Interactive Visualisation of 3D City Models Based on Adaptive Streaming of 3D-GIS Data

Zhenping Zhang
Police Office of Qingdao
Qingdao, P.R. China
police_qd@163.com

Jitao Fang
Shanghai University
Shanghai, P.R. China
fjt801219@126.com

Runhuan Jing
New star company of Ocean University
Qingdao, P.R. China
jingrunhuan@126.com

Abstract—Present an approach to adaptive streaming of complex three-dimensional GIS data over public networks. In the client-server based architecture, the CityServer3D, both geometrical and texture data describing highly detailed three-dimensional city models is streamed to the client. In order to achieve significant reduction of the huge amount of data, take advantage of LOD concepts and a predictive approach to anticipate user navigation: After estimation of future viewing parameters, the server chooses bandwidth-adequate representations of three-dimensional objects and delivers them to the client. By this means, could ensure that only user-relevant data is sent.

Keywords-3D city models; remote rendering; information visualization; adaptive streaming; geodata server; 3D graphics

I. INTRODUCTION

Visualisation and interaction with geospatial data is more and more becoming subject of wide public interest. Besides professional users of GIS applications for planning, statistical evaluation and decision support, private users such as tourists and web surfers gain insight into the multitudinous application areas of spatial data incorporating semantics. Free available applications like Google Earth, NASA World Wind and Microsoft VirtualEarth mark a new era of information visualisation. By interactively navigating through the earth's virtual representation the user is provided with a geo-based interface to information. Suddenly, new links between different data can be established since they now can be located in the actual sense - a giant step towards a user-centred application design.

Another development in the GIS segment is the increasing usage of real 3D data1 for modelling the real world and the data that is related to it. Recent research concentrated on the design and implementation of an interoperable 3D-GIS architecture. Because of these developments more and more three-dimensional spatial information systems are introduced on the market. A generally accessible economical view of this market was missing so far. A market analysis for 3D-GIS showed a growing interest to the topic and attendance of software manufacturers and data providers to invest into the development of solutions like services and software products. The work described in this paper focuses on the streaming component of the 3D city model architecture. Our approach combines streaming techniques comparable to those of systems like Google Earth with accurate and diversified 3D-GIS data. 3D city model clients can connect via public networks and navigate through 3D models at real-time speed.

This is done by sending only data that is directly visible to the avatar in the 3D scene. The server uses an adaptive algorithm in order to estimate future viewpoints, choose between different representations of 3D objects according to the available bandwidth and stream the data to the client.

II. THE FEASIBILITY ANALYSIS FOR ADAPTIVE ALGORITHMS IN INTERACTIVE VISUALISATION OF 3D CITY MODEL

A. Previous Work in the Field of Remote Rendering

The research in the field of remote rendering and performance optimization by using adaptive algorithms takes origin in the early 90s. To that time, even the performance of professional graphics hardware was insufficient when it came to rendering of complex 3D scenes. Funkhouser et al. addressed the problem of virtual walkthroughs within buildings. Their algorithm incorporates visibility determination, reduction of details (by using different LOD) and, most important, a cost-benefit-heuristics: Each object in the virtual scene has a certain cost, which in that case is the rendering time. Determining to what extent an object benefits to the whole scene is more complicated since it involves human perception. The main challenge is to choose the best possible candidate for rendering from a set of representations. In addition, Maciel and Shirley make use of so-called impostors, which are view-dependent texture representations of 3D-objects that are pre-computed and sampled from a discrete number of positions within an object's hemisphere. During the rendering process, a two-pass algorithm is used to choose the probably optimal representation of an object.

