Modeling Very Complex Geometric Assemblies: The Use of Discrete and Hierarchical Models

Pere Brunet[1], Isabel Navazo[2] and Alvar Vinacua[3]

[1]UPC Barcelona, pere@lsi.upc.edu
[2]UPC Barcelona, isabel@lsi.upc.edu
[3]UPC Barcelona, alvar@lsi.upc.edu

ABSTRACT

Virtual Reality (VR) Systems allow immersive inspection of very complex environments, with direct interaction with the objects in the virtual scene. Low cost and affordable VR systems are becoming essential in application areas like Industrial Design, Medical Applications and Cultural Heritage and require sophisticated techniques for model inspection and interaction. Discrete geometric models can be very useful for some involved geometry processing algorithms like model repair, occlusion culling and multi-resolution and level of detail. The interest of these techniques and their potential in present and future VR applications will be discussed. This paper addresses the interest of discrete geometric models in volume-based geometric simplification, the use of discrete tiles for impostor-based extreme simplification, topology-preserving model repair techniques based on discrete membranes and specific algorithms for recovering polygonal shapes from discrete volume models. The use of out-of-core octrees for interactive navigation in very complex geometric assemblies is also presented and discussed.

Keywords: Virtual Reality, discrete geometric models, volume-based simplification, model repair, out-of-core interactive navigation.

1. INTRODUCTION

Virtual Reality (VR) Systems allow immersive inspection of very complex environments, with direct interaction with the objects in the virtual scene. Low cost and affordable VR systems are becoming essential in application areas like Industrial Design, Medical Applications and Cultural Heritage and require sophisticated techniques for model inspection and interaction. Discrete geometric models can be very useful for some involved geometry processing algorithms like model repair, occlusion culling and multi-resolution and level of detail. The interest of these techniques and their potential in present and future VR applications is obvious.

This paper addresses the interest of discrete geometric models in volume-based geometric simplification, the use of discrete tiles for impostor-based extreme simplification, topology-preserving model repair techniques based on discrete membranes and specific algorithms for recovering polygonal shapes from discrete volume models.

After presenting the discrete geometric models and the main concepts in the next Section, two different simplification algorithms based on intermediate volume representations are presented in Section 3. Section 4 is devoted to volume-based model repair algorithms, while Section 5 presents a new algorithm for recovering shape and features from binary volume models. Finally, Section 7 discusses some issues on the interactive navigation problem through gigantic models.

2. DISCRETE GEOMETRIC MODELS

Let *O* be a solid object with one or more connected components. A discrete representation of *O* may be obtained by classifying, against *O*, a set of sample points distributed on the nodes of a regular, axis-aligned three-dimensional grid. Nodes lying inside *O* or on its boundary are labelled as black and nodes lying outside *O* are labelled as white. Such a lattice may be constructed in a variety of ways from a polyhedral or curved representation of *O* through a voxelization process. A similar lattice colour-coding may be produced by considering the values of a scalar field at each node. If the value is larger than a prescribed threshold, the node is black; otherwise, it is white. In what follows, a stick ***s*** is a grid edge connecting a white grid point with a black one.

When this volume model is built from a solid model, it may contain binary information (an in-out classification of every vertex in a grid), or it may consist of a sampling of a scalar field (at the same vertices), for example a signed distance field. More information can be stored, like Hermite data or exact intersection points but these are seldom effectively used in the literature. In the rest of this paper, we will only consider volume models with binary information.

Isosurface extraction algorithms differ on how the discrete information in the grid is generated, on what information does the grid store and the properties of the output surface. Recent algorithms offer different solutions for the disambiguation problem and for controlling the final topology, [2]. Once a choice has been made for every ambiguous cell of the grid, the topology of the resulting isosurface is fixed. This isosurface (that will be named as the implicit isosurface of the volume model in the rest of the paper) stabs all sticks in the grid. Iso-surfaces in Figure 1 correspond to several extraction algorithms and assume that the implicit isosurface stabs all sticks in their central point [8]. Algorithms using improved schemes that result on smoother iso-surfaces will be presented in the next Sections. Figure 1 shows the final isosurface obtained with the alternating tetrahedrization, the tri-linear disambiguation in [9], the dual contouring algorithm in [6] and the topology optimization criteria in [2].

