A Retrieval System for Searching 3D Models from Web

Liu Yang, Jiale Wang, Xiang Li

College of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou, China

liuyang1012@aliyun.com, wjl8026@zjgsu.edu.cn, xiangli76@aliyun.com

Keywords: 3D Model; Content-Based Retrieval; Relevace Feedback

Abstract. As there are an increasing number of 3D models available on web, it is necessary to develop a system to help people find and get them. In this paper, we propose a system for content-based retrieval of 3D models on web. Users can submit an example 3D model to the retrieval system, and it will return a ranked list of similar models. We use a clawer to search for 3D models and download them to a local database. The shape feature vectors of these models are extracted, and the similarity between 3D models can be measured by the distance between their shape feature vectors. A relevance feedback algorithm is employed to optimize the retrieval performance. A prototype system of the proposed search engine is built.

Introduction

As the volume of multimedia data is rapidly growing, many content-based information retrieval systems have been developed for a desired kind of media data (e.g. images, audio or video) [1, 2, 3]. 3D modeling and digitalizing techniques have made great progress over the last few decades. 3D models are much easier to construct than before. As a result, the number of 3D models available on the web grows rapidly. To help people find and get them, a search engine is needed for automatic content-based retrieval of 3d shape data.

This paper describes the implementation of a 3D model search engine. The key challenge of our work is how to describe the shape information of 3D models and measure their dissimilarity effectively. We employ a 3D shape descriptor based on orthogonal projections to discriminate 3D models. The descriptor uses the projected images of a model to represent its shape. Thus, the 3D shape representation is converted to a 2D one, and we can resort to CBIR (content-based image retrieval) method to solve our problem.

System Overview

The overall structure of the search engine is illustrated in Figure 1.

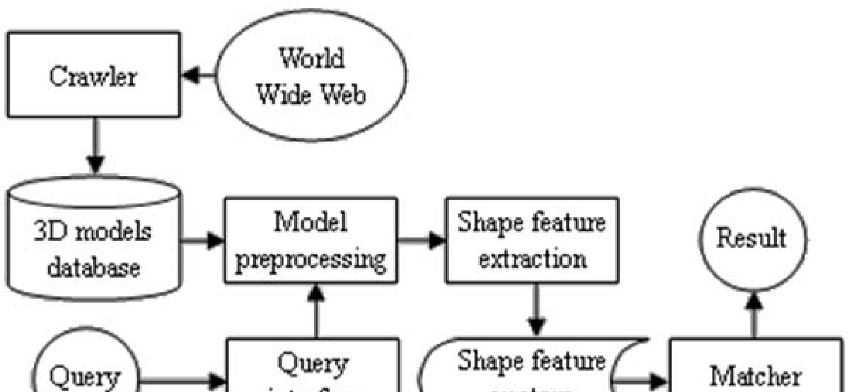


Fig. 1 Structure diagram of the 3D model search engine

The crawler searches for links pointing to files with a 3D model file extension, and then download these 3D model files to a local database. The crawler of our prototype system searches for three types of 3D model file: wrl (VRML), 3ds (3d studio) and lwo (lightwave). To do query, users submit an example model to the query interface. The queried model follows the same processing flow as models in the database until their dissimilarity is compared.

Before the Shape feature extraction module computes shape feature vectors, 3D models need to be pose-normalized by the model preprocessing module. The Matcher compares the distance between the shape feature vectors of the queried model and that of models in the database, and returns a ranked list of similar models. In following section, we will discuss the model preprocessing, Shape feature extraction and Matcher in detail.

Model preprocessing

3D models may have arbitrary position, size and orientation in 3D space. Before projecting, they need to be pose-normalized. To normalize the position, 3D models are translated from the center of the model to the origin of coordinates. To normalize the size, models are scaled to make its maximum radius equate to 1 unit. To normalize the orientation, we use the PCA method [4]. Support that the set of vertices of a 3D model is represented as: $P = \{P_1, P_2 \cdots P_m\}(P_j = [x_j, y_j, z_j] \in R^3)$, we calculate the covariance matrix $C = \sum_{k=0}^{m} P_k^T P_k$.

