The Terrain Modeling Using Adaptative Merge Procedures Applied Over Non-Uniform Mesh

FLÁVIO MELLO[1,3], EDILBERTO STRAUSS[2,3], ANTÔNIO OLIVEIRA[3], ALINE GESUALDI[2]

[1]Institute of Research and Development–IPD
fmello@ipd.eb.mil.br
[2]Department of Electronics and Computer Engineering –DEL–POLI–UFRJ
strauss@lcg.ufrj.br,aline@ufrj.br
[3]Computer Graphics Laboratory–LCG–COPPE–UFRJ
oliveira@lcg.ufrj.br

Abstract. The main problem in the walkthrough over terrain models is the rendering techniques complexity. To guarantee a natural and smooth scene sequence, it is necessary to display the actual digital elevation model at interactive frame rates. The original data must be minimized by reducing the number of rendered triangles without compromising the visual quality. This paper describes an optimized algorithm for building a triangular mesh, which combines an efficient regular grid representation with low cost memory requirements.

1 Introduction

The terrain walkthrough plays an important role in computer systems, such as Geographic Information Systems, Military Mission Planning, Flight Simulation. For the terrain elevation representation, a regular grid sampled data known as Digital Elevation Model (DEM), is required [1]. However, the relationship between the DEM resolution and its associated data can easily exceed the capabilities of typical graphical hardware, which restricts a real-time application.

The non-uniform mesh generation method we shall describe in this paper is based on quadtrees structure [2][3]. Our algorithm divides a DEM into a regular grid, and then merges redundant triangles into bigger ones. Our approach solves not only the crack elimination problem, but also the problematic T-junctions without getting into the troublesome Röttger recursion of splitting operations.

2 The Adaptative Merge

The underlying structure of the algorithm is a Röttger quadtree. In this paper, it is assumed that the DEM sides are $2^n \times 2^m$, where n and m might not be equal. First, the DEM is recursively divided into four quadrants until a customized tree height is reached. Since no simplification criteria are made, the resulting tree is a full divided tree. Every tree node corresponds to a square patch of the DEM. The squares represented by the tree leaves correspond to the most refined subdivision, which is a regular and uniform grid of the DEM.

The second step of the algorithm implements the merge of the redundant triangles into bigger ones. It is performed a posorder tree walk, where only the leaves are inserted into a rendering list. The redundant triangles merge occurs during the insertion into the list. At the end of all insertions, the list is composed by triangles with different dimensions. The merge method eliminates the crack problem and also the problematic T-junctions.

Each tree node may merge with a rendering list node if they have two properties. First, the nodes must have a coincident edge, which means that its edges should not only be adjacent but also that the squares represented by the nodes may not merge with other square if they have different side sizes. It is checked if a horizontal merge may occur, and then, if it may occur in the vertical direction. The second property demands that the squares represented by the nodes must be at the same spatial plane, or with the same normal.

It must be observed that the merge of two nodes might result into a new larger node that can also be combined with the others rendering list elements. So, many merges might occur by inserting just one node into the rendering list.

3 Results and Conclusions

The 1024×1024 grid of Figure 1 represents elevation data of Rio de Janeiro City South and its generated mesh. The classical Röttger regular grid uses 150.782 triangles. By applying out merge techniques, it was generated a rendering list with just 137.606 triangles, representing a 8.738% optimization. Critical future issues include view frustum culling and efficient paging mechanism that allows to render DEMs, which do not entirely fit into RAM.

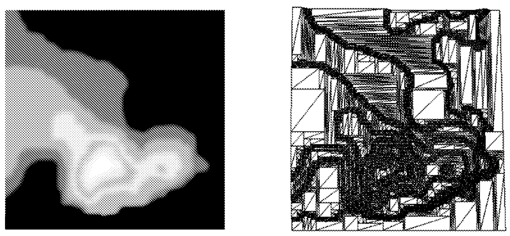

Figure 1: A DEM and the resulting adaptative merge mesh;

References

[1] TURNER, Bryan. *Real-Time Dynamic Level of Detail Terrain Rendering with ROAM*, Gamasutra, 2000.

[2] P. Lindstrom, D. Koller, et al. *Real-time continuous level of detail rendering of height fields*, Computer Graphics, SIGGRAPH '96 Proceedings, pp.109-118 (1996).

[3] S. Röttger, W. Heidrich, et ali. *Real-Time Generation of Continuous Levels of Detail for Height Fields*, (1998).