Dynamic Data Retrieval and Distance Decay of Triangulated Irregular Network(TIN) in Three Dimensional Visualizations

Wenji Zhao[1], Tao Tang[2], Huili Gong[1], Fushou Duan[1], Ying Mo[1]

[1]Beijing Municipal GIS and Remote Sensing Key Laboratory College of Natural Resources, Environment, and Tourism Capital Normal University, Beijing, China 100037
E-mail: TANGT@BuffaloState.edu
[2]Department of Geography and Planning State University of New York-College at Buffalo 1300 Elmwood Ave., Buffalo, NY, USA 14222-1095

Abstract

This study presents two computational methods in improvement of large quantity data retrieval and display in 3D GIS virtual reality visualizations. The first method is dynamic data retrieval using quad-tree database structure and level of detail (LOD) algorism. The second is to dynamic simplify TIN mesh during a virtual travel. The streamlined data display was achieved combining partial data retrieval of original quad-tree according LOD parameters, and dynamic erase of nodes not necessary in the TIN as observational scene changes. The results indicated that the computational algorisms drastically increased retrieval and display speeds without compromise the quality of the visualizations.

Keywords

3D virtual reality, dynamic data retrieval, distance decay of TIN mesh

I. INTRODUCTION

Three dimensional (3D) visualizations in GIS have been widely applied in land use management, terrain based fly over or drive through simulations, and urban planning. 3D virtual reality projected in GIS provides new opportunities to users for viewing the planned or natural hazard induced landscape changes. Currently, the triangulated irregular network (TIN) is one of the major models to represent 3D terrains or human made buildings. However, retrieval of large quantity of digital data and project them on a computer screen or a virtual reality device in a real time pace is always a great challenge. This is mainly owing to the limitations of computational storage space and the CPU speed. Milne and Sear (1997) modeled the river channel topography using TIN and grid data structure. Abdelguerfi, et al. (1998) compared different hierarchical TIN techniques in 3D representations for accuracy and aesthetical appearance. Verbree, et al. (1999) initiated the approach of incorporating GIS enabled 3D models into virtual reality. Speckmann and Snoeyink (2001) discussed the technique of triangle strips in speeding up the display of TIN models. Zhu et al. (2001) presented a method in converting contour lines from topographic maps to 3D TIN models. The algorithms provided users of means in extracting TINs from contour lines. Park, et al. (2001) introduced a Delaunay compression method of TIN model in progressive 3D visualizations. The research showed an average of 0.17 bits reduction per node of irregular triangles. Integrations of 3D GIS, virtual reality, and internet disseminations of the 3D scene were also studied recently (Kraak and Maceachren, 1999; Rhyne, 1999; and Huang, et al. 2001). Masumoto et al. (2004) developed an algorism of 3D visualization of terrain and geological strata using an open source GIS–GRASS. Kumsap et al. (2005) presented the distance decay visibility method for forest landscape visualizations. The research presented the vegetation cover in a 3D scene with level of detail (LOD) technology. As one moves away from the forested area, the 3D view was gradually simplified. A dynamic edge collapse and vertex split method of TIN was also presented (Yang, et al., 2005). The study indicated that the time cost of 3D visualization is reduced by multi resolution TIN mesh display.

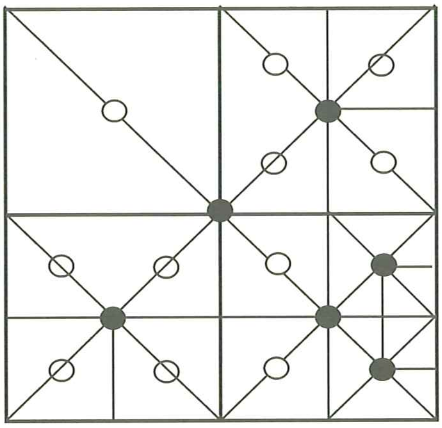
This paper presents two new methods in GIS 3D real time visualizations to facilitate effective storage and quick retrieval of large quantity of digital data. The first is a quad-tree structure based dynamic level of detail (LOD) algorithm; and the second is a dynamic simplification method of TINs based on observational viewpoint and view-axis. This research drastically decreased retrieval and display time in a 3D visualization process. It optimizes the computing storage space and display speed in GIS enabled 3D visualization and virtual reality.

