Non-linear Summarization of a Database for 3D Model Retrieval

Toshiya Shimizu, Yoshiki Akaike, Jun Kobayashi, Ryutarou Ohbuchi
University of Yamanashi

ABSTRACT

In this paper, we propose a 3D shape model database system in which a user finds desired 3D shape model by browsing the visualization of a summary of the database. In the visualization, postage stamp (2D) images of 3D models in the database are arranged in the 3D space based on similarity of the 3D shapes. A 3D coordinate of a postage stamp image is computed from a shape feature vector of the 3D model, which typically has tens to hundreds of dimensions. To reduce dimension down to three while maintaining shape similarity, we employ a non-linear dimension reduction algorithm called Laplacian Eigenmaps [3] by Belkin, et al., which is an unsupervised non-manifold learning algorithm. To facilitate exploration by shape similarity, our system also provides the users with an optional annotation indicating a group of similar models.

CR Categories and Subject Descriptors:
Additional Keywords: 3D model database, content based retrieval, 3D geometric modeling, 3D computer graphics, non-manifold learning.

1 INTRODUCTION

3D shape models are now ubiquitous, from games, web contents, CAD data, to games on cellular phones. The trend promoted studies on shape-based retrieval of 3D models [2, 6].

A typical shape-based 3D model retrieval framework assumes some form of shape specification as a query, for example, a 3D model, a set of 2D sketches, or a 3D sketch. However, forming such a shape-based query may not be easy. For example, finding a sufficiently similar 3D model may be difficult (or, why not use the query example?) Sketching a complex 3D shape either in 2D or 3D could be difficult and/or cumbersome for many of us.

2 METHOD

The system visualizes 3D models in a 3D model database by following the steps below.

Extract shape feature vector: Extract shape feature vectors for the models in the training database.

Learn manifold: Using Belkin’s LE [3], perform an unsupervised learning of the mapping from the input n-dimensional space onto the m-manifold, in which m=3.

Visualization in 3D space: Place a postage stamp (2D) image of a 3D shape model rendered from a canonical view into a 3D space for interactive visualization.

Grouping (optional): Using k-means clustering algorithm, shapes the clustering algorithm thinks similar are marked by color. This guides the user in finding the shapes similar to the one already discovered.

2.1 Dimension Reduction Based on Manifold Learning

Belkin’s Laplacian Eigennaps (LE) performs dimension reduction of feature vectors by learning (estimating) the m-
dimensional manifold formed by the n-dimensional shape feature (n>m). It is hoped that the distance (dissimilarity) of 3D models computed on the found m-manifold may reflect the shape similarity better than the distance computed in the original n-dimensional feature space. (See Figure 2 for an illustration.)

LE finds the mapping from a n-dimension feature vector to a point on a m=3 dimensional manifold by the following the steps below;

1. **Construct an adjacency graph**: A feature vector of a model in the training database is a point in the n-dimensional feature space \( \mathbb{R}^n \). Construct an mesh \( G \) by connecting \( k \)-nearest points using Euclidian distance in \( U \).

2. **Create mesh Laplacian matrix for \( G \)**: Create a mesh Laplacian matrix \( L = D - W \), in which \( W \) is an adjacency matrix for \( G \),
\[
W_{ij} = \begin{cases} 
1, & \text{if vertices } i \text{ and } j \text{ are adjacent;} \\
0, & \text{otherwise.}
\end{cases}
\]
and \( D \) is the diagonal matrix satisfying the following equation.
\[
D_{ij} = \sum_j W_{ij}
\]

3. **Perform Eigenanalysis of \( L \)**: Find eigenvalues \( \lambda_i \) (1 \( \leq \) i \( \leq \) n) and eigenvectors \( f_i \) (1 \( \leq \) i \( \leq \) n) of \( L \) by solving the generalized eigenproblem \( LF = \lambda DF \).

4. **Find a mapping \( g \)**: Sort eigenvectors in an ascending order. Using the least \( m \) eigenvectors (but excluding the first eigenvector \( f_1 \)), find a mapping \( g : x_i \rightarrow \{ f_1(i), ..., f_m(i) \} \) that maps a point \( x_i \in U = \mathbb{R}^n \) onto \( m \)-manifold \( D \).

**Figure 2.** Computing distance among shape models as a geodesic distance on a manifold formed by shape feature vectors.

**2.2 Visualization in 3D**

Using the 3D coordinate of each 3D shape models in the database found by the LE, the system places postage stamp 2D images of the models in 3D space for visualization. Each image is rendered from a canonical view, and rendered on a “billboard” rectangular polygon. A billboard polygon rotates so that it always faces the camera (the viewer) during the 3D projective transformation. We employed perspective transformation to convey the distance from the viewpoint.

To further facilitate the exploration, we present an optional “group” annotation. As the user clicks a model, a group of models similar to the model are marked by colored border. The group of models is selected by using a clustering algorithm. We currently use the \( k \)-means algorithm with user-defined \( k \).

**3 RESULTS**

We used the Absolution Angle-Distance histogram (AAD) shape feature by Ohbuchi, et al. [4] and the Surflet Pair Relation Histogram (SPRH) shape feature by Wahl, et al. [7] for the experiment. The AAD is a 256 D vector, while the SPRH is a 625 D vector.

Figure 3 shows the projection of Princeton Shape Benchmark [6] training set 907 models by using the AAD and the SPRH shape features. Figure 4 shows the group formed by selecting the top-left model for the AAD shape feature.

A subjective evaluation showed that, while the SPRH performs better in some 3D model retrieval experiments using 3D models as query, several users favored the AAD shape feature in our explorative retrieval system. It is possible that the 3D projection of the AAD feature fits human notion of shape similarity better.

**4 CONCLUSION**

In this paper we proposed an approach for 3D shape retrieval by database summarization that combined non-linear dimension reduction and 3D visualization.

In the future, we would like to enhance both the dimension reduction technique and the visualization technique. Also, we have to find visualization that scales to a large 3D model database containing millions of 3D models. We would also need to perform quantitative performance evaluation.

**REFERENCE**


