CyberCity GIS (CCGIS): Integration of DEMs, Images, and 3D Models

Qing Zhu, Deren Li, Yeting Zhang, Zheng Zhong, and Duo Huang

Abstract

A CyberCity is a virtual representation of a city that enables a person to explore and interact, in cyberspace, with the vast amount of environmental and cultural information gathered about the city. A GIS software for CyberCity, called CCGIS, has been developed, and this paper reports its technical characteristics, including the three-dimensional hierarchical modeling technique, the integrated database structure, and the interactive method of visualization of the three-dimensional data of urban environments. The effective integrated data organization strategy for dynamical loading and progressive rendering, which enables CCGIS to support the development, design, and presentation of a large CyberCity, is stressed. Finally, a pilot project for CCGIS software application is also demonstrated.

Introduction

Since geographic information system (GIS) technologies have been successfully applied in urban planning and management, there have been increasing demands for three-dimensional (3D) realistic representations of GIS about the urban environment in spatial planning, design, and decision-making applications (Ranzinger and Gleixner, 1997; Germs *et al.*, 1999; Pullar and Tidey, 2001). Examples include making virtual walk-through streets of a city, virtually visiting a hotel room, planning a tour, the computer-aided design (CAD) of a garden, etc. (Fritsch, 1999). This demand spurred the research and development of 3D GIS, and the transition from two-dimensional (2D) GIS to the completely 3D GIS of a city's environment has therefore become the most important modern issue in this area (Gruber and Wilmersdof, 1997; Li *et al.*, 2000b). Even though the research and development of 3D GIS face considerable challenges, strong activities to support this important step in the evolution of a city's GIS are evident. It indicates that CyberCity is more often mentioned than 3D GIS. This paper therefore first examines the differences between CyberCity, 3D GIS, and CyberCity GIS (called CCGIS), and then introduces the research and development of a CyberCity GIS prototype in China.

The rest of this paper is organized as follows: after briefly reviewing the historical development in related areas, we will emphasize the object-oriented 3D data model, the hierarchical modeling methods, the integrated database issues, the interactive visualization strategy, and the pilot application of CCGIS. Finally, the summary and conclusions will be presented.

From CyberCity to CyberCity GIS

The term "CyberCity" is used to represent the virtual representation of a city that enables a person to explore and interact, in cyberspace, with the vast amounts of environmental and cultural information gathered about the city (Gruber, 1999; Gruen and Wang, 1999a; Li *et al.*, 2000a). The CyberCity not only shows data in three dimensions, i.e., 3D city models in most cases, but also presents a photorealistic surface description (Gruber and Wilmersdof, 1997). Therefore, the description of surface character and material parameters, including geometry, photo texture, and additional information, are the contents of a CyberCity database, and this would result in the CyberCity database of an entire city containing some hundreds of gigabytes of data (Gruber, 1999).

Even though conventional 3D CAD packages or virtual reality systems are able to model and visualize the CyberCity, they are not suitable for the data management of vast data sets as mentioned above. Because GIS is a generic tool for storing, manipulating, querying, and displaying geographically referenced data, it is a sensible choice to use a GIS as the support platform for the CyberCity. However, the most common 2D GIS has little or no information about the 3D relationship between a city's buildings, traffic areas, terrain, etc. The ability to interactively visualize design ideas in three dimensions is a vital demand in urban design. Current commercially available 2D GIS are, unfortunately, poor at handling and displaying three-dimensional information (Dodge and Jiang, 1998). For this purpose, there are two quite different strategies for coupling the 3D visualization toolkit with a GIS (Dollner and Hinrichs, 2000). The first strategy involves using the visualization toolkit as an independent system, e.g., the MultiGen-Paradigm's SiteBuilder 3D as a plug-in to the ArcView GIS system from the Environmental Systems Research Institute, Inc (ESRI) (MultiGen-Paradigm, 2000). The main disadvantage of this approach is that the visualization features are restricted by the file exchange format and that dynamic or large data sets are thus difficult to transfer. Alternatively, the visualization toolkit could be integrated as part of the GIS (tight coupling), communicating through shared data structures at the object level, or by object communication services: this is the ideal approach. In comparison with the advances in 3D visualization and virtual reality, relatively little has been accomplished in the realization of a practical 3D GIS. Furthermore, commercial 3D GIS software for general purposes is still not available.

Aiming at the increasing requirements of 3D GIS for the urban environment and the situation of 3D GIS development, a hot topic in recent years has been the development of a special 3D GIS for the CyberCity, which the authors have called CyberCity GIS, as explained below (Kofler *et al.*,1996; Gruen and Wang, 1999b). The CyberCity GIS is just one prototype for a real 3D GIS. Unlike the true 3D GISs needed in geological or oceanic applications, the main objective of the current CyberCity GIS is to deal with 3D solids like buildings as surface models, tempo-

National Key Lab for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan, Hubei, 430079, P. R. of China (zhuq@rcgis.wtusm.edu.cn).

Photogrammetric Engineering & Remote Sensing
Vol. 68, No. 4, April 2002, pp. 361–367.

