

# Decomposition of Volumetric Models Based on Distance Transform and K-means Clustering

Motofumi T. Suzuki <sup>+</sup>

The Open University of Japan, Japan

**Abstract.** This paper describes a simple decomposition technique for volumetric models. 3D polygonal models are converted to volumetric models, and these models are decomposed by a standard K-means clustering technique. Although the standard K-means clustering technique uses randomly generated initial seeds to determine the starting centroids of the clusters, our technique uses distance maps to determine unique centroids. Preliminary experiments are conducted on a database of 3D models with various shapes. Our method shows better decomposition results compared to random decomposition results, since the technique uses a 3D model's geometrical features.

**Keywords:** K-means, distance transform volumetric data, voxel, decomposition, 3d model.

## 1. Introduction

Decomposition techniques are important, and techniques can process 3D models into components based on some geometric shapes or semantic criteria. Intensive research was conducted on 3D model decomposition techniques, because many applications require these techniques, including partial shape matching [1], local shape retrieval, digital watermarking and salient parts detection. Although much research focuses on techniques for mesh or polygonal model of 3D data, our technique focuses on simple volumetric data. In our experiments, 3D polygonal models are converted to volumetric models, and these models are decomposed by a standard K-means clustering technique. Our technique uses distance maps to determine unique centroids for initial seeds of K-means clustering. A number of initial seeds are controlled by the local maxima detection ratios. These enable the user to determine how many decomposed parts are generated from a 3D model.

In our experiments, some portions of popular benchmark 3D model data sets [2] are used. 3D polygonal models are converted to volumetric models using a software called "binvox," [3] which uses several efficient voxelization algorithms [4]. Voxelizations of 3D polygonal models are time-consuming tasks, especially in regard to obtaining higher resolutions of volumetric data. It takes about 1.5 hours to convert 1804 3D polygonal model data [2] to volumetric data with a resolution of  $64 \times 64 \times 64$ . Once 3D polygonal model data are converted to volumetric data, two important processes are required, including (1) a conversion of volumetric data to graph represented data and (2) a conversion of volumetric models to a distance map created by distance transform algorithms.

## 2. Graph Representations

In the initial process, volumetric data have to be converted to graph representations for computing distances between each voxel in volumetric data. As shown in Fig. 1, each voxel represents a node of the graph, and neighboring voxel relations represent edges of the graph. Fig. 2 shows an example of the volumetric model "rabbit" ( $64 \times 64 \times 64$ ). It is represented in binary values of either zero or one, and there is a

---

<sup>+</sup> Corresponding author. Tel.: +070-5560-5814  
E-mail address: motofumi@ouj.ac.jp

total of 15,399 voxels with value one. Since there are 15,399 voxels with value one out of 262,144 total voxels, about 5.87% of a volumetric cube is filled. When the 3D model is represented by a graph, there are 15,399 nodes and 357,094 edges. Once the model is converted to a graph, it is not difficult to find the shortest path between two voxels. We have used the famous Dijkstra's algorithm to solve the problem of finding the shortest path between two voxels in a graph. It is important to address voxel connectivity by using a graph to find the shortest path for concave models, because simple voxel coordinate-based Euclidean distance computations fail for concave models.

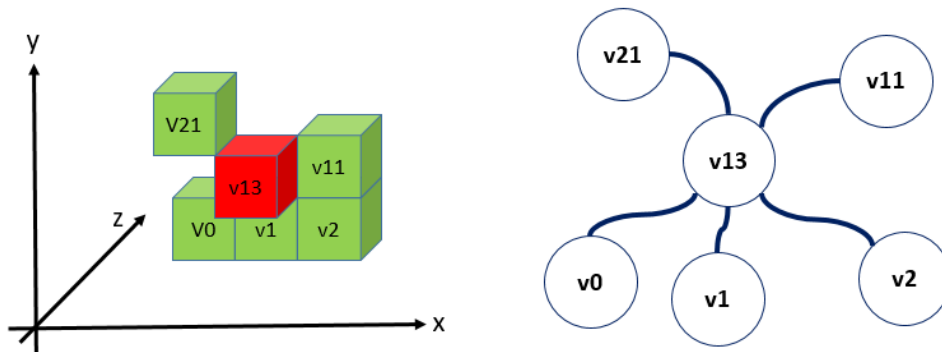


Fig. 1: Graph representation of volumetric data.

