SHAPE SIMILARITY SEARCH FOR SURFEL-BASED MODELS

M. R. Ruggeri, D. V. Vranić, and D. Saupe
Department of Computer and Information Science, University of Constance, D-78457 Konstanz, Germany

Abstract: We present a 3D object retrieval system in which a surfel-based model serves as a query and similar objects are retrieved from a collection of surfel-based models. The system comprises a surfelization technique for converting polygonal mesh models into a corresponding surfel-based representation, pose normalization of surfel-based models, a depth buffer-based method for describing shape of surfel-based models, and a search engine. Our surfelization technique consists of an enhanced triangle rasterization procedure adapted on the original triangulated model geometric features. Surfel-based representation is normalized by applying a modification of the Principle Component Analysis to the set of surfels. The 3D-shape descriptor is extracted from the canonical coordinate frame of the surfel-based model using the orthographic depth images. Each image is used as the input for the 2D Fast Fourier Transform. Appropriate magnitudes of the obtained coefficients are used as components of the feature vector. The resulting feature vector possesses an embedded multi-resolution representation, is invariant with respect to similarity transforms and robust with respect to outliers. The retrieval effectiveness of the presented method for characterizing shape of surfel-based models is evaluated using precision-recall diagrams.

Key words: point-based graphics; 3D model retrieval; point sampling.

1. INTRODUCTION

Point-based representation of 3D objects has recently received a lot of attention by the computer graphics community. The advantages of the point-based representation are mostly appreciable in representing and rendering highly detailed surfaces[1,2] where the rendering of traditional primitives (polygons) amounts to less than a pixel per primitive. We consider surfels as our principal point-based primitives. Surfels (SURFace ELements introduced by Pfister et al.[1]) are a powerful paradigm that allows representing and collecting different information about the characteristics of the surface in the neighbourhood of each sample point.

The number of surfel-based models is swiftly increasing. In the near future, this will result in the development of large and complex databases of surfel-based models requiring powerful retrieval systems, which will be able to perform appropriate and efficient queries. A 3D model retrieval system extracts several low-level features for each model, and measures the similarity between any two models in the low-level feature space. These low-level features are represented in a vector called *feature vector*. So far, shape similarity search of surfel-based models has not been studied in depth, even though the literature provides a rich variety of shape similarity methods for polygonal mesh models retrieval[3].

Our motivation is to propose a 3D object retrieval system in which a surfel-based model serves as a query and similar objects are retrieved from a collection of surfel-based models. To achieve our goal we need to extract an appropriate and compact feature vector from each surfel-based model and test its retrieval effectiveness on a database of such models. Such databases are not yet available and creating surfel-based models is still a difficult task. Thus, we converted already existing databases of triangular mesh models by using a surfelization procedure. Surfelization is the process of extracting surfels from a generic geometric representation of a 3D model (e.g., triangular mesh, implicit surface, parametric surfaces, CSG, etc.).

We propose a geometric feature preserving surfelization method, which is able to automatically convert huge databases of triangulated 3D models into the corresponding surfel-based representation. The main idea of our surfelization method is to follow the triangulation of the original triangulated models so that all the geometric structures of the models can be appropriately sampled. In fact, we do not know a priori what the source of our triangular meshes is. Therefore, by following this approach we ensure a correct geometric conversion preserving all the shape characteristics of the original models.

As shape descriptor of a surfel-based model we propose an extension of a depth buffer image-based descriptor for polygonal models[4], which produced good retrieval results. First, each surfel-based model is normalized and aligned with respect to its principal component axes by applying a modification of the Principle Component Analysis to the set of surfels. A normalized depth buffer image is then extracted from each of the six sides of a suitable bounding cube by using orthographic projection. The resulting depth buffer images are processed with the 2D Fourier transform. The feature vector is built by appropriately sampling the magnitude of the

Fourier coefficients. The resulting feature vector is invariant with respect to similarity transforms, and robust with respect to outliers and level of detail. The retrieval effectiveness of the presented method for characterizing shape of surfel-based models is evaluated using precision-recall diagrams. The retrieval performance of our feature vector has then been compared with other different feature vectors (including the one based on depth-buffer images) extracted from the original triangulated 3D models.

2. PREVIOUS WORK

Extraction of surfel-based representations from geometric models has been studied in different settings. Some techniques require complicated computations with the geometry and/or image space[1,5]. Other techniques are based on estimations of the local curvature or on a local parametric approximation of the underlying surface[2,6]. Turk[7] randomly spreads sampled points on triangular surfaces, and applies a relaxation procedure to optimize their position. The last category of surfelization techniques are conceived to be performed on the fly exploiting graphics hardware[8,9]. Cohen et al.[9] present a triangle rasterization procedure that, given the distance between samples, covers the entire triangle surface with square tiles adapted to their edges. Our algorithm for surfelization of triangular meshes is inspired by[9] as we use an adaptive triangle rasterization process, and by[7] as our method is driven by the curvature and the geometric characteristic of the triangular surface.

