DESIGN AND IMPLEMENTATION OF FEATURE-BASED 3D GEO-SPATIAL RENDERING SYSTEM USING OPENGL API

Seung-Yeb Kim and Kiwon Lee

Dept. of Information System Engineering, Hansung University
E-mail : {gisksy, kilee}@hansung.ac.kr

ABSTRACT:

In these days, the management and visualization of 3D geo-spatial information is regarded as one of an important issue in GIS and remote sensing fields. 3D GIS is considered with the database issues such as handling and managing of 3D geometry/topology attributes, whereas 3D visualization is basically concerned with 3D computer graphics. This study focused on the design and implementation for the OpenGL API-based rendering system for the complex types of 3D geo-spatial features. In this approach, 3D features can be separately processed with the functions of authoring and manipulation of terrain segments, building segments, road segments, and other geo-based things with texture mapping. Using this implementation, it is possible to the generation of an integrated scene with these complex types of 3D features. This integrated rendering system based on the feature-based 3D-GIS model can be extended and effectively applied to urban environment analysis, 3D virtual simulation and fly-by navigation in urban planning. Furthermore, we expect that 3D-GIS visualization application based on OpenGL API can be easily extended into a real-time mobile 3D-GIS system, soon after the release of OpenGL|ES which stands for OpenGL for embedded system, though this topic is beyond the scope of this implementation.

KEY WORDS: OpenGL, 3D-GIS, Rendering System, Prototype

1. INTRODUCTION

3D geo-spatial information in GIS is required in recent industry issues such as ITS, LBS, Telematics, landscape analysis/simulation and visualization. While, the management and visualization of 3D geo-spatial information is regarded as one of an important issue in GIS and remote sensing fields. 3D GIS is considered with the database issues such as handling and managing of 3D geometry/topology attributes, whereas 3D visualization is basically concerned with 3D computer graphics.

In this study, we designed a prototype of rendering system in 3D GIS. In this prototype, the design and implementation for the public domain OpenGL API-based rendering system was carried out with the consideration to the 3D GIS database issues. However, non-provision of 3D graphic data structure and shortage of user interactive interface as two main weak aspects of OpenGL prevents prevent it from migrating to 3D GIS handling large volume of database.

Despite these, we attempted to design rendering system using OpenGL API, to demonstrate easy accessibility of pixel pipeline. In this system, 3D features can be separately processed with the functions of authoring and manipulation of terrain segments, building segments, road segments, and other geo-based things with texture mapping.

This integrated rendering system based on the feature-based 3D-GIS model can be effectively utilized to urban environment analysis, simulation and navigation. Furthermore, it is expected that 3D-GIS visualization application based on OpenGL can be easily extended into a real-time mobile 3D-GIS system, soon after the public release of OpenGL|ES, and it is thought that this approach contributes to basic foundation for this mobile system. This study achieved to Visual C++7.0 under Windows XP.

2. SYSTEM DESIGN

2.1 Prototype Overview

Fig 1 shows a reference model for 3D-GeoSpatial Rendering System. Especially we tried to feature-based approach, separate authoring and manipulating terrain segment, building segment, road segment, or other geo-based things from GUI. The logical division between the data management and the other four view workspaces as urban examples supports feature manipulation. Also, we tried a spatial DBMS approach, authoring and manipulating spatial information, attributes information using DBMS and File System.

2.2 OpenGL (Open Graphic Library)

OpenGL is graphic Application Programming Interface (API) that can control graphic hardware. OpenGL provides core pipeline functions that can process more easily complicated 3D information. OpenGL has a low-level rendering function that offers geometry primitive of point, line, polygon. Therefore, OpenGL is not offering a windows manipulation, GUI and data structure. But OpenGL provides a function of high level library such as a geometry primitive, GLU, Glut, and WGL(Windows Graphic Library) can express complicated model in GIS.

As well, OpenGL can embody special effects functions such as a RGBA color type and lighting, shading, blending, fog, texture mapping function to help more realistic rendering process, shown as a simple case of Fig. 2. Therefore, OpenGL can easily represent the 3D GIS model. Another motivation of this study is that OpenGL can be easily extended in real-time mobile 3D-GIS, soon after the public release of OpenGL|ES, and it is thought that this approach contributes to basic foundation for this.