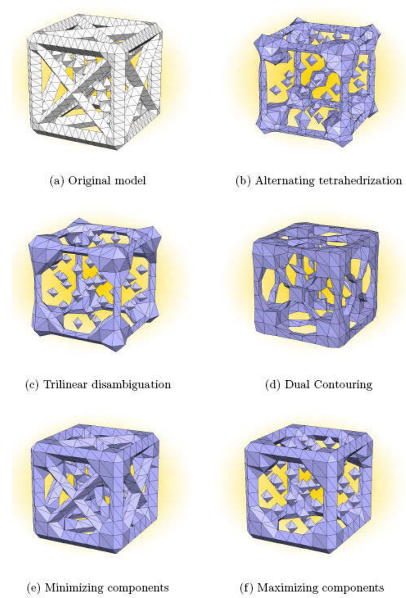

Fig. 1. Result of several iso-surface extraction algorithms.

3. GEOMETRY AND TOPOLOGY SIMPLIFICATION

Discrete volume models are a fundamental tool for geometry and topology simplification. The next two subsections present two different approaches that generate polygonal and impostor-based simplified representations.

3.1 Discrete Geometric Models in Volume-based Geometric Simplification

Simplification in scenes containing a large number of simple objects (Figure 2) can be performed using volume representations. The algorithm in [3] combines appearance preservation and topology reduction by converting 3D models to and from an intermediate octree representation. The input model is converted into an octree, and the final polygonal surface is then extracted from it. This last step involves a Discretized Marching Cubes reconstruction [8] and an iterative edge-collapse constrained to the sticks of the volume representation. The algorithm preserves the appearance and involves an improved surface fitting. Some results are presented in Figure 2: the simplification starts from 4096 shells and 49659 triangles and generates a simplification (Figure 2 – right) with 3 shells and 6947 triangles.

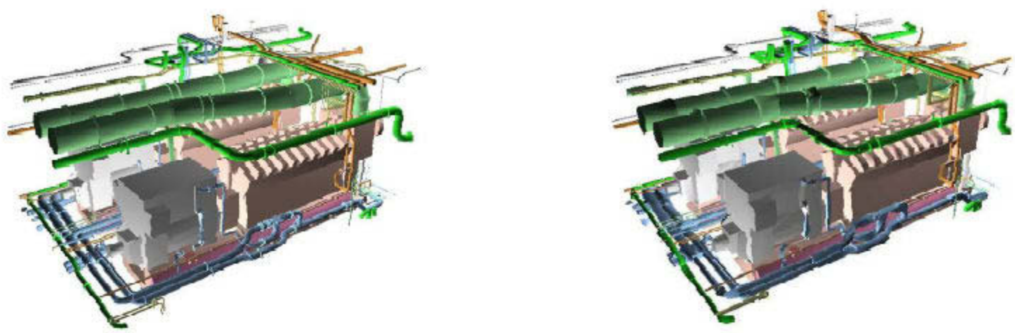

Fig. 2. An assembly with 4096 shells, and its simplification with three shells.

3.2 The use of Discrete Tiles for Impostor-based Extreme Simplification

The problem of finding the maximal planar region for a given geometric model is a very complex one, with many local maxima. Approximate and greedy solutions to this or similar problems can be found in previously published works. But the problem can be reformulated in a discrete way [1], as the problem of finding the plane (or tile) R in a suitable parameterization of planes such that R stabs largest set of sticks. The method in [1] presents an efficient algorithm that computes and guarantees this optimum under a sampling frequency condition. Theoretical conditions on discrete plane parameterizations and the advantages and drawbacks of several plane representations are discussed, and a suitable plane representation with appropriate mathematical properties for this problem has been chosen. The algorithm in [1] uses a pre-computed look-up table (dictionary of planes) that quickly gives the set of planes that stab the neighbourhood of a given stick. The largest tile is found through a voting scheme during a stick traversal. As shown in Figure 3, a reduced number of tiles (14) is sufficient to represent a taxi model by projecting the surface details into textures in the planes of the tiles.

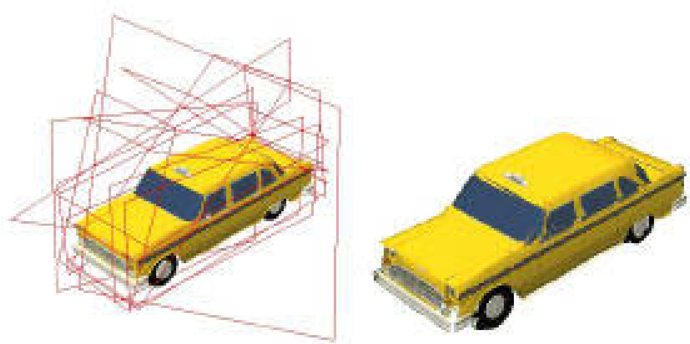

Fig. 3. A taxi model and its 14 largest tiles. At right, these 14 tiles have been used for an impostor-based rendering.

