

A 3D Model Retrieval Method Using 2D Freehand Sketches

Jiantao Pu and Karthik Ramani

Purdue Research and Education Center for Information Systems in Engineering (PRECISE),
Purdue University, West Lafayette IN 47907-2024
{pjiantao, ramani}@purdue.edu

Abstract. In this paper, a method is proposed to retrieve desired 3D models by measuring the similarity between a user's freehand sketches and 2D orthogonal views generated from 3D models. The proposed method contains three parts: (1) pose determination of a 3D model; (2) 2D orthogonal view generation along the orientations; and (3) similarity measurement between a user's sketches and the 2D views. Users can submit one, two or three views intuitively as a query, which are similar to the three main views in engineering drawing. It is worth pointing out that our method only needs three views, while 13 views is the minimum set that has been reported by other researchers.

1 Introduction

Up to this point, many methods have been proposed to retrieve the desired models from a database. These methods can be classified into four categories: feature-vector based method [1]; statistics-based method [2]; topology-based method [3]; and image-based method [4]. An advantage of the feature-based method is their simplicity, but there is no feature or feature set that can describe all 3D shapes in a uniform way. The statistics-based methods do not require pose registration and feature correspondence, and are fast and easy to implement, but they are not sufficient to distinguish similar classes of objects. From the perspective of structural similarity, topology-based methods have many desired properties, such as intuitiveness, invariance, and robustness. And not only global features but also local features are depicted. However, they require a consistent representation of an object's boundary and interior, and it is not easy to register two graphs robustly. The motivation of the image-based method is to imitate the ability of the human visual system to recognize objects. However, many images from different perspectives are needed.

In this paper, we propose a method to retrieve 3D models by measuring the similarity between a user's sketches and three 2D orthogonal views generated from 3D models. The idea arises from a practice: *engineers usually express their concept of a 3D shape with three 2D views without missing any information*. For this purpose, we present three algorithms: (1) pose normalization of 3D objects; (2) 2D drawing-like view generation; and (3) similarity measurement between 2D views. In the remainder of this paper, the approaches to the three problems will be described respectively, and some experimental results are presented to show its validity.

2 Pose Normalization

As a classical statistical method, principal component analysis (PCA) [5] is used to estimate the intuitive directions along which the mass is heavily distributed. However, it is not good enough at aligning orientations of different models within similar shapes. Extended Gaussian Image (EGI) [6] is another classical method to determine the pose of a 3D object. For a 3D object, each point on its Gaussian sphere corresponds to a particular surface orientation and the respective surface area. However, for nonconvex objects, different shapes may have the same EGI representation. To overcome the above-mentioned limitations, we propose a new orientation determination method. A 3D shape is represented by a triple $S = \{ p_i \mid (N_i, A_i, D_i), 0 \leq i \leq n \}$, in which N_i represents the normal of polygon p_i , A_i represents the area, and D_i represents the distance between the mass center C and the polygon p_i . Our aim is to find out the normal along which the summed surface area is the largest and these surfaces have the same distance from the mass center:

Step 1: Compute the normal direction N^k for each triangle $\Delta \mathbf{p}^k \mathbf{q}^k \mathbf{r}^k$ and normalize it. The normal of a triangle is equal to the cross product of its two edges:

$$N^k = \frac{\mathbf{p}^k \mathbf{q}^k \times \mathbf{q}^k \mathbf{r}^k}{\|\mathbf{p}^k \mathbf{q}^k \times \mathbf{q}^k \mathbf{r}^k\|} \tag{1}$$

Step 2: Summarize the areas of all triangles with the same normals and same distance from the center mass.

Step 3: Determine the three principal axes. The normal associated with the maximum areas is selected as the first principal axis \mathbf{b}^u . To get the next principal axis \mathbf{b}^v , we can search from the remaining normal distributions and find out the normal that satisfies two conditions: (a) with maximum areas; and (b) orthogonal to the first normal. The third axis can be obtained by doing cross product between \mathbf{b}^u and \mathbf{b}^v .

Step 4: Find the center and the half-length of the bounding box. This can be done by projecting the points of the convex hull onto the direction vector and finding the minimum and maximum along each direction. In Figure 1, some examples obtained by the MND method are shown.



Fig. 1. Some pose determination examples

3 2D Orthogonal View Generation

To compute the view of a 3D model on a plane, we design an algorithm and explain it as follows with the help of the example shown in Figure 2.