II. METHODS

A. Quadtree data storage and retrieval

Structure of data storage directly influences the retrieval efficiency. Previously most of 3D visualization data was stored in a binary tree structure of one dimensional array. In this study, we used quad-tree structure of two dimensional arrays to store the data for 3D real time display. Two two-dimensional arrays were constructed in the database corresponding to quad-tree structure applied in the data storage. One array is used to store the elevation of 3D terrain or human made landscape in the scene according to row and column of

1082-4006/06/1201-21$5.00



(a) Quad-tree node array

?	?	?	?	?	?	?	?	?
?	?	?	?	?	0	?	0	?
?	?	0	?	?	?	1	?	?
?	?	?	?	?	0	?	0	?
?	?	?	?	1	?	?	?	?
?	0	?	0	?	0	?	1	?
?	?	1	?	?	?	1	?	?
?	0	?	0	?	0	?	1	?
?	?	?	?	?	?	?	?	?

(b) Quad-tree node

Figure 1. The sequential storage arrays of quad-tree

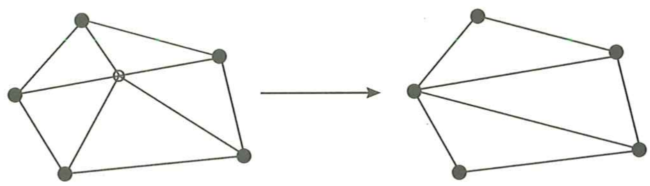
displaying area (Figure 1(a)). The other is an array to store status of the quad-tree nodes (Figure 1(b)). If a quad-tree node needs to be divided further in a data retrieval event, value 1 is returned. Return value 0 if the node does not need to be divided further. In the same time, "indefinite" value that is represented as a question mark is returned if the data of a particular grid cell has not been read (Figure 1(a) and 1(b)).

Quad-tree data structure for 3D visualization not only makes the data organization more efficient, but also provides flexibility of data retrieval in 3D virtual reality display. In order to enhance the retrieval efficiency, elevations of every node and its four subsidiary nodes as well as the pointers to the subsidiary nodes are stored with the node. X, Y coordinates for each of the nodes and other attribute information are stored in a look-up table that can be computed during the process of traverse.

In this study, level of detail (LOD) algorithm was applied to display real time virtual reality. The original quad-tree structure database was traversed again and again for displaying of 3D scenes. Each time, only part of the quadtree was retrieved according the parameter yielded from LOD. Using this method, coarse sampling of original data was conducted when displaying the entire 3D scene of large dataset. As the user moves into (or zoom into) a particular part of the entire 3D scene, detail sampling of that part in the database was conducted and displayed. The partial quad-tree retrieval method optimizes the 3D visualization speed and streamlines the 3D virtual reality display. New 3D scenes produced by partial quad-tree are continuously formed, and the used partial quad-tree duplications are constantly deleted in the computing memory.

B. Dynamic simplifications of TIN based on observation point and observation axis

Difficulties of slow processing or jumping displays from one scene to another were encountered using dynamic LOD model in the TIN enabled 3D virtual reality. This is owing to the intensive drawings of large quantity triangles in the displayed TIN mesh. Currently, the widely used method is to replace the entire TIN mesh as the resolution of display changes. A method of dynamic simplification of TIN was developed in this research to streamline the virtual reality display. The algorism of simplification is the node erase of the triangles and reorganization of the TIN mesh according to the requirement of level of detail (LOD). In essence, this process is a many to many simplification in comparison to that of one to many simplification of regular coordinate grid mesh as the map scale changes from large to small (Figure 2). The TIN simplification process ultimately replaces a detail oriented TIN mesh by a coarse TIN mesh gradually other than through large quantity of data erase and retrieval.



