View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by UPCommons

GENERATING AND MANAGING VIRTUAL CITY MODELS WITH THE CITYGRID SYSTEM

MARTIN KERSCHNER
Günter Sükar
GEODATA IT GMBH
Wien - Austria
www.geodata.at www.citygrid.at

Introduction

The generation of 3D city models has been a topic of interest for city management and researchers in recent years. Several models of urban areas have been created. They vary in its content (building models, street furniture, and vegetation), richness of geometric details (from simple block models to fully detailed building models), and in existence of texture. This paper presents a system for generating and updating large city models. The system CITYGRID offers several modules for the whole workflow of building model generation und maintenance for entire cities.

Up to now the emphasis in building modeling lies mainly in a detailed roof reconstruction. Research groups try to find algorithms for automatically detecting buildings and reconstructing roofs using aerial photographs or airborne laser data or both (*Rottensteiner et al., 2004; Brenner, 2000*). Automatic methods deliver promising results. However, the completeness of these results is still too low for automatically generating detailed building models for a whole city. Therefore, software assisting an operator optimally in creating building models, is necessary. The CITYGRID Modeler which is described in this paper is designed for that purpose.

First, the shape of buildings is reconstructed and modeled geometrically. Afterwards photorealistic texture is assigned to the faces. Roof texture can easily be generated from aerial photographs. Photorealistic texture for facades can be added in the same way as roof texture. However, data collection for façade textures is very expensive. Therefore often only a few building models (landmarks) are completed with façade texture. On the other hand customers argue that realistic visualizations need texture on facades. Therefore, the system CITYGRID offers a multi-sensor platform mounted on a van called CITYGRID Scanner. It is designed for efficiently collecting both façade photographs and 3D data needed for detailed façade modeling and orthophoto generation.

Some of the larger cities have started experiments creating city models, although a main problem is not solved yet by most commercial software systems. It is the key issue for city surveying offices: how to update a city model after its generation. The CITYGRID Manager is designed to manage building models over a long period of time in different versions. The core is a relational database holding the data relevant for building models.

The CITYGRID Planner and Explorer complete the system for utilizing a 3D city model for urban planning and visualization purposes.

Figure 1: Textured building models generated by CITYGRID.

Data Management

All modules of the system CITYGRID make use of a common relational database (e.g. Oracle). Various data structures for building models have been proposed. We use a wire-frame representation as the core representation in the database (DB). From this wire-frame model, a boundary representation (B-Rep) can always be derived automatically.

There are some arguments for storing mainly the wire-frame model in the DB:
It is suited best for a relational representation (with points and lines).
The faces of the building can be derived automatically.
Updating a building model is reduced to updating the stored points and lines.

The last reason is the most important for the municipality responsible for a 3D city model. Cities rapidly change. Buildings are teared down, new ones are built. Once a city offers a 3D model, the problem of how to hold the model up to date must be solved. The city of Vienna plans to manage some 300.000 buildings with the system CITYGRID (*Dorffner, Zöchling, 2004*).

For efficiency reasons not only the vertices and edges of a wire-frame representation but also the derived faces are stored in the DB. Furthermore texture coordinates are attached to the corners of these faces. This concept allows for fast loading of textured building faces.

Buildings are organized in a structure of geometric, logical and administrative groups (cf. figure 2). Each building complex (a building or building part with a closed outline) is stored as individual “Object”. All objects belonging together from an administrative point of view (e.g. the main building together with its garage) are represented as one “Unit” which is saved with a unique ID. This unit-ID – e.g. a building code which has been given by the municipality – allows for integrating building related data with GIS. In this way, independent databases can be combined. Several “Units” can be grouped logically to so called “Models”. This enables the user to access easily several districts or building blocks (e.g. “Whole City”, “Inner City”, or “Main Street”). If street names are available, they can be assigned to the according building-models, too.