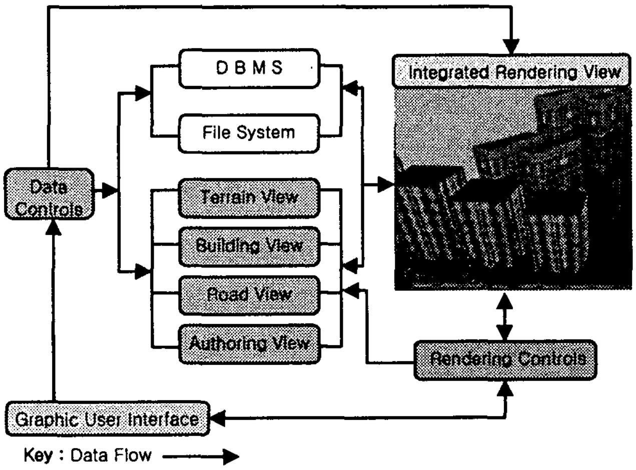

Fig. 1. Reference model for 3D-GeoSpatial rendering system in this prototype.

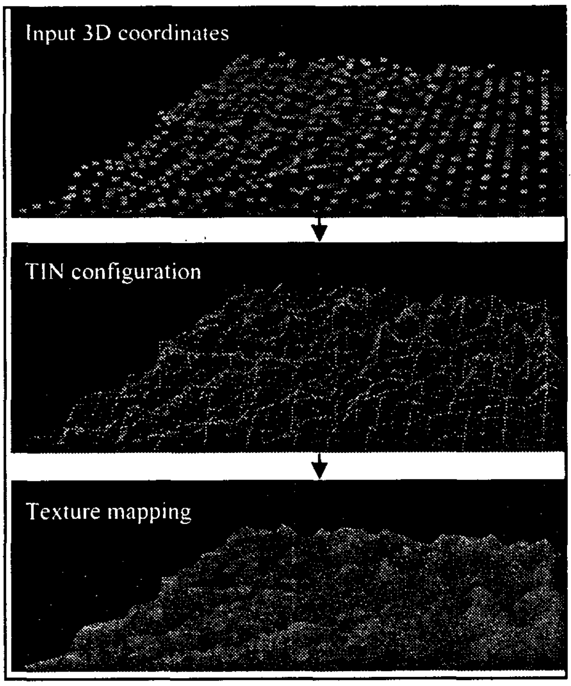

Fig. 2. TIN generation/rendering process using OpenGL: A simple case.

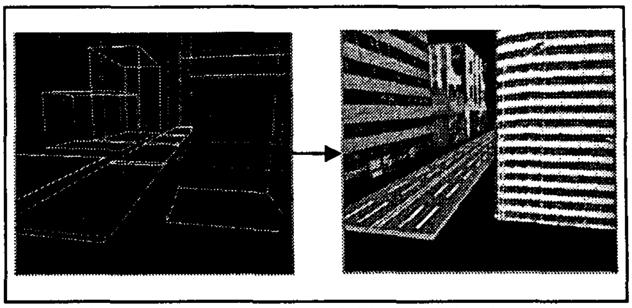

Fig. 3. A case of urban modelling and draping.

2.3 Visualization

In this system, a core part of this rendering system is composed of 3D urban feature modelling and following draping Image. Draping is a conventional function in the 3D visualization techniques and can be applied to rendering process of 3D shape of features within the real world.

Fig. 2 ~ 3 show image draping case. Image draping extract 3D coordinates in Lidar, DEM, image which have a 3D coordinates (x,y,z), Altitude, Heightfield. And it generates surface relief through TIN to construct surface of terrain. Fig 4 shows a building and road, as urban feature, by rendering function.

2.4 Geo-DBMS

In this study, it put essential point and made progress that literacy connected with OpenGL than DBMS design of geometry information.

Database within the 3D geospatial rendering system includes geo-spatial data and attributes components. Geometry Information consists of index of objects and 3D coordinates of objects, and attributes of terrain, building and road are stored to attribute Information. Also, attribute information is linked to geometry components.

Fig. 4 shows a structure from this point of view DB. Database operation function controls transfer of various data between 3D GIS database and integrated rendering view. Also, file system that supply various data pass the data of 3D coordinates, image, object to 3D GIS Database and integrated rendering view through database operation function.

3. IMPLEMENT

This study focused on the design and implementation for the OpenGL-based rendering system in the consideration to 3D GIS database. GUI and class diagram of main modules are shown with Fig. 5 and Fig. 6, respectively.

3.1 Graphic User Interface

Because 3D GIS data is large volume of data sets, GUI helps to structure of feature-based system that can process much information and efficient memory structure.

