Survey on Simplification Algorithm for Large-Scale Terrain Modeling

WU HuiXin, ZHENG Hui
Dept. of Information Engineering
North China University of Water Conservancy & Electric Power
Zhengzhou, China
wuhuixin2001@tom.com, hnzh201@ncwu.edu.cn

Abstract—With the increasing demand for the application of future simulation technology and virtual reality, the virtual environment modeling techniques will become increasingly important. Three dimensional terrain is one of important factors of the natural scenes in virtual environment and it is also a significant composition of view system in virtual simulation field. Firstly, this paper describes the research content and research objectives of the large-scale terrain modeling. Secondly, it systematically analyses research methods and techniques of terrain modeling simplification algorithms in recent years, including static and dynamic simplification algorithm, and compares the advantages and disadvantages of each method. Finally, future research directions are discussed.

Keywords- 3D terrain; Model simplification; LOD; Spatial Data Model

I. INTRODUCTION

Real-time terrain rendering for large-scale terrain has been widely used in geographic information systems, visual simulation and virtual reality. In recent years, it has become the research focus in the related fields. One of the core issues of terrain visualization is to solve the contradiction between complex terrain surface model formed by massive terrain data and the limited computer graphics hardware rendering capabilities. For example, when roaming terrain simulation, we must pass through a width of 10 km, 100 km long strip of terrain. Assuming the size of the basic terrain grid size is 10m x 10m, if we want to handle all the terrain data, about 10^7 grid nodes need to be recorded and 2 x 10^7 triangular patches have to be rendered. These complicated computing makes a very high demand for computer storage capacity, processing speed, rendering speed and so on. Figure 1 shows a grid mesh of 4097 x 2047 vertices containing both color and elevation data [4].

![Figure 1. A terrain grid with both color and elevation data](image)

However, in many cases, high-resolution model is not always necessary and compromise needs to be made between the accuracy of the model and processing time. Therefore, we must use some simplified model to replace initial model. That is terrain model simplification. Model simplification refers to use the appropriate algorithms to reduce the number of triangular patches, edges and nodes on the premise of keeping the topological structure of initial model unchanged. Early in 1970s, scholars discuss the problem of terrain grid model simplification, but the terrain grid simplification technology did not have in-depth research until 1990s. There are a variety of categories of mesh simplification algorithm. For example, according to whether the topological structure can be maintained, it can be divided to maintain shape and non-topological to maintain shape [3]. According to the model simplification process it can be divided into stepwise refinement and geometric simplification. According to the controllability of error, it can be divided into constrained error and non-constrained error. Based on Viewpoint correlation, it can be divided into view-dependent and view-independent [1, 2].

II. RESEARCH CONTENT AND RESEARCH OBJECTIVES

Present research on terrain model simplification techniques mainly focus on the following areas:

(1) View frustum culling

View frustum culling means before rendering the scene image, removing the polygon model out of the view frustum first and when rendering scene image, transporting the polygon model within the view frustum to render pipeline. This approach will not have bad effect on correct image rendering result, but it greatly reduces the number of polygons processed by render pipeline, which can improve real-time rendering speed effectively.

(2) Back face culling

Generally, polygons used to describe the terrain have front face and back face. When rendering the image, we usually check whether the polygon is back towards the view point based on sight direction and normal vector. If so, the back face of this polygon is visible to the view point, and effective front is not visible. Therefore, removing the type of polygon can reduce the number of polygons processed by render pipeline and improve real-time rendering speed effectively.

(3) Visibility culling

If some polygons are shielded by other polygons in the image rendering, these polygons should not appear in the final processing results. As a result, removing the type of polygon can improve real-time rendering speed. Although
most of the graphics acceleration hardware support for polygon depth test (z-buffer) technology, however, if the number of polygons drawn is too large, it will still affect the overall rendering speed. Therefore, visibility culling has good effect on improving the rendering speed.

(4) Level of Detail (LOD)

Level of detail refers to a set of model described in different detail for the same scene, which can be chosen to use for drawing scene. The basic idea is to use different level of detail to construct terrain. When roaming terrain, we hope to achieve space continuity and temporal continuity. That is, the terrain should be continuous with no gaps or holes, and the terrain scene should change synchronously with the moving of viewpoint without delay.

III. RESEARCH PROGRESS OF TERRAIN SIMPLIFICATION ALGORITHM

To achieve space continuity and temporal continuity when roaming terrain, researchers proposed a number of related algorithms. Through the comparative analysis of these algorithms, we divide them into static (view-independent) and dynamic (view-dependent) simplification algorithms.

A. Static (view-independent) simplification algorithms

Most of the early simplification algorithms are static which consider the information of model itself and have nothing to do with the viewpoint. Static simplification can construct level of detail model, but it should store many approximate model of different resolution, which takes up more storage space. What’s more, as the resolution between adjacent two levels is inconsistent, it will lead to a sense of image popping. Static simplification mainly includes the following algorithms.