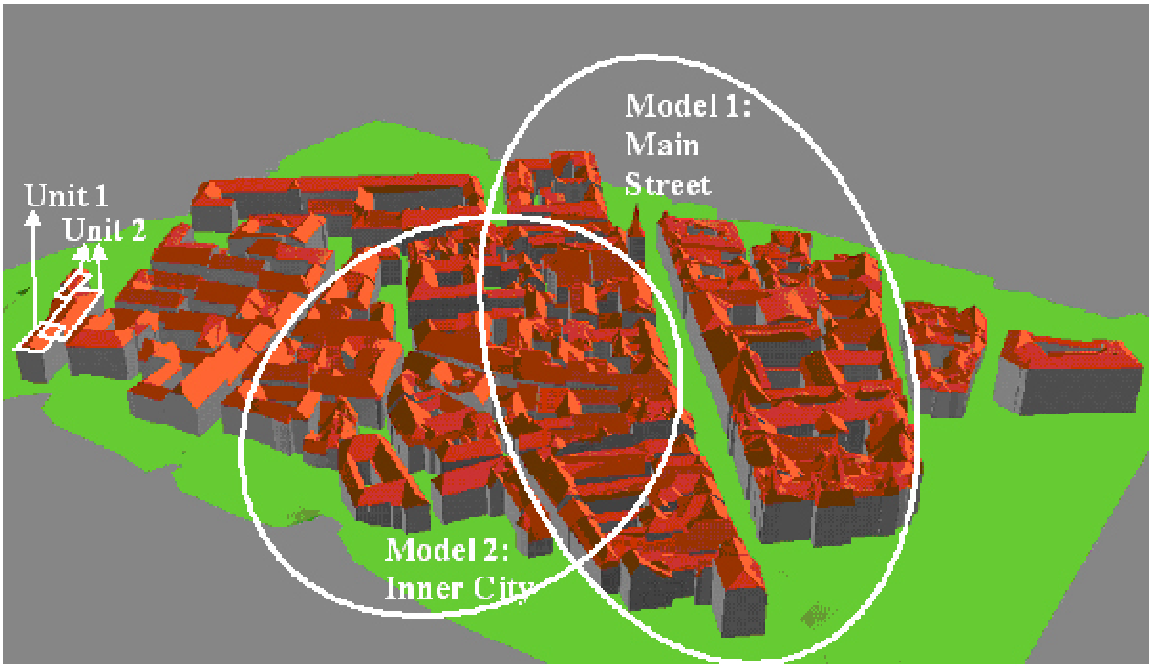

Figure 2: Units contain one or more building complexes. Models group units logically.

Data exchange is provided via files in XML format according to a GML3 application scheme (Geography Markup Language proposed by the Open GIS Consortium; *OGC, 2004*).

CITYGRID Modeler

The CITYGRID Modeler is the software for creating and editing building models. As a plugin in Autodesk VIZ or MAX it is embedded in a powerful software for 3D-modeling and visualization. The key paradigm for generating the building's shape is: Provide the wire-frame model and let the system derive a boundary representation (B-rep).

Generating the roof model

Roof lines are usually derived by means of photogrammetric measurements or airborne laser-scanning. Importing these lines in different layers into the database immediately generates the 3D model of the roof. Each roof is represented by its outline (the outer eave) and (if available) further roof edges: inner eaves, ridge lines or other roof lines (cf. figure 3).
A sophisticated triangulation algorithm is used to derive the roof shape automatically using the wire-frame model as constraint edges in the triangulation (cf. figure 4). Even if the wire-frame model is not complete, often the correct roof faces are immediately generated. Otherwise, the operator can provide additional constraint edges.

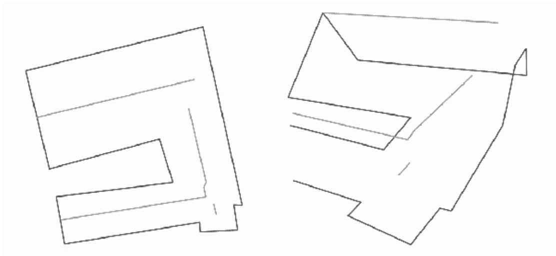

Figure 3: The ground view and a perspective view of a roof
(outline in dark gray, ridge lines in light gray).

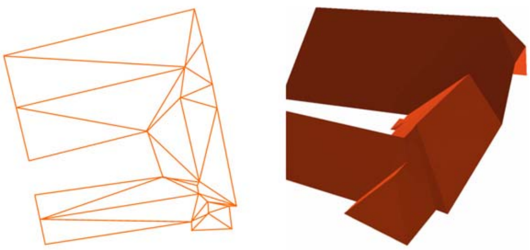

Figure 4: The triangulated roof.

Although the triangulation algorithm works primarily two dimensionally, it allows for break lines on the roof as well. Thus vertical roof faces can be generated as shown in figure 5.

The CITYGRID Modeler is used to correct the data provided by photogrammetric measurements or automatic extraction algorithms. Typical editing actions include moving lines into the proper layers, close topological connections, insert further roof edges, etc. The aim is to edit lines until the derived roof faces are correct.

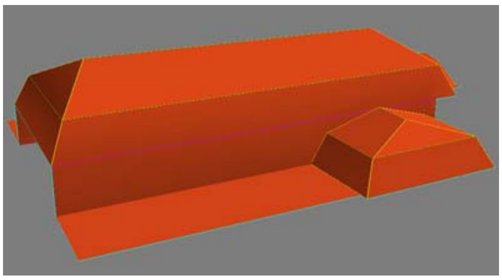

Figure 5: Break lines allow for vertical roof faces.

Generating facades and protrusions

After creating the roof model, the facades are generated in order to derive a complete building model. For a coarse approximation, the roof outlines are extruded to the ground and intersected with a digital terrain model (cf. figure 6).