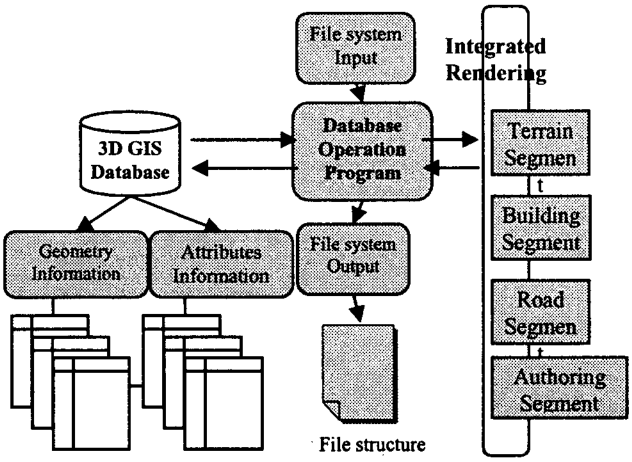

Fig. 4. Attributes handling in this prototype.

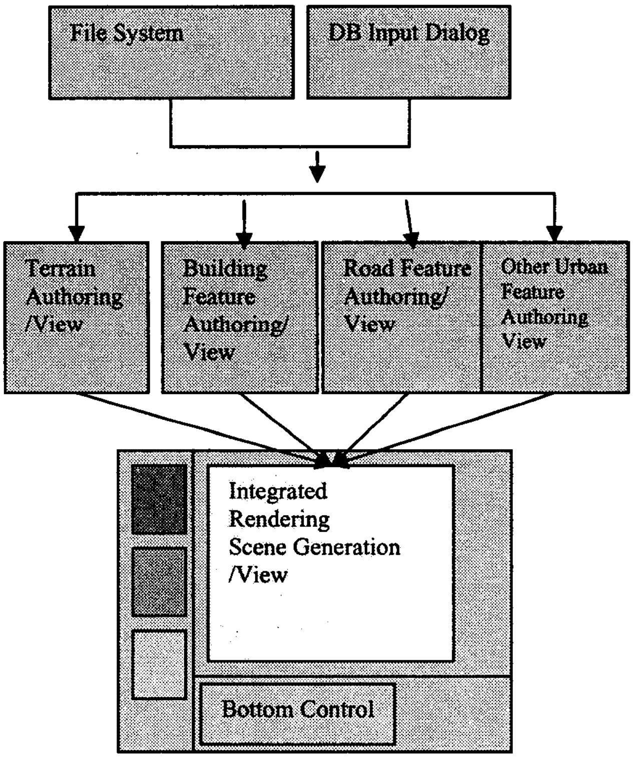
File System
DB Input Dialog
Terrain Authoring /View
Building Feature Authoring/ View
Road Feature Authoring/ View
Other Urban Feature Authoring View
Integrated Rendering Scene Generation /View
Bottom Control

Fig. 5. Graphic user interface of system.

Fig. 5 shows a GUI which can manipulate in integrated window handling by terrain view, building view, road view, authoring view. Selected object in GUI base needs necessary attribute data to DB or file system. These data communicates with each sub-module and the data process individually in the another sub-module.

Control box for object manipulation offers on each sub-module. In the main module, some frames such as attribute information, tree view and control box to manage composition object are provided, like most commercial ones.

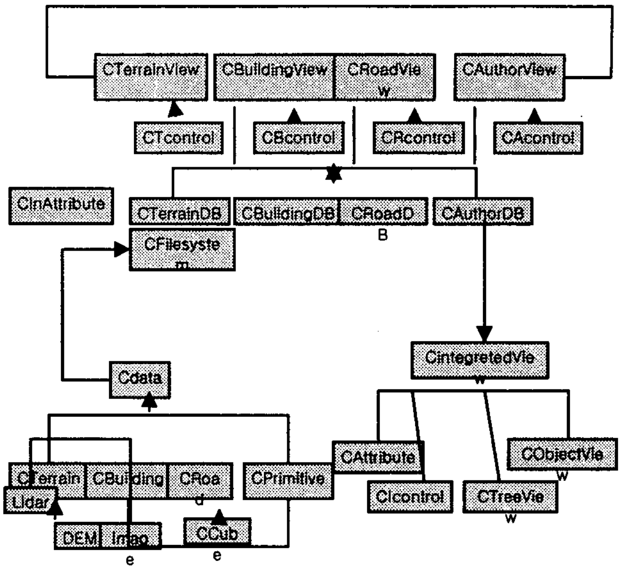

Fig. 6. Class diagram of system.