Nowadays, common desktop PC hardware allows the visualisation of highly-detailed 3D real-time games such as flight simulators or ego shooters. Therefore, today's graphics hardware is a magnitude faster than professional workstations ten years ago. Through the increasing usage of public networks in the past decade, large-scale distributed systems became reality. Hesina addressed the problem of bandwidth as bottleneck in remote rendering and proposed a network architecture based on the previous work and developed a method for on-demand transmission of 3D-data. The presented approach is a further step in the integration of approved methods and algorithms from previous research with extended focus on real-world applications that consider semantical data. Based on an abstracted data model and an interoperable architecture, we extend the former approaches by decoupling the mentioned methods from specific data or hardware.
B. The objectives of this method

The system design for adaptive streaming of 3D GIS geometries and textures has to fulfill the following objectives:

1. Dynamic transmission of geometries and textures instead of downloading the whole scene allowing users to navigate through large city models in real-time.

2. The system is capable of user movement prediction in order to prefetch objects that are most likely to be seen in near future.

3. Quality of object representation must be chosen dependent on connection bandwidth and stability in a way that the best possible visualisation is guaranteed.

4. Both large terrains and smaller 3D-objects (e.g. houses, trees) must be considered and handled adequately.

5. Semantical data that accompanies the geometries must be preserved and handled appropriately.

6. The mechanism is not depending on hardware specifics, since the approach shall be used in several application contexts and on different Java-based clients.

C. Conditions and Constraints

Since modern graphics hardware is used even in standard desktop PCs, our approach assumes the client to be able to performantly render whatever objects the server sends. Nevertheless, the client can use the initial hand-shake to tell the server not to send objects in very high LOD, but this is mainly used within an certain application context, e.g. in the context of route visualisations it may be not useful to display every building at a high LOD.
what order. This is done by using an adoption of the established cost-benefit-algorithm as proposed by Funkhouser and network customizations described by Teler. Our cost-benefit implementation is explained in the next section.

Finally, the client receives new serialized Java objects in the metamodel format which are transformed into J3D scene graph branches. The client replaces or adds the new geometries dynamically and decides according to his policy (i.e. available memory) whether to maintain or reject potentially existing representations in lower LOD.

B. Metamodel and J3D scene graph

Since 3D-GIS data consists not only of geometrical visualisation data but also of detailed data like statistical information, real estate and utilization data, 3D city makes use of an abstract data model, called metamodel. By doing so, interoperability with numerous formats and services is achieved.

![Image](56x357 to 294x518)

**Figure 3** Example of a metamodels subset. Features possess Models with different properties like LOD.

Internally, all data is represented in the metamodel, which can represent all types of GIS information. Figure 3 shows how spatial data is organized within 3D city model, at the top hierarchy level, a world node is used to group all further elements. Gazetteers are used for spatial aggregation of geological data (e.g. countries, cities, districts) whereas layers group features belonging to the same domain (e.g. all hospitals, public buildings).

Features represent objects of defined position and meaning (e.g. buildings, trees, rivers) and group different models which consist of entities (geometries). The models incorporate the actual spatial data and can be used to store different LOD of the same feature where we use the classification of Gröger et al. Regarding streaming aspects, the models are of interest since they represent the objects that are transmitted to client. The objects contained in the metamodel are converted into a Java3D scene graph for visualisation, but the original metamodel data is still present. This is necessary since we need to maintain the GIS data accompanying the spatial representations. If the client receives new models in higher LOD, it has to export the metamodel data to Java 3D.

C. The database of 3D city model

The data used within system is stored in a spatial database, Oracle 9i with spatial extension, that copes with vast amounts of 3D-data and supports spatial operations directly. Here, a city model, which contains buildings in at least two different persistent LODs, is used.

![Image](314x606 to 553x649)

**Figure 4** Different LODs of a building

An example for a mid-size object is the IGD-building, which is represented through 30 to 5000 polygons (see table 1). In sum, the city model consists of approx. 3.3 million polygons at the highest LOD. In addition to the polygons, textures increase the size of an object dramatically (in our example from table 1, the texture for LOD 2 consumes about 400 k Bytes).