0099-1112/02/6804–361$3.00/0

rarily ignoring complicated 3D topology issues. It does not matter how the 3D real world is mapped into spatial databases; the main point is the availability of a 3D capability in a GIS, which should be realized in an efficient and robust manner (Fritsch, 1996). Compared with the traditional 3D GIS, the CyberCity GIS faces several critical issues for 3D data handling:

- the data model, facilitating both 3D visualization and GIS operations;
- the hierarchical object-orientated modeling of 3D data;
- support for the levels-of-detail (LOD) concept, both on the database level and on the dynamic simplification level;
- support for the integration of vast amounts of data with different types and different resolutions, including raster data such as the digital elevation model (DEM), the digital orthoimage, and texture images, and vector data such as 3D building models; and
- support for interactive dynamic 3D visualization.

In China, one of the well-recognized research and development centers for GIS, GPS, and remote sensing—the National Key Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing—has been specializing in the development of general purpose 2D GIS platform software for more than eight years. GeoStar, the laboratory's flag-ship GIS product, has been widely used in broad discipline areas (Gong and Li, 2000). In order to satisfy the increasing needs of the CyberCity GIS, aiming at the solution of the above critical issues, we have been developing the prototype of the CyberCity GIS, called CCGIS, since 1996. The first issue we faced was to design a proper data model for both the management and visualization of vast 3D city models. Therefore, the conceptual 3D data model based on the object-oriented method will be introduced in the next section.

Object-Oriented 3D Data Model

Similar to the formal data structure (FDS) (Molenaar, 1992; Gong, 2000; Wang, 2000), the CyberCity models are grouped into six different object types: DEM object, image object, point object, linear object, surface object, and body object. Both the DEM and image objects are not only fundamental data types for the description of urban landscapes, but also the base of 3D reconstruction of city models. In most cases, gridded DEM and image objects are considered as raster objects, and the others are 3D vector objects. All six objects possess geometric and thematic and/or metadata information, and the point, linear, surface, and body objects may possess texture information. The 3D vector objects and the relationship between geometric element and object types are defined as below:

- The 3D point object, the cube object, has the simplest spatial extension (the 3D position, the azimuth, and the box size) and sometimes special attributes such as CAD models in 3D Studio (3DS) format. In the CyberCity, the tree models, the street lamp models, and any elaborate 3D CAD models are usually identified as 3D point objects with special attributes.
- The 3D linear object, built up of connected line segments as a 3D points string, is mostly used to describe pipelines and communication/electric power lines. Pipelines possess extensions in aperture size and are visualized as an assembly of cylinders and joint-elbows.
- The 3D surface object is composed of one or more facet primitives such as TIN or convex polygon. Lakes, road surfaces, and green space are represented as surface objects.
- The 3D body object is bordered by special facets: for instance, a building is composed of walls, the roof, and the floor. In the CCGIS, there are two classes of body objects: one is the simple solid, i.e., primitive entities such as cubes, cylinders, spheres, and so on. The other is the more complicated solid, usually composed of a series of primitive entities.

Even though there are several methods for the description of solids, such as wire-frame, boundary representation (BR), constructive solid geometry (CSG), and so on (Molenaar, 1992; Guo, 1996), we employ only the facet model to describe the most important solid objects of the CyberCity, such as buildings, which is one of the BR techniques making use of facets. The vertices, edges, facets, and their relationships are explicitly defined. There are two reasons to use the facet model: first, due to the fact that most of the spatial applications of 3D city models are based on the outline geometry of buildings, i.e., the boundary; and second, to enable efficient real-time visualization of complex objects. Of course, a more detailed description of a building, including both exterior and interior structures, can be stored as a point object with special attributes (i.e., a 3DS file) or divided into a series of facet surfaces. Figure 1 illustrates this object-oriented data model, considered as one of the superclasses of all spatial features. Each superclass is divided into sub-classes according to its thematic attribute. One or more feature classes can constitute a logical feature layer.

Furthermore, the data structure of the 3D object is one of the most significant factors for efficient data access and fast rendering. For example, to benefit the general rendering methods based on OpenGL or DirectX, the following structure is designed for 3D surface objects:

```
{
object ID,
feature ID,
visibility (hide or display),
facet type (convex or concave: the concave facet consists of
   TIN; the convex facet is just a polygon),
3D box range,
boundary points number and 3D coordinates,
texture information (ID, type, and mapping parameters).
}
```

where the object ID and feature ID are the unique identifiers of an object in the database for geometric data and attribute data, and the visibility value is used to determine whether it is visible or not. This kind of database structure is designed to support the visualization and manipulation of 3D objects, which is favorable for the dynamic generation of virtual models and real-time rendering, because all the object data, including the geometric and texture descriptions, can be directly transformed into a display from the database. This enables the CCGIS to support the development, design, and presentation of a large CyberCity.