Then we sort the eigenvalues of C in a non-increasing order and find the corresponding eigenvectors. The eigenvectors are normalized to 1 unit and we construct a rotation matrix *R* in which the normalized eigenvectors are put in rows. To normalize the orientation, The vertices of the 3D model is rotated by the matrix *R*, so that the x-axis maps to the eigenvector with largest eigenvalue, the y-axis maps to the eigenvector with second largest eigenvalue, and the z-axis maps to the eigenvector with smallest eigenvalue. The rotation can be done according to the following formula: $P' = \{P_l' \mid P_l' = P_l R, P_l \in P, l = 0, \cdots, m\}$, in which the *P'* is the vertices set of the rotated 3D model.

Shape feature extraction and the Matcher

To discriminate different 3D models, a shape descriptor is needed for represent the shape feature of 3D models. A shape descriptor is usually formed as a shape feature vector, and the dissimilarity between 3D models can be measured by the distance between their shape feature vectors. For our search engine, we use a shape descriptor based on orthogonal projections. The basic idea of this method is that we can compare two 3D models by comparing their projected images.

When the pose normalization is done, we can project the 3D model onto 3 mutually orthogonal planes and get 3 projected images. To describe the shape information of a projected image, we first get the contour of the projected image, and then apply the Polar Fourier Transformation [5] on the centroid distance function [6] of the contour. The centroid distance function is expressed by the distance of the boundary points to the centroid of the contour.

The fourier coefficients can be used to represent the shape information of the projected image. We choose the first 12 coefficients, as the high frequency part is not steady and changes drastically when the contour has even a slight difference. The shape feature vector v of a 3D model is formed by concatenating the 12 fourier coefficients of its 3 projected images. To compare the dissimilarity of 3D models, the Matcher module compute the city block distance [7] between their feature vectors.

Matcher with relevance feedback

In many cases, the images of a 3D model varies in the amount of shape information. We can see that in Figure 2 the image of the projection along x-axis contain less shape information, while the other two images of the projection along y-axis or z-axix contain more shape information. The images that contains more information plays a more important role in 3D shape similarity comparison, thus we should assign a weight to each projection and give more weights to the projections with richer shape information:

$$D = w_1 disImg(s_1, v_1) + w_2 disImg(s_2, v_2) + w_3 disImg(s_3, v_3)$$

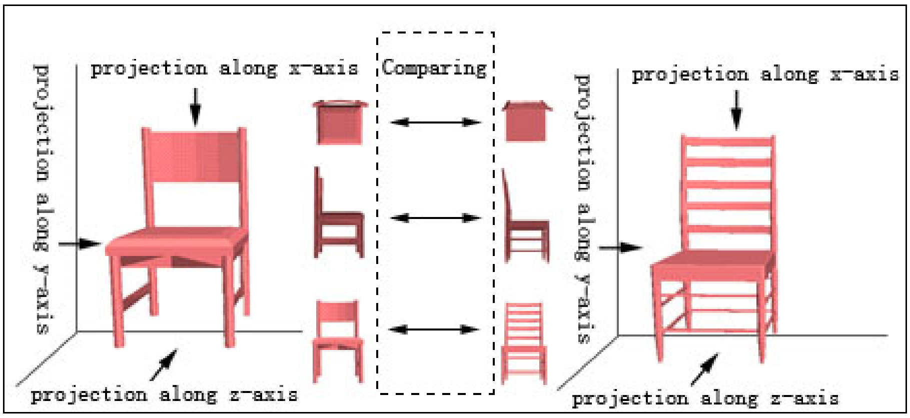


Fig. 2 3D models and projection

We use the RF (Relevance Feedback) mechanism to determine the weights of each projection. RF makes the search process as an interaction between the computer and users. For a RF based search process, the system first retrieves similar models and returns them to user. Then, the user provides feedback regarding the relevance of some of the retrieval results (user mark the relevant objects in the results and submit them back to the search system). Finally, the system uses the feedback information to improve the performance in the next iteration. Figure 3 illustrate the diagram of a retrieval system with RF.