<table>
<thead>
<tr>
<th>Level of Detail</th>
<th>Number of polygons</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>1.5</td>
<td>150</td>
</tr>
<tr>
<td>2</td>
<td>400</td>
</tr>
<tr>
<td>3</td>
<td>5000</td>
</tr>
</tbody>
</table>

IV. THE IMPLEMENTATION AND ESTIMATE OF 3D CITY MODEL SYSTEM

A. Implementation

The cost of transmitting the current object is determined through network bandwidth and the size of its representation. Considerations of client rendering performance during real-time cost-benefit estimation are explicitly excluded. Given a number of samples of measured bandwidth and a representation of an object with a certain size $s(rl)$, we can estimate the cost of streaming $rl$ to the client through calculating the ratio between LOD-dependent size and a weighted sum of current and previous measured samples of bandwidth. We parametrize the cost calculation so that it can be easily adapted to user preferences and connection stability.

B. Benefit estimation

Since the benefit of an object to a scene is matter to subjective human perception, it cannot be calculated and is therefore estimated. The benefit heuristic developed by Funkhouser considers the factors relative size, focus, accuracy, visibility, importance (semantics), motion and hysteresis of visual objects. Most of these properties are dependent on the current view point in the scene and cannot be precalculated. A representation's actual benefit is calculated by using branch&bound algorithms which are used to achieve a suboptimal solution of the NP-complete nature of benefit selection. This method is applied to a discrete number of representations of small objects and terrain tiles as well. For
the implementation of terrain streaming approximation trees for hierarchical grid tiling are currently used. For a further improvement, an integration of smooth LOD techniques as proposed by Hoppe is planned.

C. Feature meta graph

Since some factors for cost-benefit-heuristics are calculated in advance, a meta representation of the scene's object representations, called feature meta graph, is used. The feature meta graph manages precalculated factors of features and their models in an optimized structure, resulting in advantages like fast traversal and comparison of different representations as well as providing rapid access to the concrete data by using references. In addition, this structure is used for querying spatial relations like positions and bounding boxes for visibility culling. An extension of Funkhouser's original approach is the combination with an additional 3D semantic module which is able to determine the importance/semantics of features. This is accomplished by evaluating precise non-geometrical attributive data extracted from cadastral datasets and task-dependent information, e.g. in tourist or planning scenarios.

D. Network communication

An important component of this approach is the network component and since this is the limiting factor (bottleneck), need to minimize overhead resulting through communication between client and server. Due to our server-centric approach, can reduce our communication protocol to a minimum because only small portions of client parameters need to be considered. By (dynamic) periodic calling of the server action, the client updates his viewing parameters. In response, the server performs motion prediction and selection through the cost-benefit-heuristics and streams the serialized objects. The packets contain time stamps that are used for network bandwidth calculation. Optionally, the server measures the roundtrip time of the connection; this can be useful when the network latency is varying significantly during the user's session.

V. CONCLUSIONS

An approach for an adaptive streaming of complex three-dimensional GIS data over the internet is described within this paper. In the used client-server based architecture, the 3D city model, both geometrical and texture data describing highly detailed 3D city models is streamed to the client. In order to achieve significant reduction of the huge amount of data, take advantage of LOD concepts and a predictive approach to anticipate user navigation: After estimation of future viewing parameters, the server chooses bandwidth-adequate representations of 3D-objects and delivers them to the client. By this means, we ensure that only user-relevant data is sent. Future work will be dedicated to verification via usability tests and a improvement of the prediction in order to combine predictions and user's context. Not only have to transfer geometries and texture but also have to provide the user with additional information like attributive data, an approach for delivering this data has to be found. It has to be taken into account that the 3D city supports different media types like movies, textual information or pictures which can be can be assigned to a feature.

ACKNOWLEDGMENT

It is a project Sponsored by projects of Science Research for the Plan of Qingdao under the Grant #20072316jch. The authors are grateful to doctor Jitao Fang for his help in programming. Prof. Xiangqian Ding in addition, proofread the paper and offered valuable suggestions for editing.

REFERENCES