Web-based Visualisation of the Generalised 3D City Models using HTML5 and X3DOM

Bo Mao[1,2], Yifang Ban[2]

[1]Jiangsu Provincial Key Laboratory of E-business, Nanjing University of Finance and Economics, 210003, Nanjing, P. R. China

[2]Geoinformatics, Royal Institute of Technology - KTH, Drottning Kristinas vag 30, SE-100 44 Stockholm, Sweden

{bo.mao, yifang.ban}@abe.kth.se

Abstract: Efficient visualisation of 3D city models in various scales is one of the pivotal techniques to support applications in mobile and Internet environments. In this paper, a framework is proposed to visualise online 3D city models using Internet Browsers without plugins. The City Geography Makeup Language (CityGML) and Extensible 3D (X3D) are used to represent and present the 3D city models. Then, generalisation methods are studied and tailored to create 3D city scenes in block, building and facade levels dynamically; multiple representation structures are designed to preserve the generalisation results in different scales. Finally, the quality of generalised 3D city models is evaluated by the experimental results.

1. Introduction

3D city models can improve many applications such as urban planning, navigation, disaster management, traffic control etc., in various aspects especially in visualisation. More and more applications start to integrate 3D city models into their platforms (Zhang and Zhu 2004), because on one hand, 3D city models can be quickly generated using remote sensing such as Lidar and photogrametry (Brenner et al. 2001). On the other hand, the development of web and mobile devices makes it possible to access 3D city models anywhere (Coors and Zipf 2007). However, compared with the data volume and complexity of 3D city models, the capability of mobile devices and network band are still limited. To improve the visualisation efficiency, generalisation of 3D city models is required.

For 3D city models, generalisation creates models in lower Levels of Detail (LODs) from higher LODs. In CityGML standard, five LODs are defined from internal building to the overall digital elevation model (DEM) to describe the city objects. Considering the complexity of 3D city models, it is essential to generalise the models automatically. Meanwhile, the generalised results of models in different LODs should be organised together for dynamic visualisation, so the multiple representation data structure is required.

The rest of the paper is organised as follows. Related work is introduced in Section 2. Section 3 describes the proposed generalisation methods for different levels. Section 4 presents the experimental results and Section 5 concludes the whole paper.

2. Related work

To visualisation 3D city models with browser, different plugins are development. Lerma *et al.* (2004) create VRML plugin for browser to visualise their virtual museum. Zhou *et al.* (2006) created a backend server to process the parameters from the user, generated 3D models, and sent them back to the VRML-enabled browser for display/navigation. Manferdini and Remondino (2010) developed a 3D city model visualisation methodology based on O3D technologies. Marvie *et al.* (2011) proposed an approach for adaptive streaming of online multiuser virtual worlds, using generic transfer protocols and a unified representation of the worlds based on X3D. However, it is difficult to maintance so many different plugins for both user and developer. Therefore, Behr et al. (2009) proposed X3DOM to show native X3D within an HTML page. Lornet (2011) demonstrated a showcase in the Web3D 2011 conference about the 3D city models of Dijon.

These online applications indicate that visualisation of 3D city models in multi-scale is essential for online presentation. Zooming user interface is used to change the scale of viewed area in order to see more detail or less, and to browse through different areas (Bederson and Hollan 1994). It is important to find methods to generate lower LODs from higher LODs automatically by generalising city objects in block level and building level, because generalisation not only reduces the data volume but also hides the unnecessary details in different scales.

In block level, a number of aggregation methods have been developed for aggregation of 2D cartographic objects and 3D buildings. Bundy et al. (1995) introduced *Direct-merge* operator and *Snap-merge* operator; Anders (2005) proposed an approach for 3D building aggregation based on

projection. Typification is also applied in the block level generalisation. Müller and Wang (1992) made use of mathematical morphology and proposed a solution to the automated generalise one particular class of geographical objects. Sester and Brenner (2000) gave another typification algorithm based on Kohonen Feature nets (Kohonen 1982). Christophe and Ruas (2002) proposed a linear structure detection method based on straight line. Anders (2005) introduced an algorithm to detect the grid structure of 3D building groups.