(a) TIN mesh before node erase (b) TIN mesh after node erase

Figure 2. Dynamic node erase process of TIN mesh simplifications

The steps of algorism of dynamic node erase are as follows:
1. Computing the focal observation point at a particular user's observation point and observation axis.
Where: observation point is the pint that observer's eyes are located in the 3D virtual reality observations; observation axis is the axis that depicts the observer's eye sight direction, it is the imaginary line that connects observation point to focal observation point; focal observation point is the cross point of observation axis to the 3D virtual reality observation surface.
2. Computing the significant parameters of each node on the TIN mesh according to their observation distance and angle that is related to observer's observation point, axis, and focal point.
3. Relating significant parameters to LOD requirement in determining which irregular triangle node would be erased.

4. Traversing all the nodes and erasing those disqualified nodes in step 3. In the meantime, reorganizing irregular triangles in the polygons after node erase.
5.Repeat step 4 until all of the irregular triangle nodes that disqualified by the significant parameter of the observation point and axis are erased, and a new TIN mesh is completed.

The format of database structure of this process is shown as follows:

```
////////////// TIN Irregular triangle node structure
struct TIN Point
{
unsigned long ID;          //Node ID
double x;                  //Easting coordinate of node
double y;                  //Northing coordinate of node
float z;                   //Node elevation (or altitude)
bool IsDel;    //Erase ID false: do not erase; true: erase
bool IsAdd;    //Add ID false do not add; true add
float Sration;  //Area projection ratio
unsigned long NearStartPoints[16]; //Adjacent nodes group
unsigned long NearEndPoints[16]; //Adjacent nodes group
                                   after erase
unsigned long NearStartTriangles[16]; //Start adjacent
                                   triangle group
unsigned long NearEndTriangles[16]; //End adjacent
                                   triangle group
}.
////////////////TIN triangle structure
struct TIN Triangle
{
unsigned long ID;          //Triangle ID
unsigned long TrianglePoints[3]; //Node IDs of the triangle
bool IsShow;               //Triangle display ID
};
```

Adjacent irregular triangles and nodes of an erasing node can be easily obtained using the above database structure. Two groups of adjacent triangles and nodes were identified in the TIN mesh node data structure. The start group represents the adjacent triangles and nodes before the nodes are examined for erase, and that of the end group represents the adjacent triangles and nodes in the process of dynamic simplification after node erase examination. The reason of this design is that if one node is erased, it must change the topological relationship of the adjacent triangles and nodes. If an adjacent node of the erased node also needs to be erased in the simplification process, the erasing must take place in the new topological relations of adjacent triangles and nodes. The consecutive erase process is again re-arranged the topological relations of the adjacent triangles and nodes. The design of start and end data group in essence is to use computer storage space in exchange time of 3D scene deployment during virtual reality visualization. This ensures the dynamic simplification of irregular triangle in a real time pace.

The area projection ratio (sration) of a node is also called terrain projection ratio in practice. In order to calculate the terrain projection ratio, the area of each adjacent triangles and the total area of all adjacent triangles need to be computed first. Then, the mean elevation plane of all adjacent nodes was calculated. A polygon can be drawn using the projected adjacent nodes on the mean elevation plane. The ratio of area of this polygon to the total area of the adjacent triangles is the terrain projection ratio. In practice, the calculation can be simplified using the ratio of distances of the node and adjacent nodes to the mean elevation plane to replace the ratio of areas. The objective of 3D virtual reality visualization is to provide the user perceived real world image of terrain in a digital environment. Although certain nodes of TIN may supply important characteristics of terrain, the user may not recognize the micro-scale changes made using dynamic TIN simplification method simply owing to that the objects are far away from or on the margin of the observer's view. The significant parameter of a node during the 3D virtual reality visualization may be determined by following model.

$$SP = f(OD, OA, sration)$$

Where SP is the significant parameter; OD is the observation distance; OA is the observation angle; and sration is the terrain projection ratio (area projection ratio).