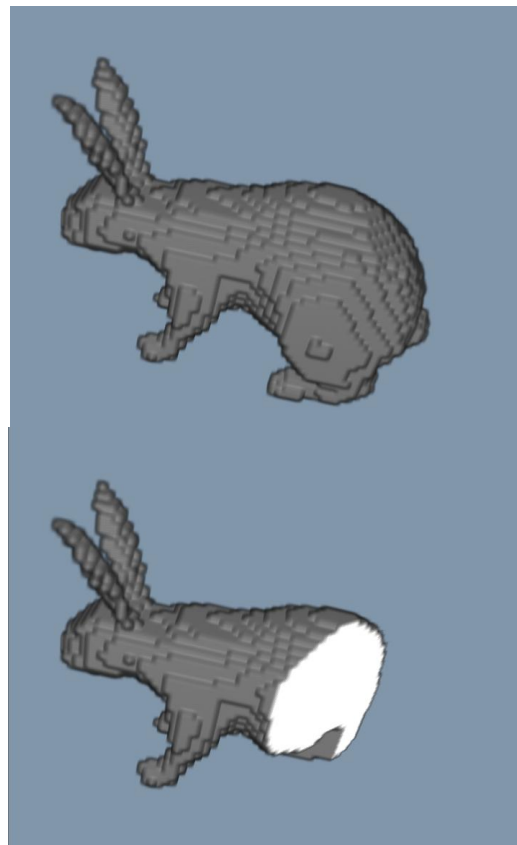


Fig. 2: Volumetric model "rabbit" (64x64x64).

### 3. Distance Transform

Another important process is the conversion of volumetric models into distance maps. A distance map is created by distance transform (DT) algorithms. The distance transform algorithms map each volumetric data voxel into its smallest distance to regions of interest. There are efficient algorithms for creating distance maps for 3D volumetric data. For example, paper [5] presents the Euclidean distance transformation of 3D binary pictures or images. In our experiments, distance transform algorithms are applied to volumetric

models to detect candidates for centroids of K-means clustering. Fig. 3 shows a distance map of a “rabbit” ( $64 \times 64 \times 64$ ). In Figures, dark regions of the distance map correspond to small distances, and bright regions of the distance map can be thought of as hills with the local maxima. Each local maxima in the distance map is a candidate for centroids. Each local maxima is detected by scanning it through the distance map using a  $3 \times 3 \times 3$  filter. Every voxel has 26 adjacent voxels, and the local maxima is computed comparing the value of the center voxel and values of adjacent voxels. If the value of the center voxel is always higher or equal to values of adjacent voxels, the center voxel is a candidate of local maxima. It is possible to control the local maxima detection ratios by changing the number of adjacent voxels in the comparison. If 6 adjacent voxels are used rather than 26 adjacent voxels, more local maxima are detected. A set of local maxima candidates can be sorted based on the voxel value of the distance map. A set of local maxima with a higher value is chosen for the final local maxima set, and redundant local maxima can be removed if the distance between two local maxima is very close. Fig. 4 shows local maxima points in a “rabbit” ( $64 \times 64 \times 64$ ). Once a set of local maxima is detected, it is used for the initial centroids for K-means clustering.



Fig. 3: Duck (volumetric model, clipping plane and distance map).

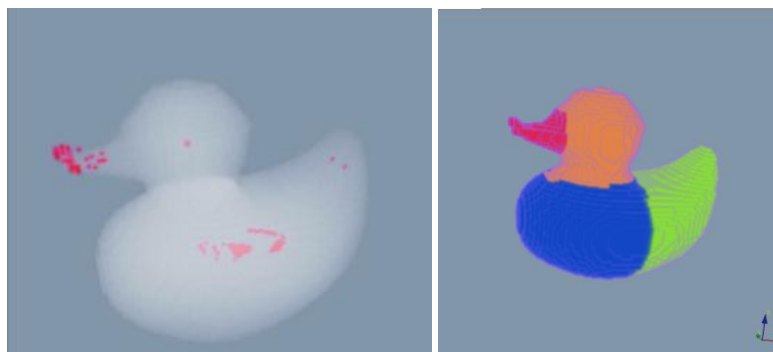
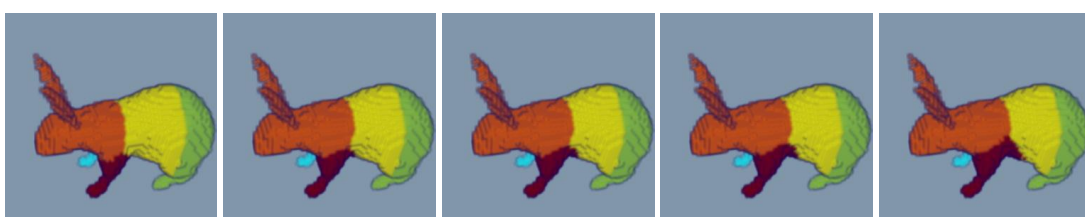


Fig. 4: Local maxima points (220 points) in a volumetric model “Duck” ( $64 \times 64 \times 64$ ).