In building level, Meng and Forberg (2007) summarised generalisation methods that change the representation between different LOD. For the single building model simplification, Mayer (2005) and Forberg (2007) created a scale-space technique partly based on the morphological operators opening and closing to simplify 3D building model. Half space model was used by Kada (2006) to detect the main outline of a building. Then he extended the simplification method to the roof structures using pre-defined roof types (Kada 2007). Zhao et al. (2012) explored the generalisation method for detailed building models in both appearance and internal structure with authentic architectural components. Fan and Meng (2012) proposed a method with three steps to simplife 3D buildings in CityGML.

3. Methodology

In this paper, the X3DOM based online visualisation framework is implemented first by converting the 3D city models from CityGML into X3D files and updating the 3D scenes in the user browser using Websocket. According to earlier experimental results (Mao and Ban, 2011 - add Ref of Cartographica paper), it is difficult for the browser to support the large data volume of 3D city models, which illustrates the necessity of the proposed generalisation methods.

3D generalisation methods are employed to create the multiple representation structure of the city models in different LODs defined in CityGML. Since the LOD0 (2.5D models) can not reflect the full properties of 3D city models and LOD4 data is not as widely acquirable as others for the privacy reasons, this thesis focuses on the generalisation of models in LOD1, LOD2 and LOD3. With the proposed multiple representation structure generated for each LOD, the 3D city models are continuously integrated together. Then the dynamic visualisation strategies are discussed based on the multiple representation structures in different LODs. Finally, the generalisation results are evaluated by comparing the visual similarity

between the original models and the generalised ones. The overall structure of the proposed generalisation framework is given in Figure 1.

	Representation	Generalisation
	X3DOM based dynamic visualisation	
Block level (LOD1)	CityTree representation	Aggregation
Building level (LOD2, LOD3)	Shell model representation	Building selection
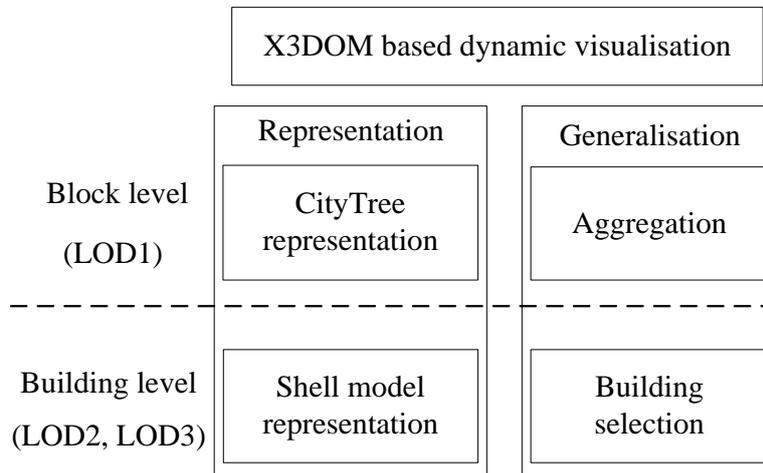
Fig. 1. Generalisation structure for web visualisation of 3D city models

The 3D city model generalisation methods in block level and building level are studied. The multiple representation structures are created in each level to storing the automatically generated generalisation results for dynamic visualisation. In the block level (LOD1), CityTree, a novel tree structure, is proposed to represent the models in different scales by aggregating the nearby buildings. In building level (LOD2 and LOD3), a new LOD of the city object is created by projecting the detailed geometrical structures of the building surfaces such as windows, doors, etc. into its exterior shell. And an index structure is created specified for the street level visualisation.

The test data of the proposed visualisation framework is in CityGML format. The test datasets in CityGML come from the web site CityGML.org, the official homepage of CityGML, and the datasets in KML come from the ViSuCity project (Ban 2008), in which 3D city models are supplied by Blom A.B. in Sweden. They export the data in KML format. The rest of the chapter will discuss these parts of the proposed visualisation framework in details.