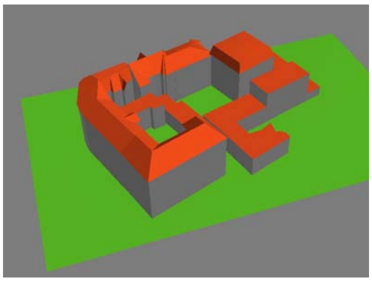

Figure 6: Facades are generated by extruding the building outline.

If the data are available, the protrusion can be modeled, too (figure 7). In that case, the roof outline and the wall ground lines do not coincide.

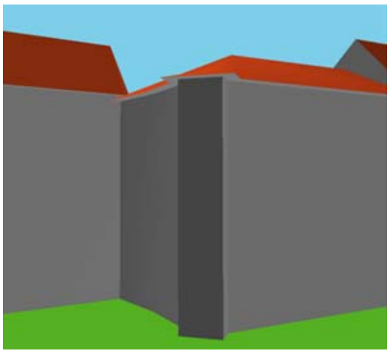

Figure 7: Building with protrusion.

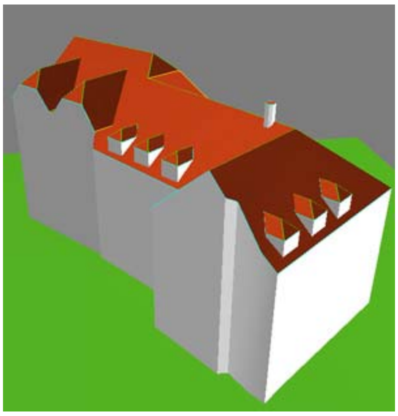

Figure 8: Dormer windows and chimney in a higher LoD.

Details on the roof (dormer windows, chimneys) or on the façade (balconies) are modeled as separate elements and are assigned to a higher level of detail (LoD). Figure 8 shows a roof with several dormers each of which has its own roof and façade.

Adding texture

At this stage building modeling often ends. The building model is geometrically correct. For generating realistic visualizations, however, it is necessary to stick texture onto it. The roof is textured automatically by importing the aerial photograph together with its orientation parameters into the database. The same could be done for the facades, but collecting several façade photos is quite an effort. Deriving all orientation parameters manually would not be affordable. The CITYGRID Modeler offers automated procedures for deriving all these data (see later). For more or less planar facades, a simple projective transformation of façade photos is usually enough. For that purpose, four points of the façade polygon have to be identified in the perspective photograph, which is shown in figure 9.

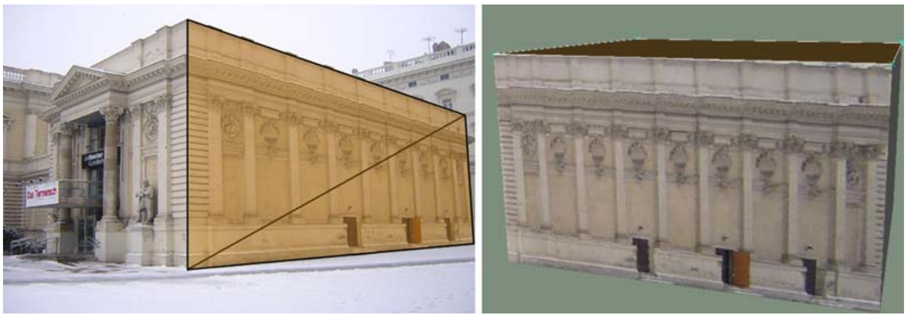

Figure 9: Quick texturing of facades by simple projective transformation.

Updating building models

The previous chapter showed how the building model is generated automatically based on a wire-frame model. The great benefit of this method is the concept for updating the model periodically. For updating, it is sufficient to check (and correct) the wire-frame model. For that purpose the wire-frame model can be displayed together with the aerial photographs (e.g. in a photogrammetric workstation with stereo view).

The CITYGRID system allows for managing the complete history of building models. Several versions are stored in a database. Any historic version can be loaded and compared to the actual version.

CITYGRID Scanner

The CITYGRID Scanner allows creating an efficient and multi-dimensional image of a city seen from the road. With such a system, all data necessary for creating detailed façade models and façade texture can be efficiently collected.

A hybrid measuring system is mounted on a vehicle. It consists of a GPS receiver, a laser scanner and several digital cameras. All sensors are compatibly calibrated to each other and deliver time-synchronized recording.

Figure 10: Multi-sensor system for façade documentation.