LOD Concept and Hierarchical Modeling Methods for CyberCity

Models at multiple levels of detail (LOD) are intensively used to control scene complexity and to accelerate rendering for real-time visualization of complex 3D scenes (Gruber, 1999; Pan *et al.*, 1998). Support for the LOD concept therefore becomes one significant feature of the CyberCity GIS. LOD models stand for a sequence of models with various resolutions and quality for

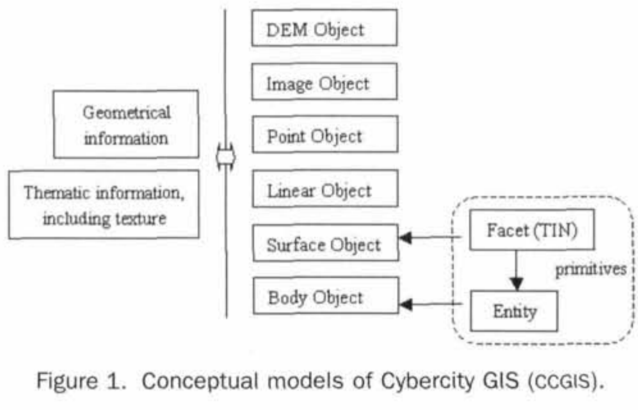


Figure 1. Conceptual models of Cybercity GIS (CCGIS).

the same object in a scene. A multi-resolution concept for images (e.g., an image pyramid) is the equivalent method for raster data. However, for 3D vector data, the realization of the LOD concept is more difficult. There are two schemes for preparing LOD models, i.e., real-time simplification or creating in advance. The view-independent methods are usually adopted for simplification in advance: these LOD models are then stored in a database, and during the rendering process a specific LOD model is selected depending on the position of the viewpoint (Zhou *et al.*, 2001). This kind of LOD concept is considered as being on the database level. Otherwise, view-dependent methods are used to dynamically simplify the object model in real time. Because of the complexity of CyberCity, the CCGIS supports the hybrid method of these two schemes. Thus, the first step is to design the hierarchical modeling methods for LOD models generation.

The DEM and aerial orthoimage are stored at multi-resolutions as a pyramid, and the data at lower resolution is automatically extracted from the highest resolution or from other data sources. However, the texture images of the object surface are stored at a certain unique resolution without the LOD concept. Aerial images are exploited for roofs, terrain, and other horizontal faces, and all of these surfaces' texture images can be automatically extracted and associated from the orthoimage. Facades and vertical faces have to be recorded by additional photographs taken from the street level. The general image processing of raw photographs, such as the modification, adjustment of brightness, and contrast, can be accomplished by special software such as Adobe's Photoshop. However, the CCGIS supports interactive texture mapping, so that image rectification based on affine transformation, the association with a specific surface, the normalization to standard size, and the choice of multiple mapping modes such as repeated and transparent mapping are available. All the raster objects can be dynamically simplified during the rendering process as described in the later section on 3D Dynamic Interactive Visualization.

A large CyberCity GIS is possible only if the 3D modeling can be done fast and (at least semi-) automatically. For 3D vector objects, many approaches for automatic and semi-automatic 3D reconstruction, especially for buildings from aerial images and from 2D GIS/CAD data, have been proposed in recent years (Gruen *et al.*, 1999a; Fritsch, 1999; Gruber, 1999). Based on the above data models and according to the level of detail (LOD) concept, the CCGIS provides three different 3D modeling schemes:

- Towards a higher level of detail, the elaborate models of some very complicated and noticeable buildings and facilities such as a TV tower and city hall can be generated by 3DSTUDIO™ or MULTIGEN™. The 3D CAD models can then be imported into the CCGIS in 3DS format in two manners: one is to completely transform the CAD model to the CCGIS model and then simplify this model during rendering; the other is to import each 3D CAD model as a one-point object with special attributes, but this model will be loaded as a whole for rendering. Such kinds of models can present the full details of objects, including interior and exterior features.
- Towards the middle level of detail, based on photogrammetry, laser scanning, and other ground surveying means, 3D coded data and real texture images can be obtained efficiently to describe the status of the cityscape. The city model is classified into different primitive entities according to the shape characteristics for data sampling, such as point, line, sphere, cylinder, polygon with feature points, and lines, and each primitive entity is sampled as a series of feature points and identified by class code and user code. Using these sampled data, the CCGIS supports the automatic generation of the topology of 3D vector objects, especially the reconstruction of the shape of roofs employing an automatic TIN algorithm within the concern of features (Li and Zhu, 2000) and the 3D geometric adjustment, i.e., to level the surface, to make one side parallel with the other, and to make the two sides perpendicular. These are the remarkable characteristics of the CCGIS for modeling with ease.
- Towards a lower level of detail, just based on the 2D GIS data, i.e., the bottom boundary data of buildings and its height attributes. The CCGIS supports the automatic conversion from 2D vector data to 3D vector models within planar roofs.

Using the above hierarchical modeling methods, from the street to each building and even to each room, various details of the city model can be expressed at the database level. Complex building models then can be assembled in the CCGIS's interactive editing environment by the use of primitives, and all the attributes including texture, material, and multimedia description also can be created and associated. If necessary, the existing models can be inspected and edited, for example, to append or delete any kind of object elements (point, line, surface, or body), or to modify the geometric shape of buildings, such as to change a planar roof to a herringbone roof. For the quality control purposes of 3D modeling, the CCGIS supports reliable geometric inspection. All the objects can be displayed one by one overlaying on the orthoimage in 2D, and it is very easy to check if the boundary features of an object coincide with the image edge or not. Furthermore, the 3D city models generated by the CCGIS can also be exported into a few open formats for sharing, such as the VRML format and the 3DS format.