3.1 Block level generalisation

In the block level of 3D city models, the number of the objects is huge and the data volume is very large. Generalisation is required not only to reduce the data volume and improve the visualisation efficiency but also to emphasise the interesting/target parts such as the destination in the navigation. In the block level, the overall structure is more important than the details of each object in the visualisation because the distance from the user view point to the city objects is too long to reveal all the details. There are many structures or patterns in the city block level e.g. linear, grid, etc. In this paper, Delaunay Triangulation and Minimum Spanning Tree are used to detect two common structures in the block level, the neighbourhood and linear structures. For the neighbourhood structure, the aggregation operation is applied, and for the linear structure, the typification operation. Since the 3D city models in the block level are widely distributed in a relatively large area, and their distances from the current user view point could be quite different. Therefore, it is necessary to create the multiple representation structures to support the different generalisation in dynamic visualisation.

CityTree (Mao et al. 2011), a multiple representation data structure of the city objects, is created to effectively implement the continuous scaling and dramatically reduce the loading time of 3D models. The CityTree structure is based on a binary tree in which leaf nodes represent the original 3D city objects (here: the building objects) and the other nodes represent the generalisation models of their children; that is, the CityTree is a dendrogram of building objects. In visualisation, selected nodes of the CityTree are shown to the user according to his/her view point. When the view point is changed, the new selected nodes in the CityTree will replace the previous ones. By utilising CityTree, it is possible to have dynamic zoom functionality in real time. When the 3D city models in different LODs are generalised, it is required to first convert the model from higher LODs to lower LODs. Then nearby buildings are aggregated together to simplify the 3D city models.

Figure 2 gives a demo example of the CityTree. Figure 2(a) shows the distribution of the original city building group which contains 5 buildings (1~5). The rectangle areas (A~D) are created by selecting nearby buildings. The CityTree is generated as shown in Figure 2(b). The leaf nodes (1~5) are original objects in city model. The other nodes (A~D) are new generated middle nodes with geometry feature of aggregation of children models.

(a)   (b)
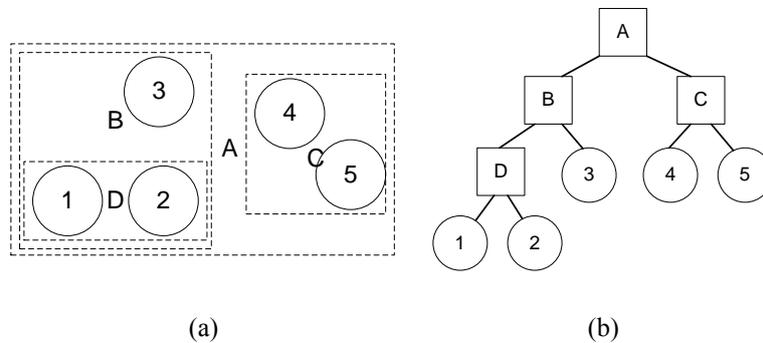
Fig. 2. CityTree model

In the visualisation step, the CityTree nodes are selected based on the user's view point and the features of the node. If the parent node is selected to visualise, all his children will not be loaded which can reduce the computational complexity dramatically.

3.2 Building level generalisation

In many 3D city related applications such as road navigation, urban planning, etc., the indoor details are not necessary and should not be visualised for privacy considerations for most of the city objects. The model appearances are enough for these applications. Therefore, by creating the exterior shell representations of the detailed 3D city objects, not only the visual efficiency is increased the but also the user privacy is protected. Therefore, the shell model is an important bridge linking the models in higher LODs into lower LODs.

To extract the exterior shell of the detailed 3D city models, four steps is required: (1) extract exterior shell of wall elements, (2) project the selected polygons of windows and doors on the corresponding walls, (3) select the polygons of windows and doors whose plane are parallel to the walls where the windows and doors are sealed, and (4) extract exterior shell of roof structure. The details of the algorithm implementation are given in Fan et al. (2011).

For building level visulisation, the viewpoint is usually near to the buildings, such as street level viewing. In these applications, the number of visible buildings at each viewpoint is limited. Therefore, the selection operation in the model generalisation is chosen to simplify the 3D city models.

To detect the visibility of each building, all polygons of the 3D models are projected into 2D according to the current user viewpoint. Then cover-

age relationship between each polygons are analysed based on the distance to their current viewpoint. Only these buildings with visible projections are selected as the generalisation results for the current viewpoint.