## 4. Experiments and Results

In the experiment, the convergence properties of the K-means clustering are evaluated. Fig. 5 shows that the K-means algorithm converges to stable centroids of clusters. Both randomly generated initial centroids and our distance-map-based initial centroids are examined. Fig. 6 shows both clustering results for 5 models (duck, fish, human, turtle and snowman). Our method shows meaningful clustering results in terms of 3D model shapes. For instance, each cluster size is balanced in a way that reflects the symmetry and shape of 3D models.



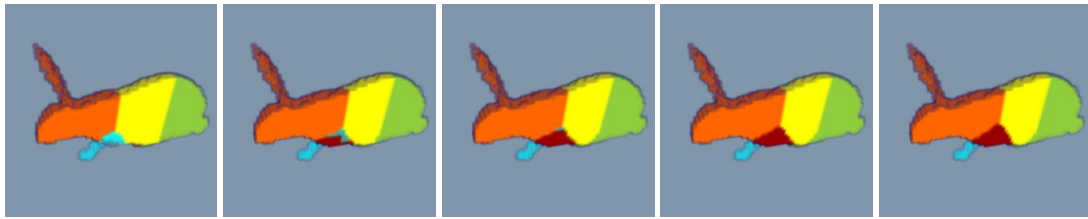
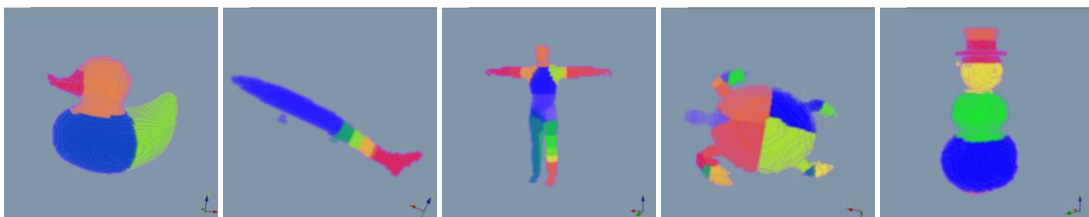


Fig. 5: K-means algorithm converges to stable centroids of clusters.



Randomly generated initial centroids (clusters: 4, 6, 17, 15 and 5)



Distance-map-based initial centroids (clusters: 4, 6, 17, 15 and 5)

Fig. 6: Examples of clustering results (random method and our method).

## 5. Conclusion and Future Work

In this research, we examined a simple decomposition technique for volumetric models. 3D polygonal models are converted to volumetric models, and these models are decomposed by a standard K-means clustering technique. Although there are an infinite number of ways to cluster 3D models if randomly generated initial centroids are used, our simple method determines identical clustering results regardless of the implementation of the software. It is convenient that the system generates identical clustering results by specifying a few parameters. Our method can control a number of decomposing parts for 3D models by controlling the number of local maxima in use. In future works, we will investigate other initial centroid determination algorithms for better decompositions.

## 6. References

- [1] M.T.Suzuki, Y. Yaginuma, Y.Shimizu, A partial shape matching technique for 3D model retrieval systems, Proceeding SIGGRAPH '05 ACM SIGGRAPH 2005 Posters, Article No. 128, 2005.
- [2] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, The Princeton Shape Benchmark, *Shape Modeling International*, Genova, 2004.
- [3] P. Min, "binvox", <http://www.google.com/search?q=binvox>
- [4] F. Nooruddin, G. Turk, Simplification and repair of polygonal models using volumetric techniques, *IEEE Transactions on Visualization and Computer Graphics*, 9 (2), pp.191-205, 2003.
- [5] J. Toriwaki, K. Mori, Distance Transformation and Skeletonization of 3D Pictures and Their Applications to Medical Images Digital and Image Geometry, *Advanced Lectures*, pp.412-428, Springer-Verlag, 2000.