4. MODEL REPAIR

Geometric models coming from scanner devices are in general not valid. They are not closed and they contain a number of holes and cracks, see Figure 4 (left). To repair such a 3d model M we start by immersing it in a voxelization of a suitable resolution *l*. The voxels are labelled according to whether they have parts of the surface S of M inside them (hard voxels) or not (soft voxels). In the next step, a discrete closed membrane is obtained. A discrete membrane [5] is a closed set of 6-connected (face-connected) voxels that contain the surface S. This voxel set is formed by 6-connected hard and soft voxels and divide the remaining voxels in inside and outside. The algorithm works by deforming the membrane like a discrete rubber band: it initially starts as the discrete membrane composed by the voxels of the 6 exterior faces of the Voxelization Universe, and then this membrane is contracted at the locations where soft voxels exist [5], while hard voxels stop the shrinking process. When this membrane cannot be further contracted, the 6-connected discrete membrane has been found. In a final step (Figure [4]) a relaxation of the discrete membrane is performed to obtain a smoother surface. This works by perturbing soft voxels of the discrete membrane in order to reduce the local curvature, [5].

The algorithm does not require to know the topological relations among the initial parts of S or other additional information. The obtained surface approximates and does not stab exactly the initial scanned points. The approximation error has a certain tolerance related to the voxelization size. By pushing the membrane by a shrinking plate of diminishing size [5], the algorithm allows to reconstruct surfaces from initial data points not having a uniform density. Surfaces with *genus* > 1 and/or surfaces with disconnected shells can be reconstructed due to the way of detecting the incursions in the interior of the surface. The algorithm is robust and efficient since it works only with discrete values (voxels).

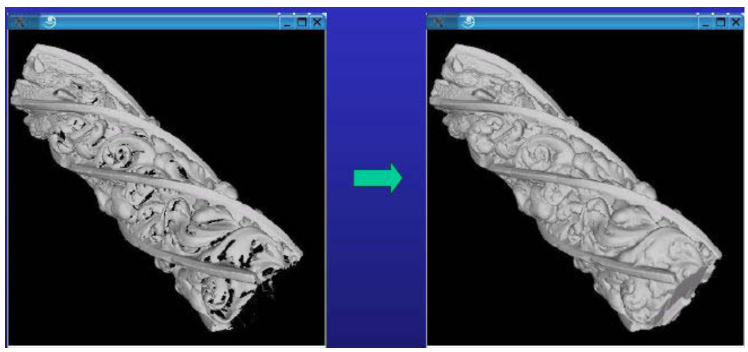

Fig. 4. Repair results on the 3D scanned model of a column.

5. RECOVERING SHAPE FROM BINARY VOLUME MODELS

The Max Tiles algorithm presented in Section 3 can be extended to reconstruct surfaces and features from binary volume models. Previous algorithms were able to recover sharp features from volume models representing scalar fields (like the Extended Marching Cubes in [7]) or from volume models that include Hermite data (like the Dual Contouring algorithm in [6]). This new algorithm (named Pressing from now on) automatically recovers flat regions, curved regions and sharp edges from raw binary voxelizations when scalar field and Hermite data are not available.

In Pressing, a Max Tiles segmentation for recovering flats is combined with a smoothing step on smooth transition surfaces. Pressing uses a new smoothing operator [4] which preserves the connectivity of the initial isosurface (implicitly defined by the binary voxelization, [2]) and optimizes smoothness under a set of stick constraints: the final isosurface is constrained to stab the initial sticks (Figure 5) and it is automatically segmented into flat and curved regions, which may facilitate shape identification, manufacturing and assembly planning

The Pressing algorithm [4] works as follows:

- Sticks are clustered into flat tiles that may be stabbed by a plane [1] (Figure 5-b).
- Junction points are identified. Junction points are vertices between flat and curved regions (Figure 5-b).
- Non-flat regions are faired by an iterative process, which, at each step and for each fresh vertex combines arc-length re-sampling, bi-Laplacian smoothing and snapping. These three operations are performed independently on each X, Y, and Z slice. Their results are combined using a special filter at borders.
- Sharp edges are recovered through the use of a modified Edge Sharpener algorithm (Figure 5-c).