3.2 Class Diagram

Class structure can divide a data I/O, data manipulation, sub-program, and main program. First, data input/output part has file I/O function, attribute information and geometry information I/O to create. Second, data manipulation is linking process for geometry and attribute information, and all data that is used using file system or DBMS in program. Third, sub-program connected with terrain, building, road, authoring function is part that process huge data of GIS individually. Finally, the main program is part that can create and manage data or composition model that user wants reading data in DB, File system and created data in sub-program. In Fig. 6, class diagram shows each component.

Data in this study calls require model through CData class which inherit Cterrain CBuilding, CRoad and CPrimitive. And this data is sent to each connected subsystem. Result data of sub-program is saved in file system. This can be used in the main program to create complex 3D models (Fig. 7, Fig. 8).

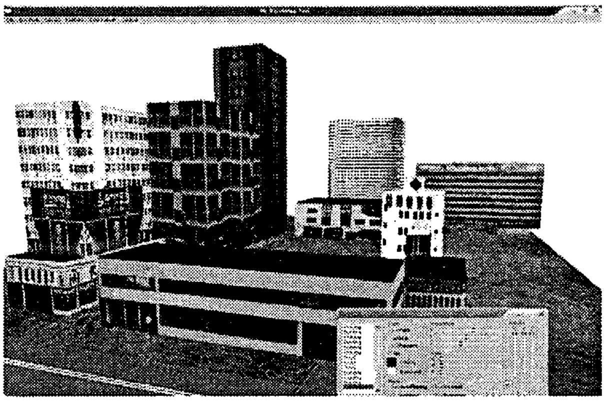

Fig. 7. Integrated scene, produced using this prototype.

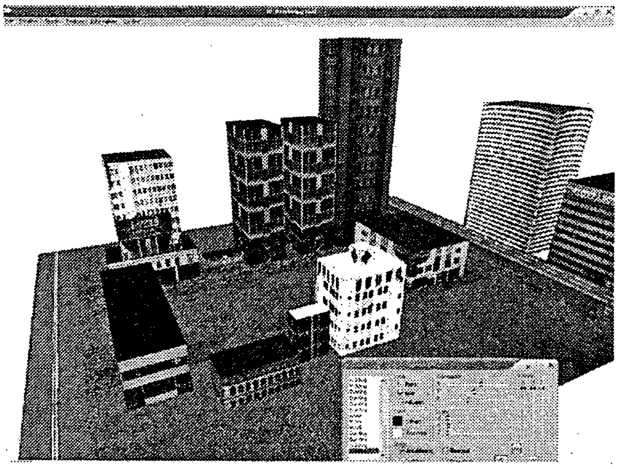

Fig. 8. Integrated scene according to other viewpoints.

4. CONCLUSION

In this study, design and implementation of rendering system as core of 3D GIS was performed using OpenGL base by graphic pipeline/rendering function. Basically, it orients feature-based system; in this system, composition object integration processing and rendering processing are available. This integrated rendering system based on the feature-based 3D-GIS model can be extended and effectively applied to urban environment analysis, 3D virtual simulation and fly-by navigation in urban planning. Furthermore, we expect that 3D-GIS visualization application based on OpenGL API can be easily extended into a real-time mobile 3D-GIS system, soon after the release of OpenGL|ES which stands for OpenGL for embedded system, though this topic is beyond the scope of this implementation.

References

Y. Takase, N.Sho, A. Sone, K. Shimiya, 2004. Automatic generation of 3D city models and related applications. Romote Sensing and Spatial Information Sciences, Vol. XXXIV-5/W10.

Siyka Zlatanova, Alias Abdul Rahman, Wenzhong Shi, 2004, Topological models and frameworks for 3D spatial objects, Computers & Geosciences 30 419-428.

Michael Beck ViewTec AG, Zurich ,2004, Real-Time Visualization of big 3D city models, Romote Sensing and Spatial Information Sciences, Vol.XXXIV-5/W10

Calin Arens, Jantien Stoter, Peter van Oosterom, 2005, Modelling 3D spatial objects in a geo-DBMS using a 3D primitive, Computers & Geosciences 31 165-177

Kilian Ulm, 2004, Improved 3D city modelling with cybercity-modeler using aerial, satellite imagery and laserscanner data, Romote Sensing and Spatial Information Sciences, Vol. XXXIV-5/W10.

Arnaud de la Losa, Bernard Cervelle, 1999, 3D Topological modelling and visualisation for 3D GIS, Computers & Geosciences 23 469-478