Point-based model retrieval has not yet been explored in depth. So far, only few matching techniques have been proposed for point-based models: based on differential geometry and algebraic topology[10] and based on sampling a segmentation of the model built exploiting the Delaunay triangulation[11]. On the other hand a lot of work has been done to retrieve polygonal models. We classify methods for shape similarity of 3D models in three general categories: geometry-based, topology-based, and image-based. The geometry-based approach is characterized by using shape descriptors based on the geometric distribution of vertices or polygons[12,3,4]. Topology-based methods establish the similarity of two 3D models by comparing their topological structures[13,3]. Finally, image-based approaches consider two models similar if they look similar from any angle. Generally, these approaches extract a feature vector from a set of images obtained projecting 3D models onto different projection planes[14,3,4]. We refer to[3] for a more extensive overview of shape descriptors and to[15,4] for a detailed comparison between the different methods mentioned above. Recently shape benchmarks have been published for comparing different shape descriptors[15,16], which show the promising retrieval effectiveness of image-based methods[15]. In order to retrieve surfel-based models, we modify a depth-buffer image-based approach[4], which shows high retrieval effectiveness for triangle mesh models (Section 4).

3. SURFELIZATION

Our system comprises a surfelization technique for converting already available retrieval test databases of triangular mesh models into a corresponding surfel-based representation. The goal of our surfelization technique is to sample an arbitrary triangular surface with surfels, by preserving both its geometric properties and its contour properties. This will ensure a correct shape approximation of the original triangulated model.

The main idea of our surfelization algorithm is to follow the original triangulation, by sampling triangles with respect to the following geometric features:

- crease edges and boundary edges;
- smooth local triangular surface at a vertex.

An edge is considered as a crease edge, if the dihedral angle θ between two adjacent triangles exceeds a predefined threshold θ_t. An edge is detected as a boundary edge if it belongs to only a single triangle. We call these two types of edges *preserved edges*.

For each vertex v, we consider all triangles sharing the vertex v as our local triangular surface Σ. We then compute the average plane π, formed by all the vertices belonging to Σ[17]. The surface Σ is considered as non-smooth, if the orthogonal distance between the vertex v and the plane π exceeds a threshold d_t.

The sampling algorithm starts from a seed vertex and samples the underlying local triangular surface near the features with a specific pattern. The remaining planar surface is sampled on a regular grid.

Triangles sharing a preserved edge are sampled exactly along the edge line (Figure 1a). Surfel disks are then placed along the edge line. The radius and the position of each surfel are selected in order to keep the over edge error e (Figure 1b) under a threshold e_{max}. Furthermore, surfels radii must be in a predefined range $[r_{min}, r_{max}]$.

When a triangular surface is locally smooth, a surfel is centred on each of its vertices parallel to the average fitting plane[17] of its neighbour vertices. To select the radius of the surfel we consider two cases. In the first case where at least one triangle sharing the vertex has a preserved edge (see Figure 1c), the radius of the surfel is set to the minimum length of the connected edges (r in Figure 1c). In the other case we set the radius to the maximum length of the connected edges (r in Figure 1d). In this last case, we have also to ensure that preserved edges that belonging to some neighbour triangle, do not overlap the surfel disk (see Figure 1d). In both cases the radius r is cropped to be in the range $[r_{min}, r_{max}]$, and the remaining unfilled area is sampled on a regular grid.

Beside the two special cases mentioned above, the remaining triangles are sampled on a regular squared lattice without covering the already sampled area near the preserved features. Basically, this is a rasterization process, where surfel disks are placed onto the centre of each output pixel with a radius equal to half the pixel diagonal. This technique is parallelizable and can be easily implemented exploiting the rasterization process of graphics hardware.

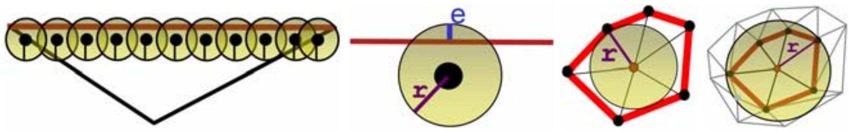


Figure 1. Surfelization near preserved edges (a, b) and of a smooth local surface (c, d).