Because the proposed visibility analysis and visual importance calculation are quite time consuming tasks, it is not efficient to perform these processes whenever the user viewpoint changed in the dynamic visualisation. Based on the characters of the street view, the limited viewpoint (on the street) and limited view angle (along the street), an index structure is pregenerated to store the visibility and visual importance information at certain viewpoint along the street. These indexes cover the road network at certain interval. In our implementation, the interval is 20 meters that could be changed according to different applications. In dynamic visualisation, the corresponding index is selected for the current user viewpoint, and the generalisation results will be created based on the information of the selected index. There may be some difference in the visible models between the actual viewpoint and its index, but it is not affect the overall visual impression in most cases according to the visual similarity comparison.

Then in the dynamic visualisation stage, the n most visually important buildings will be rendered in the 3D scene, and the number n is decided by the applications or the network condition in the online situation.

4. Case study

4.1 HTML5 based visualisation

To visualise the generalised 3D city models, HTML5 related technologies are implemented. Websocket is selected to transmit the 3D data; Webstorage (2012) is employed to cache the data and X3DOM is applied for visualisation.

Compared with HTTP, Websocket is more suitable to transmit 3D city models through Internet. First, the overhead of Websocket is much less than HTTP. The overhead in Websocket can be reduced to just two bytes, while HTTP contains usually several hundreds to several thousands bytes. Second, the communication in Websocket is complete duplex in which the browser and server can send data to each other as soon as there are is update, while HTTP is half-duplex polling solution that only allows server send the data to user after a request. We select Jetty (2012) to implement Websocket in the server. In user side, data can be received with JavaScript as long as the browser supports Websocket.

To further increase the visualisation speed, 3D data should be cached or preloaded into browsers, which can be implemented by Webstorage (Pier-

ro et al. 2011). According to HTML5 specification, Webstorage defines an API for persistent data storage of key-value pair data in Web clients. Compared with cookie in HTML5, Webstorage is larger in size, can share data from different session, and is more security. The default size of Web Storage is 5MB for a domain and it can be set according to user requirement. To visualise the generalised 3D city models, Webstorage should be applied to improve the visualisation efficiency.

Based on the received 3D city data from server, 3D scenes can be created and visualised through browser with X3DOM. The 3D scenes are updated dynamically according to current viewpoint of the user. The visualisation process is composed by following three steps: checking current viewpoint, loading visible buildings and updating X3D objects. In this paper, buildings are all represented by *IndexedFaceSet* object in X3D as follows:

```
IndexedFaceSet : X3DComposedGeometryNode {
    SFNode  [in,out] coord       NULL [X3DCoordinateNode]
    MFInt32 []       coordIndex  []   [0,∞) or -1
}
```

The *coord* field specifies Coordinate and the *coordIndex* defines the polygons. The JavaScript code to generate PointSet in X3D is listed as follows.

```
var s = document.createElement('Shape');
var b = document.createElement('IndexedFaceSet');
b.setAttribute("coordIndex", coordindex);
var crd = document.createElement('Coordinate');
crd.setAttribute("point", coord);
b.appendChild(crd);
s.appendChild(b);
var ot = document.getElementById('x3d_root');
ot.appendChild(s);
```

The code will generate a segment of X3D file in the web page and append it to the X3D root node as follows:

```
<x3d_root>
    …
    <Shape>
     <IndexedFaceSet coordIndex='…'>
         <Coordinate point='…'/>
     </PointSet>
    </Shape>
</x3d_root>
```