The CITYGRID Scanner can be used in two different modes. In the "dynamic mode" it goes along the street (not faster than 5 km/h) and records the façades passing by. A vehicle-own odometer actuates the cameras in such a way that each part of the façade is recorded in a minimum of 5 recordings. Simultaneously with each photographic recording the laser scanner scans a horizontal "line" along the façade. The GPS-receiver allows for a rough positioning to be used for the record administration in a GIS. For record interpretation a matching algorithm has been developed by the VRVis Research Centre (*Karner et al., 2003*). The method uses horizontal and vertical structures of façades for the automatic orientation of the sensors. The result of this process is a true orthophoto of the façade without any distortions or dislocations, which can be used for texturising the city model.

In the „stop and go" mode the CITYGRID Scanner records urban topographic data. The laser-scanner scans a complete 360° panorama. Additionally, according photographs are taken. With this data it is possible to recognize and localize the objects in the road area by means of post-processing.

CITYGRID Planner, Explorer

The CITYGRID Planner allows for creating visualizations integrating photorealistic building models of the existing buildings with digital architectural models of a planned project. Several different planning scenarios can be visualized three-dimensionally in relation to the current environment and can be evaluated by the city-planning point of view (cf. figure 11).

The CITYGRID Explorer is a viewer optimized for the fluent, interactive navigation in very large city-models and is based on the automatic pre-selection of buildings which are within view (*Hesina et al., 2003*). In figure 12, some building models are shown in combination with a vegetation point cloud as derived directly from the laser scanner.

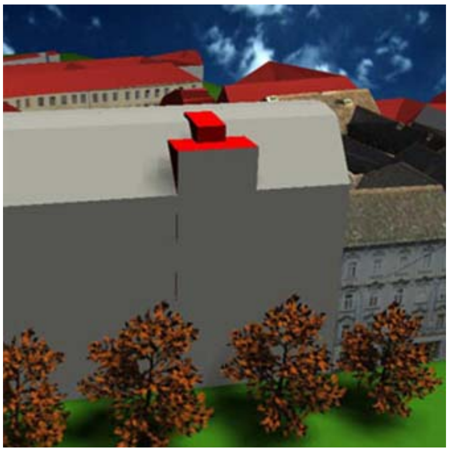

Figure 11: Integrated visualisation of planned and existing buildings.

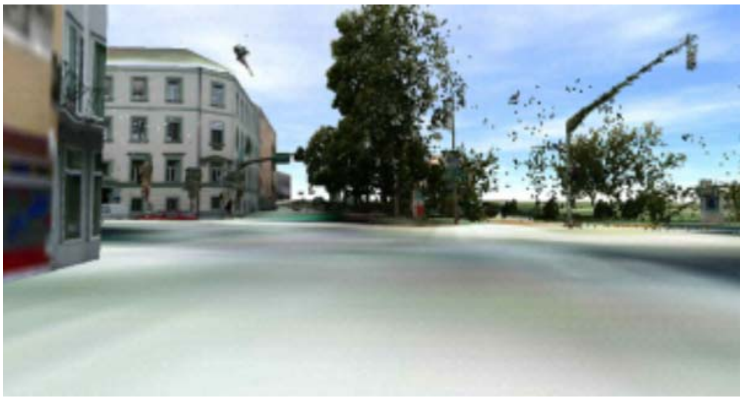

Figure 12: Interactive walk through a city model

References

Brenner C. (2000): Dreidimensionale Gebäuderekonstruktion aus digitalen Oberflächen-modellen und Grundrissen. PhD thesis, Universität Stuttgart, Institut für Photogrammetrie, Deutsche Geodätische Kommission, C 530.

Dorffner L., Zöchling A. (2004): Das 3D Stadtmodell von Wien – Grundlage für Planungsaufgaben und Visualisierungen. In: Strobl J., Blaschke T., Griesebner G. (Hrsg.): Angewandte Geoinformatik 2004. Beiträge zum 16. AGIT-Symposium Salzburg. Wichmann Verlag, Heidelberg, pp. 90-99.

Hesina G., Tobler R.F. (2003): Secure and Fast Urban Visualization. Lecture within the CORP 2003, Vienna University of Technology, 25 Feb. until 1 Mar., 2003. (http://www.corp.at)

Holzer J., Forkert G. (2004): Effiziente Erzeugung von 3D Stadtmodellen aus vorhandenen Vermessungsdaten. Lecture within the CORP 2004, Vienna University of Technology. (http://www.corp.at)

Karner K., Klaus A., Bauer J., Zach C. (2003): MetropoGIS: A City Modeling System Lecture within the CORP 2003, Vienna University of Technology, 25 Feb. until 1 Mar., 2003. (http://www.corp.at)

OGC (2004): Open GIS Consortium, Geography Markup Language, http://www.opengis.org.

Rottensteiner F., Trinder J., Clode S., Kubik K. (2004): Fusing airborne laser scanner data and aerial imagery for the automatic extraction of buildings in densely built-up areas. In: International Archives of Photogrammetry and Remote Sensing, Vol. XX, Part B3, pp. 512-517.