Compressed Irregular Triangulation Network for Level_of_Detail Visualization

Bisheng Yang

GIS Divison, Department of Geography, University of Zurich-Irchel,
Winterthurerstr. 190, CH-8057 Zurich, Switzerland.
Tel +41 (0)1 635 5256;
Fax +41 (0)1 635 6848;
Email bisheng@geo.unizh.ch

Biography

Post-doctor research fellow of Department of Geography, Zurich-Irchel University; Research interests in Multi-resolution 3D visualization, Spatial data progressive transmission, 3D data model/ reconstruction.

Introduction

Triangulated Irregular Network (TIN) is a popular representation for surface models in GIS, computer graphics and virtual reality (VR) because it has a simple data structure and can easily be rendered by common graphics hardware. High resolution representations of TIN can lead to huge data volumes and long computation times for rendering. Therefore, TIN representation should maintain a good trade-off between computation time, rendering speed and data storage where visualization is a key issue. Level_of_Detail (LOD) models provide an efficient technique for rapid rendering and visualization without losing too much accuracy in surface model representations, and much research has investigated models and algorithms for 3D compression, terrain visualization, model simplification, multi-resolution analysis, progressive transmission, and so on, for example hierarchy triangle (De Floriani 1989), progressive model (Hoppe 1996), triangle decimation (Schroeder 1992), edge collapse/split (Garland et al 1998). In these approaches many methods were proposed based on edge collapse/split for constructing LOD of a TIN (Xia et al 1997, EI-Sana et al 1999). The main difference between them is the encoding of the vertex dependency relationship when visualizing the model. The encoding of the vertex dependency relationship has a big impact on the rendering speed, data storage because it is the preconditions to judge the validity of edge collapse/split, Xia (1997) stores the explicit relationship of the vertex dependencies and EI-Sana (1999) improved the relationship by storing them implicitly. In fact, this approach has unnecessary restrictions (De Floriani et al 2001), and can prevent the further simplification of the original model. The method in this paper extends the method in EI-Sana (EI-Sana et al 1999), and adopts a new method to encode the relationship, which can reduce the data storage and running time. The main contribution in this paper is to develop an efficient algorithm to encode the vertex dependency relationship dynamically during the procedure of LOD construction based on edge collapse/split.

This paper examines existing methods before proposing a new method to extend the original work. The principles of the new method, which is based on general bracketing method (Donaghey 1980), are described and it is shown to be a concise structure for presentation of the relationships between the vertex tree (Park et al. 2001). The method is introduced as a way of constructing and encoding the vertex dependency relationships during the processing of the LOD model. Detailed information on the construction of the vertex tree based on the bracketing method and the encoding of these relationships is presented. Experiments to test algorithm efficiency are presented and compared with previous methods.

Data Structure

The compression algorithm is based on edge collapse/split method, the outline of the edge collapse/split is illustrated in Figure 1.

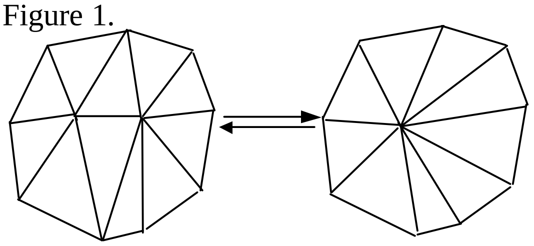

Figure 1. Outline of edge collapse/split

According to the analysis of above, the vertex dependency relationship has a big impact on the data storage and running time of algorithm. In order to efficiently encode and storage the vertex relationship dynamically, the data structure is proposed based on bracketing method (Donaghey 1980). In accordance with bracketing method, one vertex can be represented as left bracket (0) and right bracket (1), so the relationship among the vertexes in a tree can be encoded with 2 bytes per vertex. Figure 2 illustrates the encoding results with bracketing method.

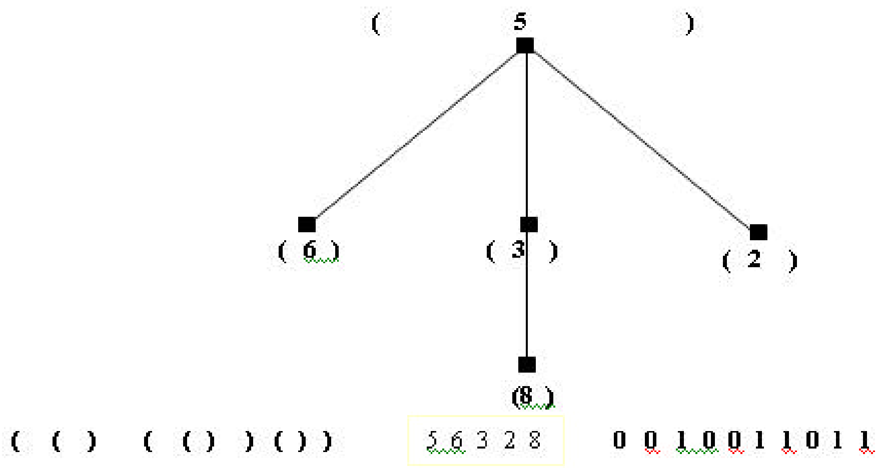

Figure 2. Encode vertex relationship with bracketing method

Whenever the vertex collapse/split, the relationship between one vertex (parent) with two sub-vertex (children) will emerge, the relationship among the parent and children can be encoded by bracketing method with 2 bytes per vertex. According to the edge split/collapse, the LOD model of the surface model at one certain running time can be represented in figure 3. The relationship among the vertexes (the nodes in the tree) composes the vertex hierarchy tree.