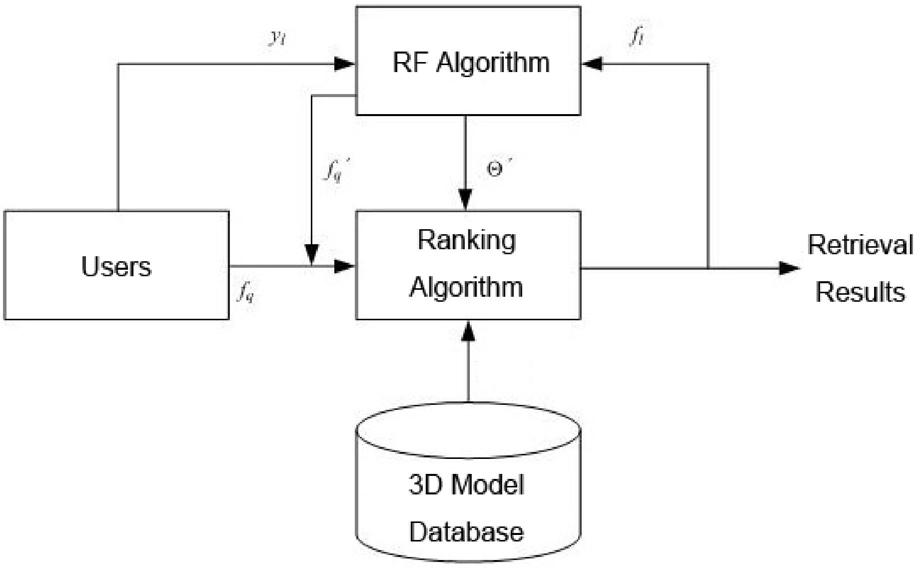
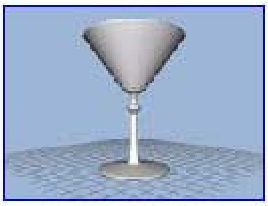