Node erase process in TIN mesh creates temporary polygons that need to be triangulated locally. Dynamic triangulation of temporary polygons can be identified into three steps. First step is to sequence the adjacent nodes of the erasing node since the adjacent nodes are not sequenced either clockwise or counter clockwise. The algorism is to find any of adjacent nodes of erasing node first. Then, find the common adjacent node of both the previous node and the erasing node. Continue the above process for the new node just found until all the adjacent nodes are located. In the same time, record the nodes by the sequence that they were found. The second step is to identify if the sequencing is clockwise or counter clockwise. The algorism is that find the adjacent node of smallest x coordinate value in the sequenced adjacent nodes. We can name this node as b, the node before it in the sequence is a, and the node after it in the sequence is c. If the slope of straight line ab to the easting line of the coordinate x is greater than bc, the sequence is counter clockwise. Otherwise, it is clockwise. The third step is to identify concave or convex terrain link of the node, and the erasing-ability of the node. The concave or convex terrain link node was identified using the algorism presented by Ma et al. (1998). The erase ability of the node was identified by testing if the irregular triangle formed by the erasing node and two adjacent nodes contains any other node. If the irregular triangle does not contain any other node, the erase is performed, and the new triangulation is conducted. If the irregular triangle contains other nodes, repeat the above erasing-ability test to the node until simple irregular triangle is reached. Erse the node traced at the last step and triangulates the left over polygon. The recursive program records the new irregular triangles formed in the TIN simplification process, and display the new TIN mesh. The flow chart of the dynamic simplification of TIN process during the 3D virtual reality observation is shown in Figure 3.

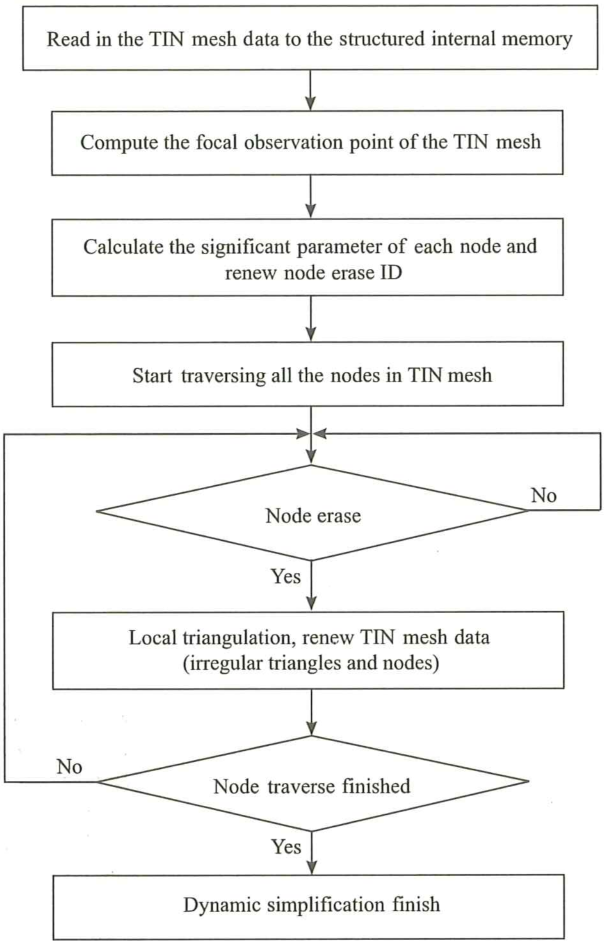


Figure 3. Flow chart of dynamic TIN mesh simplification in 3D virtual reality visualization

III. RESULTS AND DISCUSSIONS

The entire 3D virtual reality visualization software and database in this research was developed in Microsoft Visual C^{++} and OpenGL environment. It was implemented on the Windows operating system platform in a stand alone desktop computer of 1.7 GH CPU and 512MB on-board memory. The video card of the computer is ELSA Gloira4. 3D virtual reality visualization data of the Summer Palace in Beijing was used to test the dynamic retrieval of quad-tree data structure and dynamic simplification of TIN mesh using LOD enabled distance decay. The entire TIN mesh of the data contains more than 100000 nodes. The test indicated that both display speed of the TIN mesh and the virtual travel speed from one location to another in the 3D virtual visualization were drastically increased using combined dynamic data retrieval and TIN mesh simplification method. The new methods streamlined the display transition from one scene to another and avoided the jumping of scenes during the 3D virtual reality tour.