4.2 Generalised model visualisation

In this paper, generalisation in city block level is implemented. With the proposed CityTree structure, fewer buildings are visualised and the performances are improved. Figure 5 shows the results of dynamic visualisation of CityTree. The 3D city models are come from CityGML.org and show a real area around Leverkusen in Germany. It can be seen that based on the proposed CityTree structure, the buildings far away from the user view point are more generalised than the nearby ones.
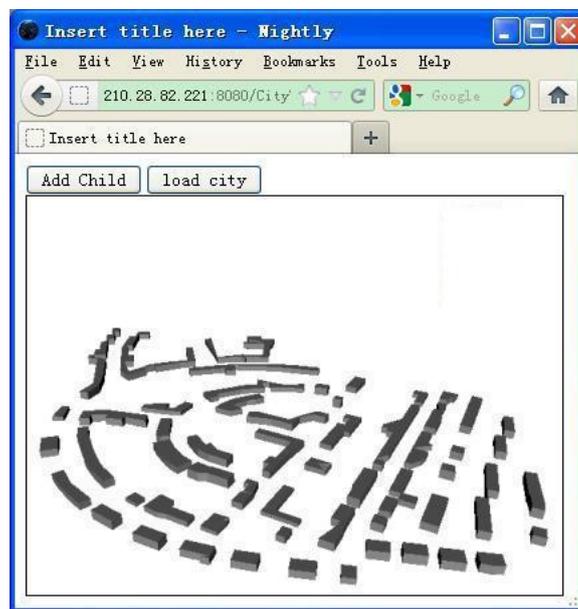
(a) Near distance
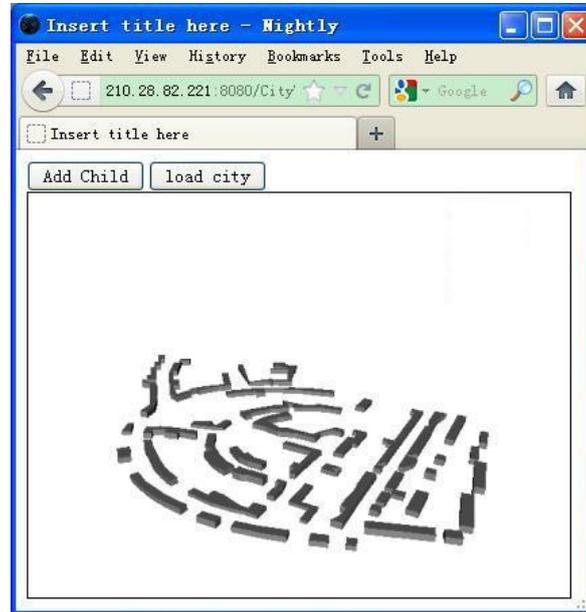
(b) Far distance

Fig. 5. Visualisation of CityTree in different viewpoints

5. Conclusions

This paper focuses on the visualisation and generalisation of 3D city models in multiple scales. An online visualisation framework is proposed based on CityGML and X3D to represent and present the 3D city models respectively. The experimental results indicate that the proposed framework can be implemented in the mainstream web browsers with HTML5 support. It is difficult, however, to support the detailed 3D city models with large data volume. Therefore, generalisation methods and multiple representation structures in different scales (block, building, and facade) are proposed to improve the dynamic visualisation efficiency. The results showed that the proposed generalisation methods in different levels can reduce the data volume of 3D city models and preserve the visual similarity from the original ones, which is essential for online visualisation of 3D city models. In the future, the framework will be tested in the mobile devices.

Acknowledgements

References

Anders K. (2005): Level of detail generation of 3D building groups by aggregation and typification, in Proceedings of XXII International Cartographic Conference 2005, A Coruna, Spain.

Ban Y., (2008): ViSuCity-A Visual Sustainable City Planning Tool: http://www.infra.kth.se/GIS/research/projects/ViSuCity/, (accessed 2011-08-10).

Bederson B., Hollan D., (1994): Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. In Proc. of User Interface and Software Technology (UIST 94) ACM Press, 17-2.

Behr, J., Eschler P., Jung Y. and Zöllner M., 2009. X3DOM: a DOM-based HTML5/X3D integration model, Web3D '09, Darmstadt, Germany

Brenner, C., Haala, N., and Fritsch, D. (2001). Towards fully automated 3D city model generation. In: Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images III.

Bundy L., Jones B., and Furse E. (1995): Holistic Generalisation of Large-Scale Cartographic Data. In: Müller, J. C., Lagrange, J. P., and Weibel, R. (Eds.), GIS and Generalisation, Gisdata 1, Taylor & Francis, pp. 106-119.

Christophe S. and Ruas A. (2002): Detecting Building Alignments for Generalisation Purposes, in International Symposium on Spatial Data Handling, Ottawa, Canada. pp. 419-432.

Coors, V. and Zipf, A. (2007) MoNa 3D-Mobile Navigation using 3D City Models. LBS and Telecartography 2007. Hongkong.