In order to encode the vertex relationship among the tree, a virtual vertex is inserted as the root of the tree and, the relationships can be represented with bracketing method. When construct LOD models of the surface model, one of important factors is to encode the right vertex dependency relationship in order to maintain the correctness of the LOD model. A technique for encoding the vertex dependency relationships will be proposed in the next section.

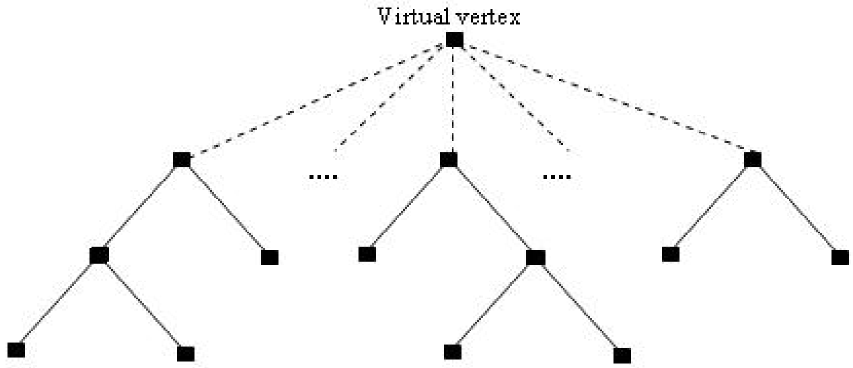

Figure 3 the vertex hierarchy relationship

Implementation

The encoding of the vertex dependency relationship and judging the validity of vertex collapse/split is of vital importance to the correctness of results. According to the analysis above, the relationship of vertex dependency consists of a hierarchy tree. From figure 3, it is easy to see the balance of the tree affects the performance the algorithm, it affects the traverse time when search vertex in a tree, so when constructing the LOD model, as many as possible vertexes should be generated at each layer of the tree in order to keep good balance. Another important factor is to define the regulations about validity of vertex collapse/split. In order to explain the principles clearly, several definitions are defined as follow.

Vertex link relationship, the vertex link relationship of one vertex (a) is a vertex set, each vertex in the set and the vertex (a) consists of one edge in the LOD model.

Affected region of vertex (a), it is constructed by vertex link relationship of the vertex (a).

The detailed description about how to judge the validity of the vertex dependency relationship is proposed in this section.

Experimental analysis

The algorithm has been implemented in C++ and visualization performed with the OpenGL graphical library, and several datasets have been selected to test the validity, efficiency, and data storage of the algorithm. Several experimental results are illustrated in this paper.

Conclusion

The method proposed in this paper can be used to construct the LOD model of the surface model based TIN efficiently in real time. The vertex dependency relationship can be encoded according to the bracketing method. Compared with other methods of encoding vertex dependency relationship, the method based on bracketing method in this paper has lower volumes of data, and it does not change the topological relationship of original model, moreover, the method can be extended to create view-dependent LOD of surface models easily. It is an efficient method for the improvement of visualization speed and the reduction of data storage of high resolution surface models.

References

Donaghey, R., (1980) Automorphism on Ctalan trees and bracketing. Journal of combinatorial theory. B 28, 75-90.

De Floriani, L., (1989) A pyramidal data structure for triangle-based surface description. IEEE computer graphics and application 8(2), 67-78.

De Floriani, L., Magillo, P., (2000) Compressing triangulated irregular networks, GeoInformatic, Volume 4 ,Issue 1, 67-88.

De Floriani, L., Magillo, P., (2001) Multiresolution Meshes, Principles of Multiresolution in Geometric Modeling - PRIMUS01 summer school, Munich, August 22-30, 2001, 193-234.

EI-Sana, J., Varshney, A., (1999) Generalized view-dependent simplification. Computer graphics forum, Vol.18 ,No.3, 84-94

Garland, M., Heckbert,P., (1998) Surface simplification using quadric error metrics, In SIGGRAPH’97, 209-216.

Hoppe, H., (1996) Progressive meshes, In: SIGGRAPH’96 Proceeding, 99-108

Park, D., Cho, H., Kim, Y., (2001) A TIN compression method using Delaunay triangulation. Int.J. geographical information science, Vol 15(3), 255-270.

Schroeder, W, Jonathan, A.Z., (1992) Decimation of triangle meshes. Computer Graphics (SIGGRAPH’92 Proceeding). 26(2):65-70

Xia, J., El-Sana J, et al , (1997) Adaptive real-time level-of-detail-based rendering for polygonal model. IEEE Transactions on Visualization and computer graphics.Vol.3 No.2 171-183