DCAFE, http://www.3dcafe.com
35. SpharmonicKit 2.5: Fast spherical transforms. http://www.cs.dartmouth.edu/∼geelong/sphere/