Integrated Database Management

Because CyberCity consists of 3D vector models, texture images, DEMs, and multimedia attributes, CyberCity GIS must support effective data access through integrated database management. Based on the successful 2D database technology of GeoStar, CCGIS improves the 3D database structure and data handling, including the spatial index, data selection, and clipping in perspective space; the data subsection process; and dynamic loading (Zhu, 2000; Gong, 2000). Because the hybrid management system of files and the relational database system is hardly able to manage and index large amounts of data and to control multi-user access, as an alternative to the object-oriented database system which is still used infrequently for GIS, an object relational database system (ORDBS) like Oracle8.i is the first choice in CCGIS for a large CyberCity, and the file system is also provided for some special applications in small districts. For the purpose of fast visualization of 3D city models, based on the object-oriented method, we design an effective data organization strategy combining the LOD concept for the integration of raster and vector data.

DEM and Orthoimage Organization

The DEM is the information carrier of all other spatial data, and also forms the fundamental framework of the landscape. Thus, if the study area is very large, such as an entire city, and the DEM data organization is critical for its dynamical loading and rendering, this problem would become more serious when it is necessary to create LOD models in real time. Therefore, we considered the handling of large amounts of DEM databases in the pyramid gridded index manner and the tile-block structure as shown in Figure 2. The DEM databases are in a pyramid structure within multiple resolutions, each layer of the DEM being divided into a series of standard tiles, with each tile subdivided into standard blocks, a block consisting of rows and columns of data elements stored as a standard record of the ORDBS table. This kind of hierarchical structure facilitates the quick spatial retrieval and seamless access of raster data. Corresponding to the DEM, the orthoimage as the texture of terrain surface is also organized in a pyramid structure in the same layer-tile-block manner, the spatial locations and ranges of tile and block being identical to those of the DEM. Because the resolution of the orthoimage is usually higher than that of the DEM in the same layer, the data volumes of each block in the same layer are quite different. For example, a DEM block may consists of 1K nodes

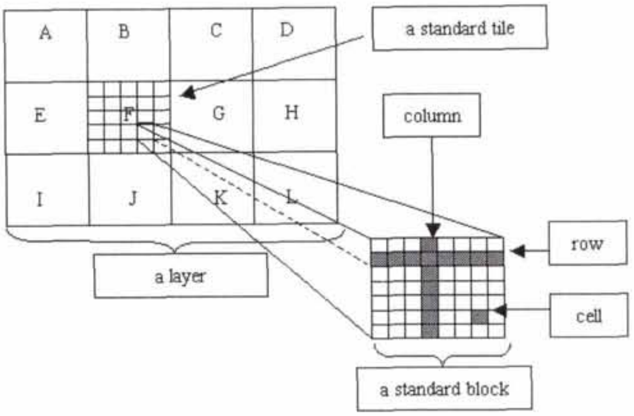


Figure 2. The tile- and block-based structure of raster data.

as 4K bytes of elevation values, but a digital orthoimage model (DOM) block may involve 64K pixels as 196K bytes of RGB color values. In order to further improve the performance of the database system, each block's data are compressed to less than 50 percent of their original volume without loss.

3D Vector Objects Organization

To satisfy the demands of the CyberCity GIS for sufficiently fast interactive visualization, the vector data structure must comply with the following requirements: fast spatial queries and adaptive LOD control. Because of the variable length of records for vector data, if a database table consists of too many records, its performance will be decreasing very rapidly. Aiming at the requirements for efficient management of vast data sets, we adopted the strategy of dividing the area and classifying the features. As shown in Figure 3a, the entire city is divided into a series of subareas, the rectangular bounding boxes of subareas defined by any meaningful concept according to the building density, such as administrative districts or city blocks. The number of objects in each subarea is limited to ensure that each database table is the optimal size for excellent performance. The 3D objects are organized according to the classified features and stored as clusters (Gong and Li, 2000; Zhu *et al.*, 2000). Unlike the node-arc-polygon relationship in the 2D GIS, even if a building involves several adjacent primitive entities and each entity is assembled by facets, we just create one independent table to store such hierarchical data types. As shown in Figure 3b, all the geometric data about one object are stored as one record in the building table, no matter how complicated it is. Only the bounding box is saved as the 2D surface in order to use the 2D spatial index supported by the Oracle Spatial for fast query. Because the first level of data retrieval is based only on the outline box of the object in the Oracle DBMS, the results may not be unique, and the second level of data retrieval is then used to obtain a more accurate index by using the geometric shape data. In particular, the geometric data of 3D vector objects are compressed before being stored into databases and decompressed after retrieval. The encoding and decoding methods are similar to the Winzip tool; the compression ratio can reach 10 to 20 percent, and the total data access efficiency is improved very distinctly.

Organization of Texture Image and other Attributes

The terrain texture as a DOM is stored in the form of a multi-resolution bitmap (bitmap pyramid) and in a unique database as mentioned above. Herein the texture images are only used to describe the 3D object surface characteristics. There may be a few surfaces which share the same image pattern, so the texture image database should not save the images repeatedly. For this purpose, only the basic texture information such as the ID and mapping parameters are saved in the 3D models database related to the geometric data; the texture image data are saved as files in a special texture database. The same texture image with the same ID will possess only one data file.

However, all the attribute data of the CyberCity are classified according to the features, but not subdivided in area. Through the object ID and the feature ID, the geometric data can be associated with the attributes as well as the texture images, and, by means of the database functions, many statistical queries and computations can be realized with ease.