4. SHAPE SIMILARITY SEARCH

Our retrieval system aims to efficiently query a database of surfel-based models. An arbitrary surfel-based model serves as a query key to retrieve similar models from a collection of surfel-based models. Our retrieval method consists of three steps: normalization (pose estimation), feature extraction and similarity search.

Normalization (pose estimation). Surfel-based models are given in arbitrary units of measurement and undefined positions and orientations. The normalization procedure aims to transforms a model into a canonical coordinate frame, so that if one chose a different scale, position, rotation, or orientation of a model, then its representation in such coordinates remains the same. Moreover, the normalized representations corresponding to different levels of detail of the same model should be similar as much as possible.

We normalize each surfel-based model by applying a modification of the Principle Component Analysis (PCA) to the set of surfels. Following the usual PCA procedure we compute the covariance matrix C through the following formulae:

$$C = \frac{1}{S}\sum_{i=1}^{N}(\vec{c}_i - \vec{g})(\vec{c}_i - \vec{g})^T S_i \qquad \vec{g} = \frac{1}{S}\sum_{i=1}^{N}\vec{c}_i S_i \qquad S = \sum_{i=1}^{N} S_i \qquad (1)$$

where $\vec{g}$ is the center of gravity and S is the surface area of the entire model, N is the number of surfels, S_i and $\vec{c}_i$ are the area and the center of the ith surfel, respectively. C is a 3x3 symmetric, real matrix. The eigenvalues of C are then computed and sorted in decreasing order. From C we build the rotation matrix R, which has the normalized eigenvectors as rows with nonnegative elements on the main diagonal. First we center the surfel-based model onto the point $\vec{g}$ by translating the surfel center points $\vec{c}_i$. Then, we apply the transformation defined by the rotation matrix R to the surfel centers points and to the surfel normal vectors $\vec{n}_i$. The resulting surfel-based model is now in canonical coordinate, which is invariant to translation and rotation.

Feature extraction. The features we extract represent the 3D shape of the objects present in a model. These features are stored as vectors (*feature vector*) of fixed dimension. There is a trade-off between the required storage, computational complexity, and the resulting retrieval performance.

We adapted the approach proposed in[4] in which a shape descriptor is extracted for each model from the canonical coordinate frame using six orthographic depth images. The depth images are computed by orthogonally projecting the surfel-based model onto the six different faces of a cube establishing an appropriate region-of-interest. The size of the resulting depth images is 256x256 pixels. This operation can be easily performed exploiting a graphics hardware z-buffer.

The choice of the region-of-interest cube is very important and may affect the retrieval performance. We propose an approach that is robust with respect to outliers. The cube is aligned with the three canonical axes and centered at the center of gravity $\vec{g}$ (defined in Formula 1). The size of the cube, d_f, is proportional to the average distance d_{avg} of the surfels to the center of gravity, $\vec{g}$.

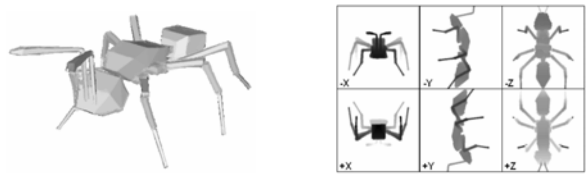
$$d_{avg} = \frac{1}{S}\sum_{i=1}^{N} \|\vec{c}_i - \vec{g}\| S_i \qquad\qquad d_f = f \cdot d_{avg} \tag{2}$$

Here f is a fixed ratio and we have used $f = 3.9$. Our experience suggests that this approach is superior to using the standard bounding cube, which may sensitively depend on outliers of the model.

Figure 2 shows the model of an ant sampled with the corresponding six depth buffer images. It shows that the two antennae of the ant have been clipped by our cube, which improves retrieval performance by eliminating outliers.

We apply the 2D discrete Fourier transform to each depth buffer image to obtain the corresponding Fourier spectrum. The feature vector is then built by storing, for each processed image, the magnitudes of low frequencies

Fourier coefficients as described in[4]. According to experimental results[4] we set the dimension of our feature vector to 438: we store 73 real numbers for each image.



(a) (b)

Figure 2. Surfel-based model (a) and the extracted depth buffer images (b).