A test was also conducted on the same computer of the same database using the visualization software without dynamic data retrieval and distance decay of TIN mesh simplification. Both the retrieval and virtual tour speeds were too low to make the 3D visualization possible on a PC workstation. Screen captures of detail scene and the scene of close to entire visualization area using dynamic data retrieval and TIN mesh simplification are shown in Figure 4 and 5. In comparison of the two figures, we recognized that there is no compromise of

Figure 4. Detail scene of 3D virtual reality visualization using dynamic data retrieval and TIN simplification

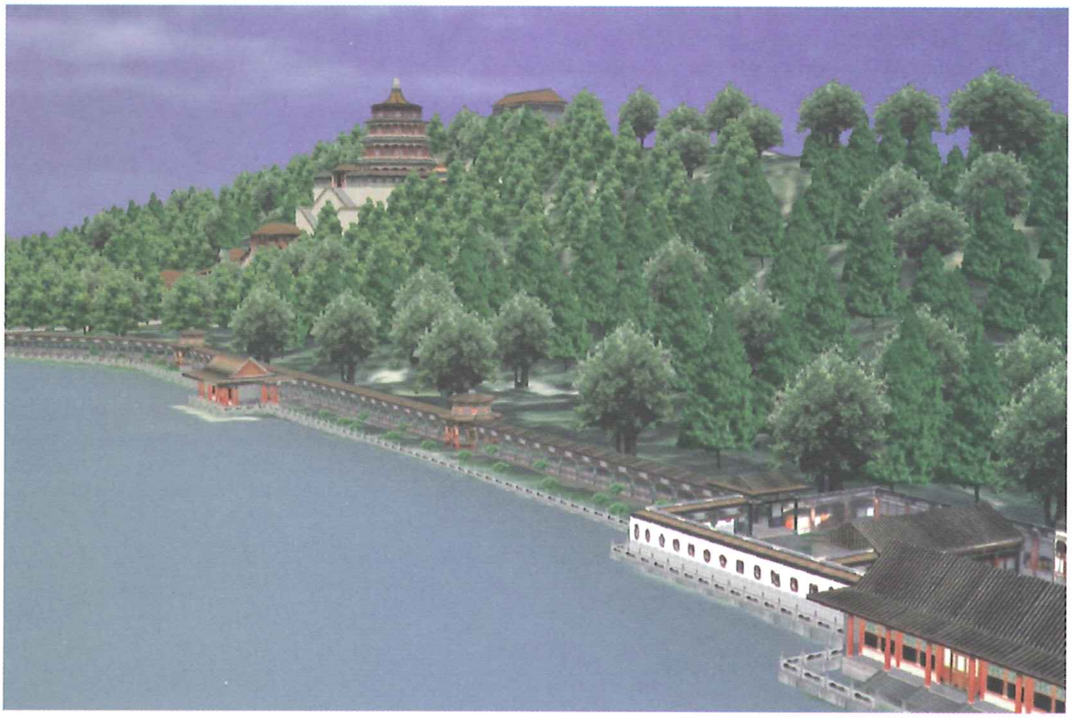


Figure 5. Distance decayed scene at proximity of entire viewing area in the 3D virtual reality visualization using dynamic data retrieval and TIN simplification

feature displays either in a close look or viewing from far away as far as human eye can detect. The dynamic data retrieval and TIN mesh simplification method mimic the human visualization in the real world that is reasonable and logic to eliminate the unnecessary data display.