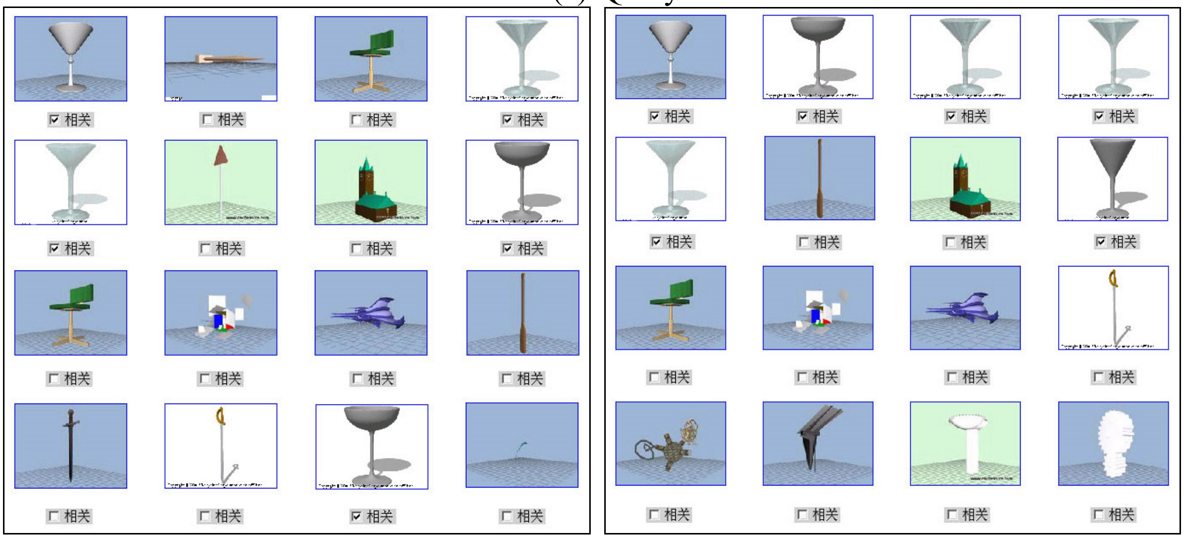
Fig. 3 Retrieval system with RF

Prototype system

We implement a prototype system. Users submit the queried model to the search engine, and it will return the relevant models, which are ordered left to right and top to bottom based on similarity distance from the query. Figure 4 shows the retrieval results returned by the system, according to the query of a human body.



(a) Query



(b) Initial retrieval results (c) Retrieval results after a iteration

Fig. 4 Query results returned by the prototype system

Figure 5 illustrates the precision recall curves of different iteration. We can the retrieval perform enhanced with the RF iteration increasing.

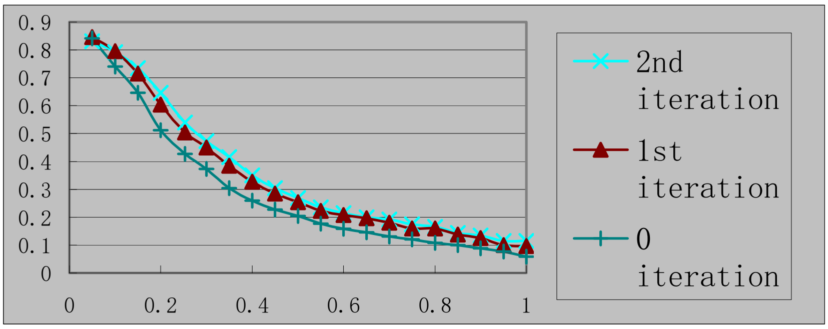


Fig. 5 Precision recall curves of different iteration

Conclusions

In this paper, we propose a system for content-based retrieval of 3D models on web. Users can submit an example 3D model to the retrieval system, and it will return a ranked list of similar models. We use a clawer to search for 3D models and download them to a local database. The shape feature vectors of these models are extracted, and the similarity between 3D models can be measured by the distance between their shape feature vectors. A relevance feedback algorithm is employed to optimize the retrieval performance.

At present the 3D models database of our prototype system contains about 600 models. In the future wok, we will make the crawler continue downloading more 3D models and evaluate the retrieval performance in a quantitative way.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No.61202200) and the Scientific Research Fund of Zhejiang Provincial Education Department (No.Y201018008).

References

[1] Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The Princeton Shape Benchmark. In Proceedings of Shape Modeling International (Genoa, Italy, June 7--9). IEEE, 2004, 167-178

[2] Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A., Dobkin, D., Jacobs, D.: A Search Engine for 3D Models. Transact. Graphics 22, 1 (Jan. 2003), 83-105

[3] Chen, D.-Y., Tian, X.-P., Shen, Y.-T, Ouhyoung, M.: On Visual Similarity-Based 3D Model Retrieval. Comput. Graphics Forum 22, 3 (Sept. 2003), 223-232

[4] Ohbuchi, R., Nakazawa, M., and Takei, T.: Retrieving 3D Shapes Based on Their Appearance. In Proceedings of MIR'03 (Berkeley, CA, USA, Nov. 2003), 39-46

[5] Turk, G.: Generating Textures on Arbitrary Surfaces Using Reaction-Diffusion. Computer Graphics 25, 4 (Proceedings of ACM SIGGRAPH, July 1991), 289-298

[6] Zhang, D.-S.: Image Retrieval Based on Shape. Phd dissertation, Monash University March, 2002.

[7] Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D Models with Shape Distributions. Shape Modeling International (2001), 154–166