(1) Vertex clustering

The model reduction based on vertex clustering of multiple levels of detail can be viewed as re-sampling from the vertex on geometric model. If the projection of many adjacent vertexes in three dimensional space surround approximately the same pixel, these vertexes can be replaced by one representative vertex, which not only can reduce complexity of the geometry model, but also the quality of the output image will not be greatly affected. Vertex clustering method first divides square boundary (bounding box) into several zones through uniform spatial subdivision. Then in each region chooses representative vertexes to replace other vertexes. Finally we can obtain the geometric model of lower complexity by re-triangulation of the representative vertexes. Kok-Lim Low and Tiow-Seng Tan improved the vertex clustering algorithm by defining a better important level under maintaining the general nature of the original algorithm. Kun Zhou used octree to classify bounding box adaptively and adopted the second error measure to control the generation of new vertex. Peter Lindstrom expanded the vertex clustering method, and applied it to super large-scale grid model reduction[7].

(2) Edge Collapse

In edge collapse simplification algorithm, edge is regarded as basic geometric element to be deleted in each simplification operation. Facets can be reduced to arbitrary levels of detail by carrying out repeated selective edge collapse. Vertex split is the inverse transformation of edge collapse, which can be used to restore lost information. Hoppe build a progressive mesh (PM) with edge collapse and vertex split and get a multi-resolution level of detail model. The key of edge collapse is the sequence of edge collapse and the position of new vertex. Hoppe determined the sequence of edge collapse and the position of new vertex with energy optimization in 1993. Computational complexity of energy optimization is high and more computing time is necessary, but the effect of the generated model is best in compared with other simplification algorithms. Garland and Heckbert proposed a simplification algorithm based on quadric error metric (QEM) in 1997[7]. The error metric of QEM algorithm is based on the sum of squared distance from point to plane. This algorithm is fast and the quality of generated model is only second to energy optimization algorithm. Hoppe added vector, color and texture information to the QEM algorithm, and then used winged-edge data structure to realize it, which attained good results. Lindstrom and Turk also got a similar mathematical expression with QEM, using the changes of volume and area after simplification as error metric[8]. This method only need model surface connection information and the position of vertexes when computing collapse sequence and new vertex position. Therefore, small amount of memory for this algorithm is enough and computing speed is high.

The advantage of edge collapse is a inverse operation can be obtained each edge collapse operation—vertex split. So orderly recording the steps of edge collapse and vertex split, we can subitize the geometry model by vertex split and coarsen the geometry model by edge collapse. Because rough and refined data is too trivial, the conversion speed of LOD becomes slower, which does not apply to large-scale real-time terrain rendering.

(3) Region merging

Kalvin and Taylor proposed a typical region merging algorithm which chooses a triangular face as the seed, and according to certain criteria combines the surrounding surface into a larger surface (super surface)[9]. The users can use this algorithm to control the error of simplification model by themselves and the topological structure of model can be maintained by the algorithm. But due to unavoidable surface with holes, the efficiency of the algorithm is low. Jie Li and Ze-Sheng TANG improved the algorithm with super-surface region segmentation algorithm to eliminate the self-intersection of projection profile perimeter, which avoids generation of super surface with holes. Hinker and Hansen first identified the triangles with parallel normal vector to form a surface group and carried out region merging by re-triangulation. Jonathan Cohen put forward a framework of simplification envelopes which can ensure simplification operation within the context of global error. It does not change the topological structure of initial model while the sharp features can be maintained.

As a typical application, Yuguo WU proposed a multi-scale terrain mode real-time generation algorithm[10]. The first step, the algorithm builds n levels terrain TIN model.
under each comparing rule and sets different resolution for each level based on the existing DEM data. Each comparing rule is called primary-level and each level under primary-level is called sub-level. The second step, it creates a full model-level series and records all the primary-levels and sub-levels which may be used. The third step, it creates an existing model-level series, records the existing main level and adds n sub-levels to each primary-level. The last step is to set the resolution evaluation function in order to choose the appropriate terrain model resolution based on view position. The main problem of this algorithm is how to relieve the popping effect in the process of rendering terrain model.

B. Dynamic (view-dependent) simplification algorithms

The basic idea of dynamic simplification algorithms is that we can get the approximate model with suitable resolution dynamically. The level of model simplifying depends on the factors outside model itself, such as viewpoint. Dynamic simplification is generally achieved by a number of simple local geometric transformations which is extension of static simplification algorithms.

(1) Simplification algorithm based on viewpoint

In 1996, Julie C. Xia presents a real-time simplification algorithm based on viewpoint which can choose different levels of resolution in different areas of the same model. Based on surface normal of view frustum and the screen space geometric error, Hoppe defined a refined standard which includes the principle of view frustum, front direction and screen space geometric error. With the principle for edge collapse and vertex split, he constructed multi-resolution model. David Luebke used octree for space division. If the projection value of spatial octree node on the resolution model. David Luebke used octree for space edge collapse and vertex split, he constructed multi-model of multiple levels of detail. The algorithm generates several key simplified models in advance and ranks the view important degree of triangles. In real-time rendering, we select a higher resolution than necessary approximation model and delete triangles based on important degree until meet the current level of detail. The algorithm can generate a series of consecutive simplified mode of the original grid. The PM algorithm records the position of initial and new vertex and the changed information of connectivity relationship among vertexes. In this way we can get a PM expression pattern composed by a series of simplified information.