For the LOD concept of vector objects at the database level, the CCGIS prototype provides a simple process, i.e., one object can possess other more detailed CAD models as special attributes. When there is a requirement to visualize the CyberCity at a specific level of detail, the CAD model can be retrieved and displayed independently or embedded into the scene.

3D Dynamic Interactive Visualization

Even if the entire database of a CyberCity is available, but it is no longer possible to keep all the data in the memory of the workstation, 3D dynamic interactive visualization will become a fundamental feature of a CyberCity GIS (Zhu, 1998). Dynamic visualization stands for the real-time loading of the necessary data subset, and interactive visualization means the manipulation in virtual modes, such as the walk-through or fly-over models. Meeting the needs of dynamic interactive visualization not only requires a well-organized database system supporting a fast data retrieval mechanism, but also demands progressive rendering techniques supporting the real-time generation of virtual models at different levels of detail.

As shown in Figure 4, the memory paging technique is first used for dynamic data loading and the real-time generation of virtual models, i.e., a certain amount of memory is located at first. For a DEM and orthoimages, the volume and subdivision of the memory page coincide with the block as planned when creating the databases. For the object texture image and the vector models, the volume of the memory page depends only on the planned rendering times and number of triangle facets per frame. Only at the start should the full page of data be loaded; at any other time only small parts of data should be loaded and updated, so we call this the dynamic data page. Because of the special data organization, the DEM and orthoimage from the database to the data page in the memory are simply mapped from block to block based on the main resolution computed according to the view field. Through this kind of process, the time needed for dynamic data loading is limited to within a very short range.

Another efficient measure to improve the performance of real-time visualization is to simplify the virtual model for each frame, i.e., the LOD control at the real-time level. If we want to obtain a reasonable frame rate per second at a mid-level desktop workstation, the maximum number of triangle facets per frame is usually limited to within the range of 10,000 to 40,000, and the total volume of texture images including the terrain texture is less than 20 megabytes. However, the length of the view frustum may be changed from several cells to several blocks as demanded during the animation. If this happens, we have to control the LOD for each frame, i.e., to keep the higher detail level of objects near the viewpoint and to simplify the objects far away from the viewpoint to the lower detail level, as shown in Figure 5. In the data page, the objects in different blocks may be transformed to be displayed at various levels of resolution and quality. There are two approaches to fulfilling this demand: the first is just to directly load the LOD models from the databases; the other is to simplify the models in real time.

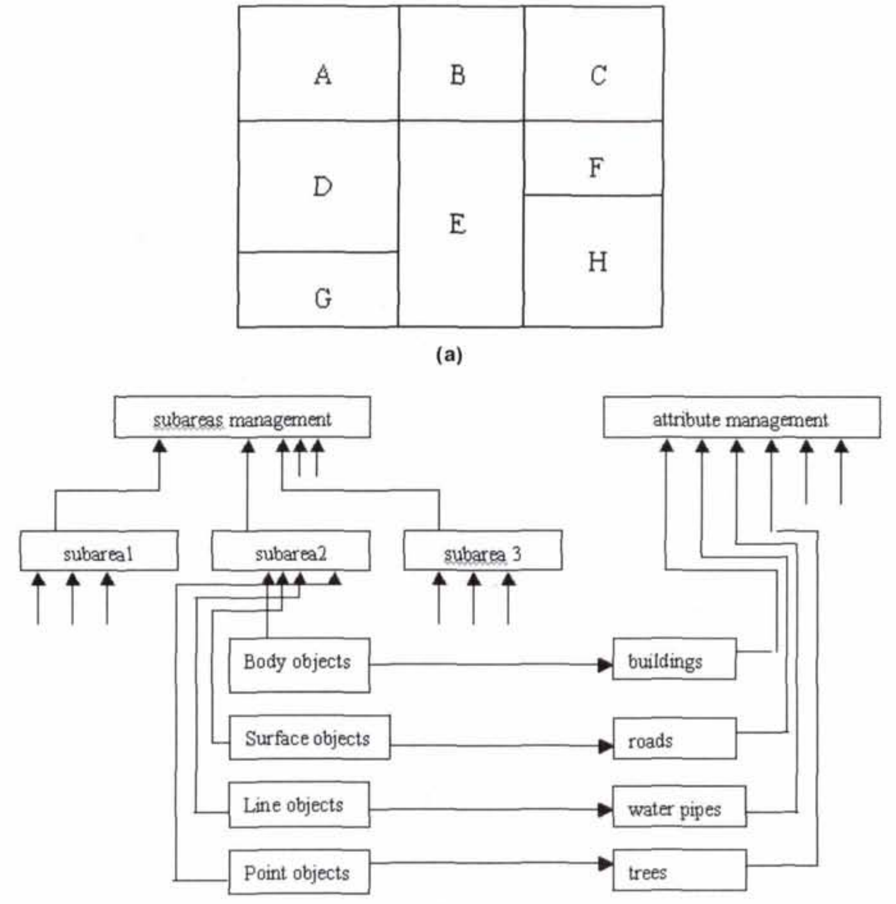


Figure 3. The divisional and classified clustering of 3D vector data. (a) Subdivision of study area. (b) Classified clustering of vector data.

