Discovery of Repeating Feature Patterns in Large 3D Mesh Models

Dinesh Shikhare and S. P. Mudur
National Centre for Software Technology,
Gulmohar Cross Rd. 9, Juhu,
Mumbai 400049, India.
{dinesh|mudur}@ncst.ernet.in

Abstract

Large 3D models documenting architectural designs, heritage monuments, power plants and mechanical CAD designs are being increasingly deployed in various applications involving interactive visualization and Internet based access. More often than not, complexity of these 3D models far exceeds the limits of what be quickly downloaded at popular connection speeds and what be easily stored and rendered for interactive exploration on personal desktops. Engineering models of this type invariably have a number of repeating component features, e.g. windows, pillars, fixtures, etc. We have observed that in most of the widely available models in this class, the geometric description of such component features are also repeated, of course in their positions and orientations.

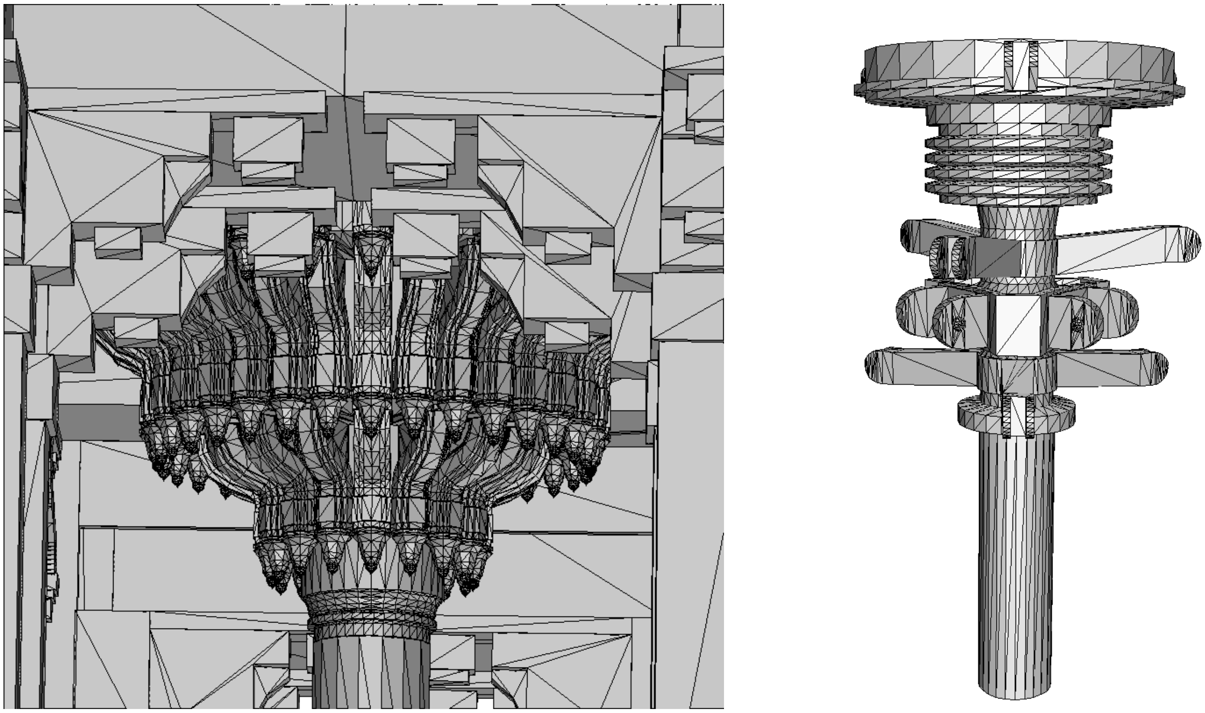
We have developed techniques that discover repetition of component features and also repetition of groups of component features. These techniques have wide possibilities in increasing the efficiency of a number of interactive visualization computational processes, such as geometry healing, simplification, compression, progressive display, etc. As an example, we derive a compression scheme for large 3D models of this kind. We use the above representation to compactly encode the large 3D model using a "master geometry – instance transform" hierarchy. The master geometry itself can be encoded using the most suitable geometry compression algorithms developed earlier [3, 4, 6, 7, 8, 4, 5, 2].