PM expression defines a grid sequence whose precision increases gradually \( (M^0, M^1, \ldots, M^N) \). \( M^0 \) is the coarsest grid and \( M^N \) is the most precision grid. In the PM representation, an arbitrary mesh \( M \) is simplified through a sequence of \( n \) edge collapse transformations to yield a much simpler base mesh \( M^0 \).

\[
(M = M^N) \xrightarrow{vsplit_{n-1}} \ldots \xrightarrow{vsplit_1} M^1 \xrightarrow{vsplit_0} M^0
\]

Because each ecol has an inverse, called a vertex split transformation, the process can be reversed:

\[
M^0 \xrightarrow{vsplit_n} M^1 \xrightarrow{vsplit_{n-1}} \ldots \xrightarrow{vsplit_1} M^N \xrightarrow{vsplit_0} (M^N = M)
\]

In the vertex split operation, the relation of father and child nodes form vertexes levels of PM (as shown in figure 2). The vertexes included in level \( M^0 \) are root nodes and leaf nodes are the initial mesh \( M \). In real-time rendering, the details of model being deleted can be restored gradually through vertex split operation. In this way we can get simplified model with continuous resolution which makes the LOD transition smooth. However, some major issues are left unsolved. The refine function is not designed for real-time performance, and fails to measure screen-space geometric error. More importantly, no facility is provided for efficiently adapting the selectively refined mesh as the view parameters change. To solve the ambiguous of PM, Zhiliang Tao proposed reversible progressive mesh supporting fast restoration. Guangzheng FEI put forward a new editable multi-level model with PM.

(2) Level representation method

Isler proposed a hybrid method to achieve semi-real-time level of detail. The algorithm generates several key simplified models in advance and ranks the view important degree of triangles. In real-time rendering, we select a higher resolution than necessary approximation model and delete triangles based on important degree until meet the current precision. R. Ronfard first defined geometric deviation between the simplified grid model and initial grid model caused by edge deletion and sorted it in accordance with small to large. Then according to the complexity of model user needs, the edge delete operations were carried out. The algorithm can generate a series of consecutive simplified mode of the original grid.

(3) Progressive mesh method

The most famous Simplication in the dynamic algorithm should be PM algorithm proposed by Hoppe. With the vertex split and edge collapse as its basic operations, The

\[
\text{Figure 2. Vertex hierarchy in PM}
\]
refining coarse mesh gradually. This method defined progressive transmission and coding suitable for network application and provided an interactive selection method for any approximate mesh quality[16].

During the flyover of large-scale terrain, in order to eliminate the popping effect of switching among levels of detail and to increase the frame rates with high image quality, Huixin WU and Xianfeng LI proposed a new bottom-up modeling strategy which constructs simplified terrain triangle mesh globally and updates mesh nodes dynamically[18]. Hybrid culling technique based on blocks and triangle faces and simplified computing method for screen-space errors were employed to select appropriate terrain nodes rapidly. Then the Delaunay terrain mesh was updated by adding nodes, deleting nodes and modifying locally. At the same time self-adaptive control for screen-space error tolerance was achieved during the terrain flyover. Results of simulation experiments demonstrated that the algorithm eliminated popping effect effectively, and had a higher frame rate compared with other algorithms. So it was particularly suitable for close-distance flyover simulation of large-scale terrain.

IV. RESEARCH TREND OF TERRAIN SIMPLIFICATION

To sum up, the main problems in the process of large-scale terrain rendering are the popping effect of switching among levels of detail and the low frame rates. Future research work will mainly focus on view frustum clipping, back face culling, visibility culling and level of details and so on to achieve real-time roaming simulation of real-time three-dimensional scene. At present, the multi-resolution model has been an important development trend of simplification technology. However, the existing simplification criterions are mostly single. That is, the criterions are based on either view point parameters or terrain surface characteristics. If the two can be combined, the data volume will be further reduced[10].

With the rapid development of computer graphics hardware and software technology, landscape simulation technology based on fractal appeared in recent years. The fractal modeling technology has developed from satisfying visual experience only to actual landscape simulation. How to generate the 3D terrain with good visibility and certain geometric precision is an important development direction of fractal scene simulation. If we can map the satellite remote sensing images to 3D terrain surface in the form of texture, then the terrain possesses not only 3D geometric information, but also rich surface features and cultural information. This type of fractal landscape will have more extensive application prospects.

V. CONCLUSIONS

With the rising demand for future simulation technology and virtual reality, the virtual environment (VE) modeling with natural visual effect will become increasingly important. 3D terrain is an indispensable factor in the Virtual natural environment and is also an important part of Vision System in virtual simulation. This paper systematically analyses static and dynamic simplification algorithm and compares their advantages and disadvantages. Finally some new research trends of terrain simplification have been discussed tersely.

REFERENCES