Currently, one discrepancy still exists in the algorism of dynamic TIN mesh simplification. That is the erased node points can not be accurately replaced with their original geometry in the TIN mesh when a user travels from a scene of general view to a detail scene. In order to ensure the quality of 3D virtual reality display, methods of dynamic data retrieval and dynamic simplification of TIN mesh are bridged during the visualization process in this study. When user travels from detailed scene to scenes of general view, both dynamic data retrieval and TIN mesh simplification methods were applied. When the user travels from less detail to detail, only dynamic retrieval was applied. Algorism extension of accurate recovery of erased nodes of irregular triangles needs to be developed in the future studies.

IV. CONCLUSIONS

This paper presented two methods in improving the data retrieval and display speed during a 3D virtual reality visualization in GIS. These are dynamic data retrieval of LOD using partial retrieval algorism of quad-tree data structure; and dynamic simplifications of TIN mesh during a 3D virtual tour session. 3D virtual reality visualization in general encounters large quantity of digital data. Data retrieval and 3D display of the scenes proving is the bottle neck of the 3D virtual reality. Tests of methodology and algorism developed in this research on a stand alone computer workstation with average CPU speed and on-board memory indicated that both data retrieval and virtual tour speeds are drastically increased. The 3D virtual reality tours were conducted on PC workstations using the method and software of this research otherwise would be impossible.

ACKNOLEDGEMENTS

The authors would like to thank the Beijing Municipal Government, China for a grant in supporting of this research.

REFERENCES

[1] Abdelguerfi M., Wynne C., Cooper E., Roy L., K. Shaw, 1998, Representation of 3-D elevation in terrain databases using hierarchical triangulated irregular networks: a comparative analysis, *International Journal of Geographical Information Science,* 12 (8): 853–873.

[2] Huang B., Jiang B., H. Li, 2001, An integration of GIS, virtual reality and the Internet for visualization, analysis and exploration of spatial data, *International Journal of Geographical Information Science,* 15(5): 439–456.

[3] Kraak M. J., A. Maceachren, 1999, Visualization for exploration of spatial data, *International Journal of Geographical Information Science,* 13 (4):285–287.

[4] Kumsap C., Borne F., D. Moss, 2005, The technique of distance decayed visibility for forest landscape visualization, *International Journal of Geographical Information Science,* 19 (6): 723–744.

[5] Ma X., Pan Z., J. Shi, 1998, Algorism of identifying concave or convex terrain link nodes in a polygon, *Computer Engineering and Design,* 19(3): 45–49(in Chinese).

[6] Milne J. A., D. A. Sear, 1997, Modeling river channel topography using GIS, *International Journal of Geographical Information Science,* 11(5):499–519.

[7] Park D., Cho H.,Y. Kim, 2001, A TIN compression method using Delaunay triangulation, *International Journal of Geographical Information Science,* 15(3): 255–269.

[8] Rhyne T. M., 1999, A commentary on GeoVRML: a tool for 3D representation of georeferenced data on the web, *International Journal of Geographical Information Science,*1999, 13(4):439–443.

[9] Speckmann B., J. Snoeyink, 2001, Easy triangle strips for TIN terrain models, *International Journal of Geographical Information Science,* 15(4): 379–386.

[10] Masumoto S., Raghavan V., Yonezawa G., Nemoto T., K. Shiono, 2004, Construction and Visualization of a Three Dimensional Geologic Model Using GRASS GIS, *Transactions in GIS*, 8(2): 211–223.

[11] Verbree E., Maren G. V., Germs R., Jansen F., M. J. Kraak, 1999, Interaction in virtual world views - linking 3D GIS with VR, *International Journal of Geographical Information Science,* 13(4): 385–396.

[12] Yang B., Shi W., Q. Li, 2005, A Dynamic method for generating multi-resolution TIN models, *Photogrammetric Engineering and Remote Sensing,* 71(8): 917–926.

[13] Zhu H., Eastman R. J., J. Toledano, 2001, Triangulated irregular network optimization from contour data using bridge and tunnel edge removal, *International Journal of Geographical Information Science,*15(3): 271–286.