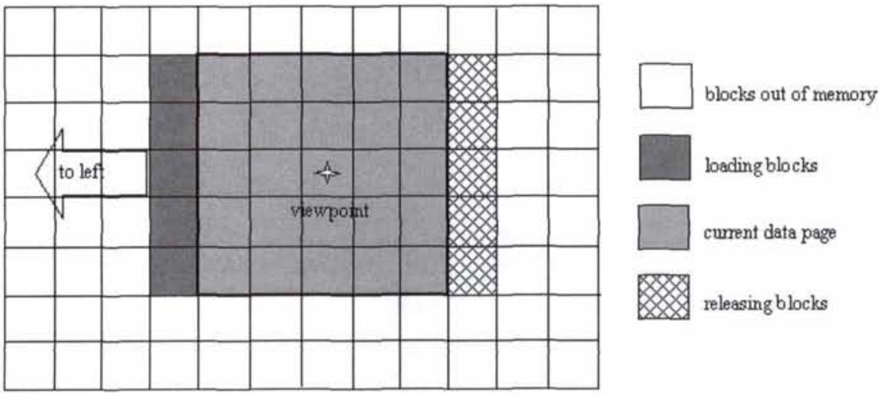
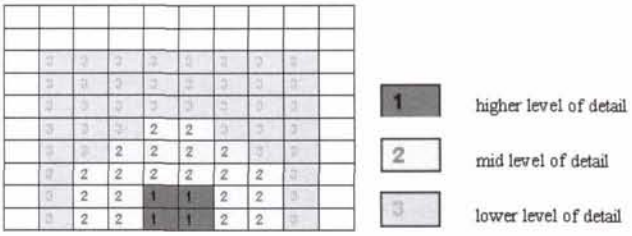


Figure 4. The dynamic data page based on subdivided blocks.

Because it is impossible to avoid the visible loss of quality when the terrain texture images of the same frame are at different resolutions, the resolutions of the orthoimages on all the data page blocks are identical. However, because the DEMs are at different levels of detail, another special process has to be adopted to remove the seam between DEMs when rendering. Fortunately, because the resolution difference of any two adjacent DEMs is determined beforehand, simple but fast approaches can be used to solve the seam problem. The most difficult problem is to generate the LOD models of 3D vector objects, because the rendering of one object involves regular and irregular data types. In fact, all the geometric data of



	3	3	3	3	3	3	3	3	
	3	3	3	3	3	3	3	3	
	3	3	3	3	3	3	3	3	
	3	3	3	2	2	3	3	3	
	3	3	2	2	2	2	3	3	
	3	2	2	2	2	2	2	3	
	3	2	2	1	1	2	2	3	
	3	2	2	1	1	2	2	3	

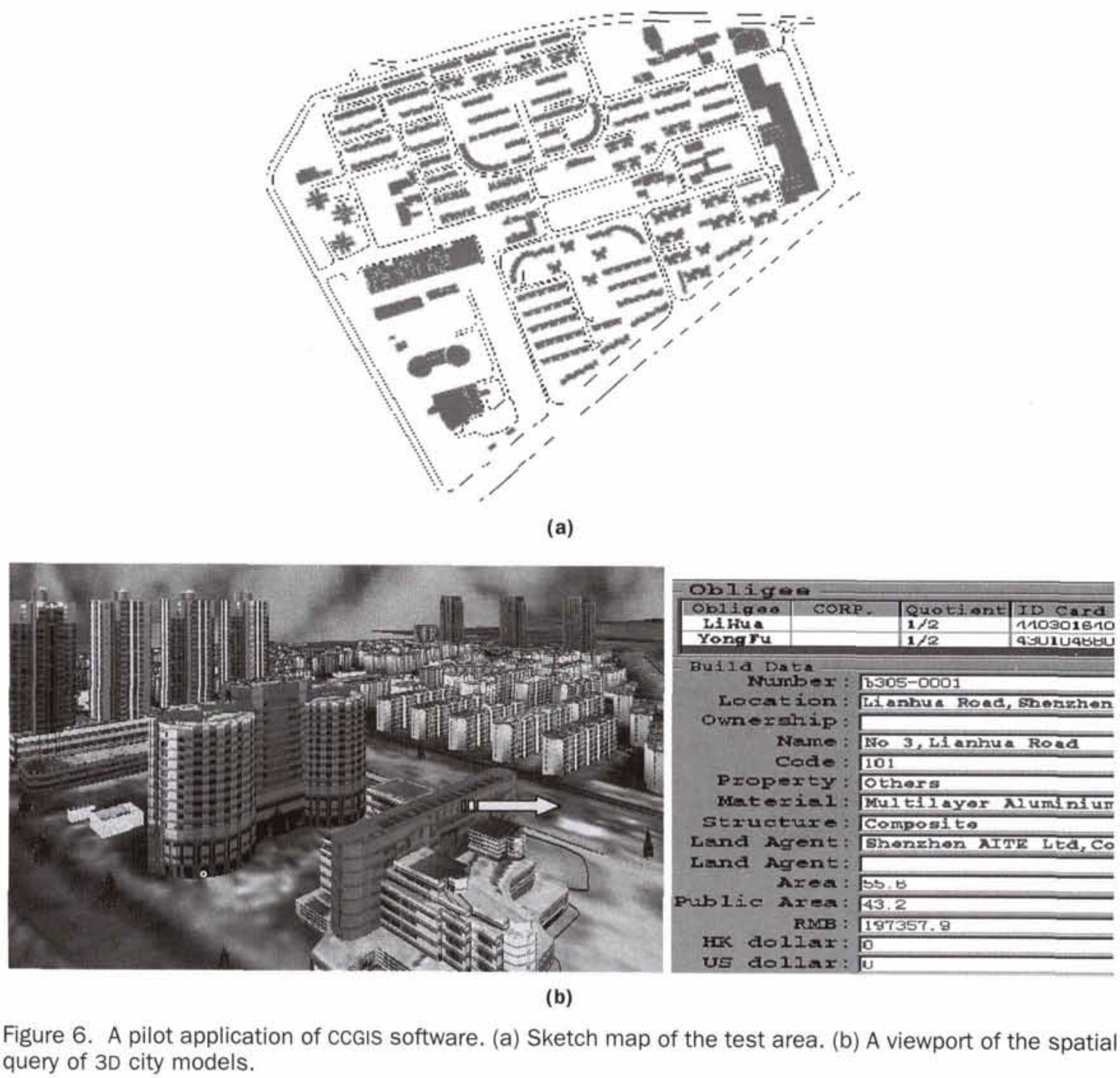
Figure 5. The LOD (level of details) concept at the real-time level.