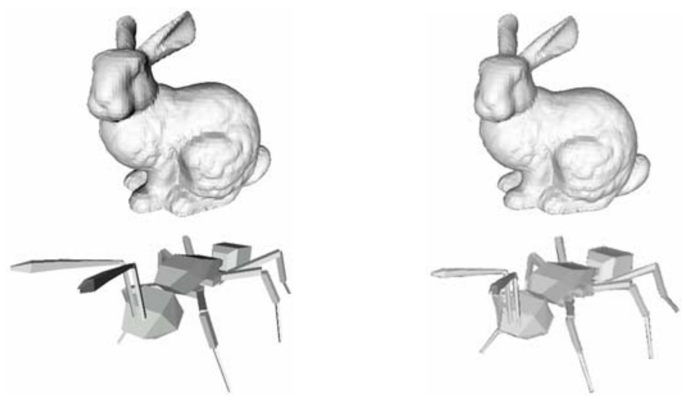
Similarity search. The extracted features are designed so that similar 3D-objects are close in feature vector space. Using a suitable metric nearest neighbors are computed and ranked. In our case the distance between two feature vectors is computed by using the l_1 or l_2 norm. A variable number of objects are thus retrieved by listing the top ranking items.

5. RESULTS

We have converted two test databases of triangular mesh models. The first database consists of 1814 3D polygonal models provided by the Princeton Shape Benchmark repository[15]. The second database consists of 1841 polygonal models provided by the CCCC repository[16]. These databases contain triangulated 3D models at different resolutions and with very different numbers of triangles. Both databases have been converted in less than two hours including the input/output time. The time required for the surfelization was 98.28 minutes for the first database, and 100.74 minutes for the second database, on a 2.4 GHz Intel Pentium 4 system with 2 GB RAM using a non-optimized C++ version of our program, and software only rasterization. Table 1 shows the execution time (in seconds) and the number of surfels produced by our surfelization technique together with the number of geometric primitives (vertices and triangles) of the original triangular mesh. The fourth column shows the number of the geometric features extracted from the triangulated models.

Table 1. Surfelization performance

Model	Vertices	Triangles	Features	Surfels	Time (s)
Bunny	7691	15294	13800	283604	2.18
Ant	298	492	347	14202	0.09
Fish	18409	35270	31563	325851	5.16
Airplane	13908	27087	24185	470660	7.57



(a) (b)

Figure 3. Result of the surfelization: the original triangulated models (a) and the corresponding surfel-based representations (b).

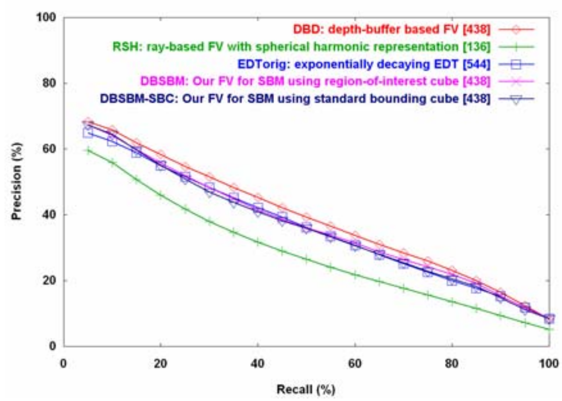


Figure 4. Precision-recall diagram computed for different feature vectors of the original triangles meshes with respect the proposed feature vector extracted from the corresponding surfel-based models. Dimension of feature vectors are given in brackets.

Figure 3 shows two different triangular mesh models (Figure 3a) and the corresponding surfel-based models (Figure 3b). As we can see both geometrically and visually the features of the original models have been preserved.

The retrieval effectiveness of the presented method for characterizing shape of surfel-based models is evaluated by using precision-recall diagrams. We have compared our method with other retrieval methods for polygonal mesh models. Figure 4 shows the precision-recall diagram we use to compare these different feature vectors. These results refer to retrieval tests performed on the polygonal models of the original Princeton Shape Benchmark test database, and on the corresponding surfel-based representation. We have compared our retrieval system for surfel-based models with three different feature vectors extracted from the original triangulated models. The first feature vector is based on depth buffer images[4] (DBD in Figure 4). The second is a ray based feature vector with spherical harmonics representation[18] (RSH in Figure 4). The third feature vector is computed by using the exponentially decaying EDT technique[12] (EDTorig in Figure 4). For surfel-based models we have considered two feature vectors. The first is extracted by using our region-of-interest cube (DBSBM in Figure 4) and the second is extracted by using the standard bounding cube (DBSBM-SBC in Figure 4). The diagram shows that the retrieval performance of the depth buffer image-based descriptor we designed for surfel-based models is approximately the same as the performance of the corresponding descriptor of polygonal mesh objects.