Fan H., Mao B., Shen J., Meng L., (2011). Shell model representation as a substitute of LOD3 for 3D modeling in CityGML, Geo-spatial Information Science, Vol. 14, No. 2, pp. 79-84.

Fan H., Meng L., 2012, A three-step approach of simplifying 3D buildings modeled by CityGML, International Journal of Geographic Information Science, DOI: 10.1080/13658816.2011.625947, 1-18

Forberg, A., 2007. Generalization of 3D building data based on a scale-space approach. ISPRS Journal of Photogrammetry and Remote Sensing, 62(2): pp. 104-111.

Jetty 2012, http://jetty.codehaus.org/jetty/ (accessed 2012-03-01)

Kada, M., 2006. 3D Building Generalization based on Half-Space Modeling. In:Proceedings of the ISPRS Workshop on Multiple Representation and Interoperability of Spatial Data, Hannover.

Kada, M., 2007. Generalisation of 3D Building Models by Cell Decomposition and Primitive Instancing. In: Proceedings of the Joint ISPRS Workshop on"Visualization and Exploration of Geospatial Data", Stuttgart, Germany.

Kohonen T., (1982): Self-organized formation of topologically correct feature maps. Biological Cybernetics, 43:59-69.

Lerma J., Vidal J., Portalés C., (2004): Three-Dimensional city model visualisation for real-time guided museum tours, The Photogrammetric Record, Volume 19, Issue 108, pages 360-374.

Lornet F., (2011): Dijon 3D by LoRNET-DESIGN, Web3D 2011 demo showcase, Paris, June 21th, http://www.lornet-design.net/wp-content/uploads/2011/07/DIJON 3D by LORNET-DESIGN.pdf (accessed 2011-08-11).

Mao B., Ban Y., Harrie L., (2011). A Multiple Representation Data Structure for Dynamic Visualisation of Generalised 3D City Models, ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 66, issue 2, pp. 198-208.

Manferdini A. M. and Remondino F., (2010): Reality-based 3D modeling, segmentation and web-based visualization. In Proceedings of the Third international conference on Digital heritage (EuroMed'10), Marinos Ioannides, Dieter Fellner, Andreas Georgopoulos, and Diofantos G. Hadjimitsis (Eds.). Springer-Verlag, Berlin, Heidelberg, 110-124.

Marvie J., Gautron P., Lecocq P., Mocquard O., Gérard F., (2011) : Streaming and synchronization of multi-user worlds through HTTP/1.1, In Proceedings of the 16th International Conference on 3D Web Technology (Web3D '11).

Mayer, H., (2005). Scale-spaces for generalization of 3D buildings. In: International Journal of Geographical Information Science. Vol. 19, No. 8-9, September-October 2005, pp. 975-997.

Meng L. and Forberg A. (2007): 3D Building generalisation, In: W. Mackaness, Ruas A., and Sarjakoski L. T. (eds.), Generalisation of Geographic Information: Cartographic Modelling and Applications, Elsevier, pp. 211-232.

Müller, J. and Wang Z., (1992): Area-patch generalization: A competitive approach. The Cartographic Journal, 29(2):137–144.

Pierro, G., Cavallari, F., Guida, S., Innocente, V., (2011) Fast access to the CMS detector condition data employing HTML5 technologies, J. Phys.: Conf. Ser. 331 042019

Sester M. and Brenner C. (2000): Typification based on Kohonen Feature Nets, in Presented in the 1st International Conference on Geographic Information Science (GIScience 2000) Savannah, Georgia, USA.

Webstorage, (2012), http://dev.w3.org/html5/webstorage/ (accessed 2012-03-29)

Zhang X., Zhu Q., (2004). Applications of 3D City Models Based Spatial Analysis to Urban Design, on line at http://www.isprs.org/proceedings/xxxv/congress/comm2/papers/148.pdf

Zhao J., Zhu Q., Du Z., Feng T., Zhang Y., (2012). Mathematical morphology-based generalization of complex 3D building models incorporating semantic

relationships, ISPRS Journal of Photogrammetry and Remote Sensing, Volume 68, March 2012, pp. 95-111.

Zhou G., Tan Z., Cen M., Li C., (2006): Customizing Visualization in Three-Dimensional Urban GIS via Web-Based Interaction. Journal of Urban Planning and Development, 132(2), 97-103.