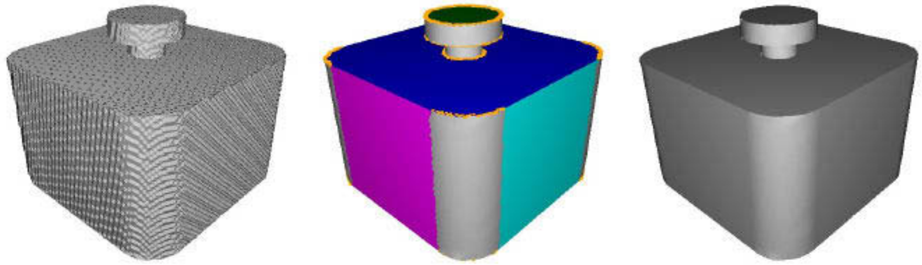

Figure 5: Left: the aliased isosurface was extracted from a 128 x 128 x 128 binary voxelization. Its vertices are at the stick midpoints. Middle: the flats were identified and colour-coded. The junction points along the boundaries between planar and non-planar regions are also identified and shown as orange dots. Note that a flat may be connected to other flats or to smooth faces through sharp edges and to smooth faces through smooth edges. Right: the resulting pressed isosurface.

Results of the Pressing algorithm on a variety of models have been reported and discussed in [4]. Pressing achieves small reconstruction errors and successfully recovers flats and sharp features in a reasonable amount of time. Potential applications include shape recognition, simplification, compression and various reverse engineering and manufacturing problems.

6. INTERACTIVE NAVIGATION

Discrete volume models are also very useful for real-time navigation on gigantic scenes. A number of algorithms based on out-of-core octrees and similar data structures have been proposed. Octrees are well suited for multi-resolution models that must dynamically adapt the level of detail as a function of the camera movements.
Combining an out-of-core octree representation of a gigantic scene with the efficient use of the GPU capabilities and the use of adaptive front algorithms that integrate temporal coherence is fundamental. Using these techniques in the optimal way for the interactive navigation in multi-object scenes is a current topic of research that will certainly lead to the new generation of VR inspection systems.

7. CONCLUSIONS

Discrete geometric models are useful for some geometry processing algorithms like model repair, occlusion culling, multi-resolution and level of detail.

In this paper we have discussed the interest of some of these discrete techniques and their potential in interactive and VR applications. Some algorithms based on discrete geometric models have been presented for volume-based geometric simplification, impostor-based extreme simplification, topology-preserving model repair techniques, recovering polygonal shapes from discrete volume models and for interactive navigation in very complex geometric assemblies.

8. ACKNOWLEDGEMENTS

The authors would like to thank Carlos Andujar, Marta Fairen, Antoni Chica, Ramón Trueba and Jordi Esteve for their ideas and for the help in the implementation of the algorithms presented in this paper.

The research on this project has been partially supported by the CICYT Spanish Agency under Grant TIN-2004-08065-C02-01

9. REFERENCES

[1] Andujar, C., Brunet, P., Chica, A., Navazo, I., Rossignac, J. and Vinacua, A., Computing maximal tiles and application to inpostor-based simplification, *Computer Graphics Forum*, Vol. 23, No. 3, 2004, pp 401-410.

[2] Andujar, C., Brunet, P., Chica, A., Navazo, I., Rossignac, J. and Vinacua, A., Optimizing the topological and combinational complexity of isosurfaces, *Computer-Aided Design*, Vol. 37, No. 8, 2005, pp 847-857.

[3] Andujar, C., Fairen, M., Brunet, P. and Cebollada, V., Simplification of Topologically Complex Assemblies, in *Advances in Multiresolution for Geometric Modelling, Springer-Verlag,* 2005, pp. 339–352, ISBN 3-540-21462-3.

[4] Chica, A., Williams, J., Andujar, C., Brunet, P., Navazo, I., Rossignac, J. and Vinacua, A., Pressing: Smooth Isosurfaces with Flats from Binary Grids, *submitted for publication*, 2006. www.lsi.upc.edu/~moving/publications.xml

[5] Esteve, J., Brunet, P. and Vinacua, A., Approximation of a variable density cloud of points by shrinking a discrete membrane, *Computer Graphics Forum*, Vol. 24, No. 4, 2005, pp 791-808.

[6] Ju, T., Losasso, F., Schaefer, S. and Warren, J., Dual countouring of hermite data, *ACM Transactions on Graphics* Vol. 21, No. 3, 2002, pp 339–346.

[7] Kobbelt, L. P., Botsch, M., Schwanecke, U. and Seidel, H-P., Feature sensitive surface extraction from volume data, *ACM Computer Graphics* 2001 (Siggraph 2001), pp 57–66.

[8] Montani, C., Scateni, R. and Scopigno, R., Discretized marching cubes. *IEEE Visualization'94*, 1994, pp 281-287.

[9] Nielson, G., On marching cubes, *IEEE Transactions on Visualization and Computer Graphics* Vol. 9, No. 3, 2003, pp 283–297.