- Step 1:* Backface culling in object space. When engineers represent a 3D object using 2D views, the invisible backfaces are not considered. Given a projection direction \mathbf{n} and a polygon P_i with normal vector \mathbf{n}_i , if $\mathbf{n}_i \cdot \mathbf{n} > 0$, then this polygon is visible; otherwise, it is invisible. Figure 2(b) shows the backface culling result for the model shown in Figure 2(a).
- Step 2:* Inside-edge culling. To obtain the drawing-like view of 3D objects, the inside-edges have to be discarded. The inside-edge has a distinguishing property: *it is shared by two polygons*. With this definition, we can cull the inside-edges completely by transcending all the triangles. The result is shown in Figure 2(c).
- Step 3:* Orthogonal projection along the viewing direction, as Figure 2(d) shows. An example obtained by this method is shown in Figure 3.

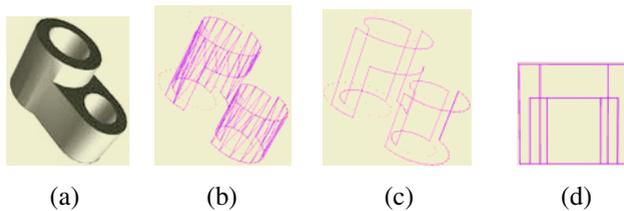


Fig. 2. Four steps for view generation: (a) a 3D model; (b) the result after backface culling; (c) the result after inside-edge culling; and (d) the generated 2D view

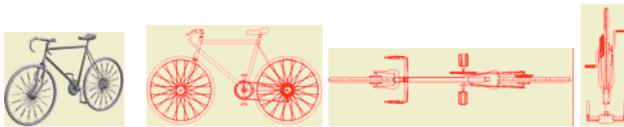


Fig. 3. A 2D view generation example

4 2D Shape Distribution Method for Similarity Measurement

To measure the similarity between 2D views, we propose a 2D shape histogram method to measure the similarity between 2D views:

- Step 1:* Uniform sampling on view edges. From the statistics perspective, a 2D shape can be approximated by enough sampled points. This process can be done by an unbiased sampling strategy similar to the method adopted in [7].
- Step 2:* Shape distribution generation. By summarizing the numbers of point pairs with the same distance, the 2D shape distribution can be generated.
- Step 3:* Similarity measurement. Generally, there are two normalization methods to account for the size difference between two views: (a) align the maximum D2 distance values, and (b) align the average D2 distance values. The similarity between the two views is measured by calculating the difference between their distributions in the form of a histogram.

5 Experimental Results

An intuitive application for this proposed method is sketch-based user interface, in which the query process is similar to when engineers represent a 3D shape on a piece of paper. In order to evaluate its validity and performance, some experiments have been conducted and show that our proposed method has many valuable advantages: (a) insensitive to geometric noise; (b) invariant to translation, rotation, and scaling; and (c) supportive of freehand sketch query. In Table 1, two retrieval examples using freehand sketches are presented:

Table 1. Two retrieval examples using freehand sketches

Sketches	Top Four Similar Models			
				
				

6 Conclusion

This paper presents a 3D model retrieval method by measuring the similarity between 2D views. The method enables the intuitive implementation of a 2D sketch user interface for 3D model retrieval. In the future, we will focus our attention on local shape matching, in which users can specify some local shape explicitly

References

1. Elad, M., Tal, A., Ar, S.: Content Based Retrieval of VRML Objects: An Iterative and Interactive Approach. Proc. 6th Eurographics Workshop on Multimedia 2001, Manchester, UK (2001) 107–118.
2. Paquet, E., Rioux, M.: Nefertiti: A Tool for 3-D Shape Databases Management. SAE Transactions: Journal of Aerospace 108 (2000) 387–393.
3. Hilaga, M., Shinaagagawa, Y., Kohmura, T., Kunii, T.L.: Topology Matching for Fully Automatic Similarity Estimation of 3D Shapes. Proc. SIGGRAPH 2001, Los Angeles, USA, (2001) 203–212.
4. Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A., Dobkin, D., Jacobs, D.: A Search Engine for 3D Models. ACM Transactions on Graphics, Vol.22 (1): 83–105 (2003).
5. Petrou, M., and Bosdogianni, P., “Image Processing: The Fundamentals,” John Wiley (1999).
6. Horn, B.K.P., “Extended Gaussian Images,” Proc. IEEE, Vol.72 (12): 1671–1686 (1984).
7. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Shape Distribution. ACM Transactions on Graphics, Vol.21 (4): 807–832 (2002).