The key challenge in (and also an important contribution of) this approach is to *automatically* identify the component features that repeat in the scene, given the geometric uncertainty inherent in any such large model due to approximations used during modeling and in numerical representations. In models represented by collection of polygon meshes, these features repeat at various levels of granularity (see figure 1):

- *Connected components*, for example, in a machine plant model, nuts, bolts, fasteners, etc. may repeat many times.
- *Groups of connected components*, for example, in an architectural model, many pillars may be found and each pillar is often constructed from multiple connected components.
- *Subsets of connected components*, for example, a gear has many teeth.

While we know some such feature components listed above, there are many other shape features that repeat and we do not know in advance what features to expect in a given model. Given this situation, we must have a set of techniques that enable us to identify the repeating features effectively.

Our technique proceeds by first associating with each vertex in the model a *footprint* that is position and orientation invariant (Examples of some useful invariant footprints can be found in [1]). The vertices of the



(a) (b)

Figure 1: Repeating feature components in the models: (a) Heritage monument, (b) Mechanical CAD model

model are then classified into equivalence classes based on the footprint's value. The vertices in a class represent possible starting points of identical and repeating features. Starting with these vertices in an equivalence class, we grow the feature with simultaneous exploration around these vertices. This growth gives us identical spanning trees about the starting points which represent candidates for repeating features. These candidates are then verified using a geometric matching test. This process of discovery of patterns is carried out on the model until all its vertices are spanned.

For engineering models a good heuristic is to attempt discovery of repeated patterns at the level of connected components. This is done by first obtaining a *normalized orientation* for all the components and then performing a vertex correspondence test among the components that have equal number of vertices, triangles and also have bounding boxes of identical sizes. On finding components that match, the repeating instances (USE-instances) of a shape are represented as a reference to the first instance (DEF-instance) and a suitable geometric transformation to reconstruct the original orientation and position.

At the granularity of groups of component shapes, the repetition is discovered by forming *iso-transformation* collections of the USE-instances. This technique of aggregation of USE-instances based on similar transformations further improves the compactness of the representation.

When applied to the problem of geometry compression, these new techniques are complementary to the previously developed geometry compression algorithms which largely focus on compression of connectivity. In fact, we incorporate those algorithms into our scheme to compress the DEF-instances detected while discovering repeating features. Our compression scheme is somewhat analogous to dictionary based algorithms for compression of text; component shapes and groups of such components take the place of dictionary phrases.

References

[1] Gill Barequet and Micha Sharir. Partial Surface and Volume Matching in Three Dimensions. *IEEE PAMI*, 19(9):929–948, 1997.

[2] A. Guiziec, F. Bossen, G. Taubin, and C. Silva. Efficient Compression of Non-manifold Polygonal Meshes. In *IEEE Visualization 1999*, pages 73–80, 1999.

[3] S. Gumhold and W. Strasser. Real-time Compression of Triangle Mesh Connectivity. In *SIGGRAPH 98*, pages 133–140, 1998.

[4] M. Isenbueg and J. Snoeyink. Face Fixer: Compressing Polygon Meshes with Properties. In *SIGGRAPH 2000*, pages 263–270, 2000.

[5] D. King and J. Rossignac. Connectivity compression irregular quadrilateral meshes. Technical Report GIT-GVU-99-36, GVU Center, Georgia Tech., Atlanta, USA, 1999.

[6] J. Rossignac. Edgebreaker: Connectivity Compression for Triangle Meshes. *IEEE Transactions on Visualization and Computer Graphics*, 5(1):47–61, January-March 1998.

[7] G. Taubin and J. Rossignac. Geometry Compression through Topological Surgery. *ACM Transactions on Graphics*, 17(2):84–115, April 1998.

[8] C. Touma and C. Gotsman. Triangle Mesh Compression. In *Proceeding of Graphics Interface 98*, June 1998.