selected objects have been loaded into the data page, but the texture images are just loaded as necessary. When rendering the data in memory, a special simplification of the geometric shape is processed in real time, i.e., to omit the roof shape of objects at lower levels of detail.

The above special process is useful to keep the high fidelity of visual impact as well as to reduce the data volume. Because each object in the database can be manipulated and displayed independently, thus controlling the visualization dependent on the problem and objects in question, this allows the planner, or the person affected by the plan, to structure the visualization corresponding with his or her viewing habits.

Pilot Applications of CCGIS

Based on the principle described above, the prototype of the CCGIS has been developed using OpenGL and VC++6.0 programming languages, and a pilot application for municipal planning and land information publications has also been implemented. Figure 6 illustrates the user interface of the CCGIS. This test area covers about 54 square kilometers (6 km by 9 km, Figure 6 (b)), and there are a total of 187 buildings. The basic 3D building roof boundary points are collected by the JX-4A DPW interactively; based on these coded roof boundary points and DEM data, the CCGIS automatically generates all the 3D building models in a few minutes, and all the texture images and attributes are related to the objects interactively. Each DEM block consists of 33 by 33 cells and the orthoimage block consists of 256 by 256 pixels, a tile including 16 by 16 blocks. The pyramid DEM databases are designed at multi-resolutions of 2m, 4m, 8m, and 16m, respectively, and the resolutions of orthoimage databases are about one-eighth of that of the DEMs. The total data amount is 315 mega-bytes. The desktop worksta-



(a)

Obligee

Obligee	CORP.	Quotient	ID Card
LiHua		1/2	110301610
YongFu		1/2	430104880

Build Data
Number: b305-0001
Location: Lianhua Road,Shenzhen
Ownership:
Name: No 3,Lianhua Road
Code: 101
Property: Others
Material: Multilayer Aluminiur
Structure: Composite
Land Agent: Shenzhen AITE Ltd,Co
Land Agent:
Area: 55.8
Public Area: 43.2
RMB: 197357.9
HK dollar: 0
US dollar: 0

(b)

Figure 6. A pilot application of CCGIS software. (a) Sketch map of the test area. (b) A viewport of the spatial query of 3D city models.

tion we selected is a general PentiumIII750, its texture memory is 32MB, the main memory is 512MB, and the ORDB is Oracle8.16 Spatial. When we control the number of triangles per frame to be less than 20,000, the frame rate of animation can reach 9.5 per second. In the 3D cityscape as shown in Figure 6b, we can query the information about any building, even any personal estate information in this district. This kind of lifelike query has been admired by most residents and managers. According to this idea, it is planned to construct the CyberCity GIS for the entire city (more than 400,000 buildings over the 326 square-km-area) at 1:2000 scale.

In the meantime, the prototype of the CCGIS also provides other versatile GIS functionality such as SQL query, spatial buffering, query-based selection, etc. Any individual or group of selected GIS objects can be rotated and translated in 3D space, and the object can be hidden in any case. Quick analysis of object-to-object distances, object heights, and sunlight is also available.

Conclusions

Aiming at the requirements of CyberCity development, the CCGIS simply provides an initial solution, based on an object relational database (ORDB) system and OpenGL rendering language. The levels-of-detail (LOD) concepts for integrated databases are considered at both the hierarchical database level and the real-time simplification level, which enables the CCGIS to support large CyberCity construction and dynamic visualization. The pilot application proved that the hierarchical 3D modeling method and data model are significant to the 3D GIS; the real-time visualization of a large CyberCity needs elaborate data organization and a dynamic loading strategy. As a prototype of the CyberCity GIS, in addition to the optimization of software development, the following critical issues have to be further researched:

- A more perfect spatial index approach supporting the LOD concept is the key factor to improve the access performance of a vast database.
- More convenient interactive visualization techniques though the networks are very important to promote the application of the CyberCity GIS to more extensive fields. Our near future work includes expanding the current client/server architecture to the browser/server architecture, i.e., the Web-based CyberCity interactive visualization.
- Rather than demands for visualization of integrated databases, more versatile applications of 3D spatial analysis also need the integration of different types of data. This will require/encourage the expression and automatic generation of a topological relationship between 3D vector objects.