6. CONCLUSION AND FUTURE WORK

We have presented a retrieval system for surfel-based models. The system includes: a surfelization technique for converting polygonal mesh models into a corresponding surfel-based representation, pose normalization of surfel-based models, a depth-buffer image-based method for describing shape of surfel-based models, and a search engine. With the presented surfelization technique we have converted large databases of polygonal mesh models used for retrieval purpose into corresponding databases of surfel-based models in an acceptable period of time. Converting an already classified database allows to skip the classification step, which is commonly done manually and it is time consuming. Our surfelization procedure preserves the geometric features of the original models, but does not consider their texture properties. Even if this fact is not so crucial for shape analysis and retrieval purposes, it is fundamental for realistic surfel-based model visualization. Future work will be driven in this direction.

We have also proposed a suitable feature descriptor for surfel-based models. Our feature vector is derived from the image-based shape descriptors used for polygonal mesh models[4]. This approach obtained approximately the same good retrieval results with surfel-based models, as shown by our retrieval tests. However, our main goal will be to conceive further feature vectors, which are strictly based on the intrinsic geometric properties of set of surfels. Our retrieval system[16,4] offers a user-interface that enables to non-specialist users an easy and effective access to 3D geometry database. We plan to extend our system to support also different point-based model representations.

AKNOWLEDGMENTS

This work was supported by the strategic research initiative on Distributed Processing and Delivery of Digital Documents funded by the German Research Foundation, the Kurt Lion Foundation, and the DFG-Graduiertenkolleg "Explorative Analysis and Visualization of Large Information Spaces" of the University Konstanz.

REFERENCES

1. Pfister H., Zwicker M., Van Baar J., Gross M.: Surfels: surface elements as rendering primitives. *SIGGRAPH 2000*, (Jul. 2000), pp. 335-342.
2. Alexa M. et al.: Computing and rendering point set surfaces. *In IEEE Trans. on Computer Graphics and Visualization*, (Jan.-Mar. 2003), Vol. 9, No.1, pp. 3-15.
3. Tangelder J. W. H., Veltkamp R. C.: A survey of content based 3D shape retrieval methods. *Shape Modeling International*, Genova, Italy, (Jun. 2004).
4. Vranic D. V.: 3D model retrieval. *Ph.D. Thesis*, University of Leipzig, 2004.
5. Moenning C., Dodgson N. A.: Fast marching farthest point sampling. *EUROGRAPHICS 2003*, (Sep. 2003).
6. Zwicker M., Pauly M., Knoll O., Gross M.: Pointshop 3D: an interactive system for point-based surface editing. *SIGGRAPH 2002*, (Jul. 2002).
7. Turk G.: Re-tiling polygonal surfaces. *SIGGRAPH 92*, (Jul. 1992).
8. Wand M. et al.: The randomized z-buffer algorithm: interactive rendering of highly complex scenes. *SIGGRAPH 2001*, (Aug. 2001), pp. 361-370.
9. Cohen J., Aliaga D. G., Zhang W.: Hybrid simplification: combining multi-resolution polygon and point rendering. *IEEE Visualization 2001*, (Oct. 2001), pp. 37-44.
10. Collins A., Zomorodian A., Carlsson G., Guibas L.: A barcode shape descriptor for curve point cloud data. *Symposium on Point-Based Graphics 2004*, (Jun. 2004).
11. Dey T. K., Giesen J., Goswami S.: Shape segmentation and matching from noisy point clouds. *Symposium on Point-Based Graphics 2004*, (Jun. 2004).
12. Funkhouser T. et al.: A search engine for 3D models. *ACM Trans. on Graphics*, **22**(1), (2003), pp. 83-105.
13. Hilaga M., Shinagawa Y., Kohmura T., Kunii T. L.: Topology matching for fully automatic similarity estimation of 3D shapes. *SIGGRAPH 2001*, (2001).
14. Chen D.-Y., Ouhyoung M., Tian X. P., Shen Y. T.: On visual similarity based 3D model retrieval. *Computer Graphics Forum*, (2003), pp. 223-232.
15. Shilane P., Min P., Kazhdan M., Funkhouser T.: The Princeton shape benchmark. *Shape Modeling International 2004*, Genova, Italy, (Jun. 2004).
16. Vranic D. V.: Content-based classification of 3D-models by capturing spatial characteristics. http://merkur01.inf.uni-konstanz.de/CCCC/.
17. Schroeder W., Zarge J., Lorensen W.: Decimation of triangle meshes. *SIGGRAPH 92*, (Jul. 1992), pp. 65-70.
18. Vranic D. V., Saupe D., Richter J.: Tools for 3D-object retrieval: Karhunen-Loeve-transform and spherical harmonics. *IEEE 2001 Workshop on Multimedia Signal Processing*, (Oct. 2001), pp. 293-298.