Acknowledgment

This project is supported by the NSFC (No. 40001017), the Fok Ying Tung Education Foundation (No. 71017), and a major state basic research program (No. G1999043801). The authors also appreciate Dr. Zhilin Li very much for his critical comments and helpful suggestions to this paper. The comments from three referees are very much appreciated.

References

Dodge, M., and Bin Jiang, 1998. *Geographical Information Systems for Urban Design: Providing New Tools and Digital Data for Urban Designers*, URL: http://www.casa.ucl.ac.uk/publications/learning_spaces/.

Dollner, J., and K. Hinrichs, 2000. An Object-Oriented Approach for Integrating 3D Visualization Systems and GIS, *Computers & Geosciences*, 26:67–76.

Fritsch, D., 1996. Three-Dimensional Geographic Information Systems—Status and Prospects, *International Archives of Photogrammetry and Remote Sensing*, 31(Part B3):215–221.

———, 1999. Virtual Cities and Landscape Models? What Has Photogrammetry to Offer, *47th Photogrammetric Week* (Dieter Fritsch and Rudi Spiller, editors), Wichmann Verlag, Stuttgart, Germany, pp. 3–14.

Germs, R., G.V. Maren, E. Verbree, and F.W. Jansen, 1999. A Multi-View VR Interface for 3D GIS, *Computer & Graphics*, 23:497–506.

Gong, Jianya, and Deren Li, 2000. Object-Oriented and Integrated Spatial Data Model for Managing Image, DEM, and Vector Data, *Photogrammetric Engineering & Remote Sensing*, 66(5):619–623.

Gruber, M., 1999. Managing Large 3D Urban Databases, *47th Photogrammetric Week* (Dieter Fritsch and Rudi Spiller, editors), Wichmann Verlag, Stuttgart, Germany, pp. 341–349.

Gruber, M., and E. Wilmersdof, 1997. Urban Data Management - A Modern Approach, *Computers, Environment and Urban Systems*, 21(2):147–158.

Gruen, A., and Xinhua Wang, 1999a. CyberCity Modeler, A Tool for Interactive 3-D City Model Generation, *47th Photogrammetric Week* (Dieter Fritsch and Rudi Spiller, editors), Wichmann Verlag, Stuttgart, Germany, pp. 317–327.

———, 1999b. CyberCity Spatial Information System (CC-SIS): A New Concept for the Management of 3D City Models in a Hybrid GIS, *Proceedings of 20th Asian Conference on Remote Sensing*, 22–25 November, Hong Kong, pp. 121–128.

Guo, Wei, 1996. Three-Dimensional Representation of Spatial Object and Topological Relationships, *International Archives of Photogrammetry and Remote Sensing*. 31(Part B3):273–278.

Kofler, M., H. Rehatschek, and M. Gruber, 1996. A Database for a 3D GIS for Urban Environments Supporting Photorealistic Visualization, *International Archives of Photogrammetry and Remote Sensing*, 31(Part B2):198–202.

Li, Deren, Qing Zhu, and Xiafei Li, 2000a. CyberCity: Conception, Technical Supports and Typical Applications, *Geo-Spatial Information Science*, 3(4):1–8.

Li, Qingyaun, Zhongjian Lin, and Chengming Li, 2000b. The Status and Prospect of True 3D GIS, *Science of Surveying and Mapping*, 25(2):47–51 (in Chinese with English abstract).

Li, Zhilin, and Qing Zhu, 2000. *Digital Elevation Model*, Publishing House of Wuhan University, Wuhan, China, 248 p. (in Chinese with English abstract).

Molenaar, M., 1992. A Topology for 3D Vector Maps, *ITC Journal*, (1):25–33.

MultiGen-Paradigm, 2000. *SiteBuilder3D White Paper*, URL: http://www.sitebuilder3D.com.

Pan, Zhigeng, Xiaohu Ma, and Jiaoying Shi, 1998. Overview of Multiple Level of Detail Creation, *Journal of Image and Graphics*, 3(9):754–759 (in Chinese with English abstract).

Pullar, D.V., and M.E. Tidey, 2001. Coupling 3D Visualisation to Qualitative Assessment of Built Environment Designs, *Landscape and Urban Planning*, 55:29–40.

Ranzinger, M., and G. Gleixner, 1997. GIS Datasets for 3D Urban Planning, *Computers, Environment and Urban Systems*, 21(2):159–173.

Wang, Xinhua, and A. Gruen, 2000. A Hybrid GIS for 3-D City Models, *International Archives of Photogrammetry and Remote Sensing*, 33(Part B4):1165–1172.

Zhou, Kun, Zhigeng Pan, and Jiaoying Shi, 2001. A Real-Time Rendering Algorithm Based on Hybrid Multiple Level-of-Detail Methods, *Journal of Software*, 12(1):74–82 (in Chinese).

Zhu, Qing, 1998. Three Dimensional Dynamic Visualization Model, *Journal of Wuhan Technical University of Surveying and Mapping*, 23(2):83–87 (in Chinese with English abstract).

Zhu, Qing, Deren Li, Jianya Gong, and Hanjiang Xiong, 2000. The Integrated Spatial Databases of Geostar, *International Archives of Photogrammetry and Remote Sensing*, 33(Part B4):1243–1246.