Proceedings of East-West-Vision 2002, Graz, Austria, pp. 99-108, ISBN 3-85403-163-7

ON THE COMPLEXITY OF WEB-BASED PRESENTATIONS OF LARGE URBAN SCENES

Jiri Zara [1])

Abstract

The paper analyses requirements on virtual urban scenes presented on the web. Basic principles and methods for data preparation and processing are introduced. Several useful and practical approaches are shown on examples taken from one real implementation of a virtual city – the Virtual Old Prague project. The paper explains why complex virtual cities are still missing on the web and shows ways how to extend already existing web sites by three-dimensional urban objects.

1. Introduction

A web presentation of existing urban scenes (cities) seems to be currently a challenging task for many researchers and practical programmers. Indeed, every bigger city in the world operates its own web pages with 2D city layout (maps), photographs of important places and buildings, guides for visitors, and other useful information. Several web sites contain 3D panoramic views in QuickTime VR format. Some other offer single virtual models of selected historical or modern buildings, typically in VRML format. Only a few cities (like Paris [16]) have created larger urban areas offering a walk-through a virtual space bigger than one street or a square. Huge, web-based, interactive three-dimensional presentations of existing urban environments with high informative value are still missing.

This paper presents a list of typical requirements on virtual cities. It is shown that the virtual city is not only a collection of 3D models, but it should be a complex information database matching up to a wide range of visitors' needs. The optimal model of a virtual city should be a common framework for different kind of information, where the virtual reality (or 3D graphics) is just one part of them. We consider a virtual city as a big crossroad of hyper linked information in form of text, images, multimedia, and virtual reality. The last topic is the most promising but also the most difficult part, thus we discuss it here in more details.

Presented problems and possible ways to their solutions are documented by examples taken from the Virtual Old Prague project [17]. The first version of this application has been finished in 2001 (see Fig. 1). It is considered as the first running complex virtual city presentation on the web allowing continuous walking through a model stored in a remote database. The data are progressively transferred from the server according to a visitor's position. A model is theoretically unbounded, i.e. it is scaleable in terms of amount of data describing the city, speed of the network, and rendering performance [21].

---

[1] Department of Computer Science & Engineering, Czech Technical University in Prague, Karlovo nam. 13, 121 35, Prague 2, Czech Republic, <zara@fel.cvut.cz>.

Figure 1: A screenshot from the web application. Different kinds of information help users to orient in a city and get a feeling of a real visitor: 2D map and configuration/help tools on the top, hypertext and image guide on the right and the 3D view on the left bottom.

The structure of the paper is as follows. Section 2 analyses virtual cities both from the business and technical point of view. Section 3 deals with data acquisition for 3D virtual models. The organization of data with respect to the web presentation is introduced in Section 4. Problems and features specific to the web rendering are discussed in Section 5. Section 6 concludes the paper and sketches issues to be solved.

2. Richness of Virtual Cities

A typical web application serves at least for two different purposes: users are searching for useful information; providers (web site owners) attract people from business reasons. Both sides of the game must be satisfied otherwise the application fails. This simple rule is extremely important for the case of three-dimensional virtual cities presented in the web. Most of existing virtual models provide only fun for the users without serious business intention behind it, e.g. multi-user virtual Paris [16]. Some presentations have been developed just for research purposes to show that certain methods are efficient and usable, e.g. Virtual Old Prague [17]. Real commercial goals that are a guarantee for long-term running and successful city presentations are missing.

The following subsections list various requirements to the functionality of virtual cities with respect to both providers' and users' needs. The last subsection introduces technical concepts that are specific to advanced web based applications.

2.1. Users' Requirements

The users' requirements can be as wide-ranging as the community of Internet users is. Nevertheless several needs play the most important role:

- Searching for specific places of interest (hotels, shops, offices), looking for their real appearance and a position in a city/map, visiting their official web sites for detailed examination.
- General tourist tour with the aim to visit places virtually before real traveling to the city. Learning more about historical, cultural, architectural and other relationships.
- Path preparation with the aim to arrange/optimize traveling within real city. Utilization of public transportation information.

Some of those issues can be fulfilled using textual information only, while others can take advantage of three-dimensional virtual environment.

2.2. Providers' Requirements

Organizations interested in presentation of virtual cities can be both public and private. Government, city council, museums, and non-profit cultural bodies are examples of the first group of providers. Private enterprises include travel agencies, hotels, companies having residence in a city, shops and supermarkets, etc. All those parties may want to attract Internet visitors using appealing, informatively rich and interactive virtual city on the web. Their requests are:

- Integrated search machine providing results both in form of hyperlinks to HTML pages and "hyperjumps" to virtual space. Advanced searching should prepare an animated 2D/3D tour from arbitrary place to specific target(s).
- Well-defined interface between a city presentation and individual web sites outside the system. Two-way interconnection should allow for leaving virtual space and visiting standard HTML pages as well as natural return to the last visited position (viewpoint) in a 3D space.
- Advertisements and other company specific information integrated into a model and a presentation. Virtual houses/shops should contain highlighted and animated signs possibly with further interactive features. Those features should be customizable.

2.3. Technical Issues

Web applications represent a specific distributed system based on the client-server approach. Due to the nature of the Internet, the limitations are given:

- Relatively low speed of data transfer between a server and a client constrains the amount of data
- Web browsing is considered as an interactive work, thus the system response should not exceed several seconds. Data have to be packed into small packages and displayed as soon as possible.
- Since the server serves concurrently for hundreds of clients, it cannot perform any demanding computations, but send requested data only. Programs and interactions are performed on

clients' side. Programs (scripts) are to be downloaded from the server at the beginning of the session. They can be dynamically updated later. Security limitations can play certain role, too.
- A client-server interaction is limited to a couple of state information within the http protocol. The web server usually does not hold a state of client(s), thus the client-server binding is loose – the client works mostly independently on the server and asks for new data rarely.

While the characteristics introduced above are valid for all web applications, virtual city presentations have to take into account even more issues:

- Presentation of 3D data requires special viewers/browsers. They must be either installed in advance or downloaded at the initial stage of the session. The first approach is not very popular among users; the later causes a big delay before the first 3D view is displayed on a screen.
- Users are familiar with the "click-and-go" paradigm of the web content. They expect the same behavior when switching between HTML part and other components of a virtual city. Some analogies are already in use: the <A> tag in HTML corresponds with Anchor node in VRML. For multimedia components (movies, sounds), a use of regions of interest is still very rare. Also the synchronization is not well standardized – specifications like SMIL or MPEG-21 are still under development.

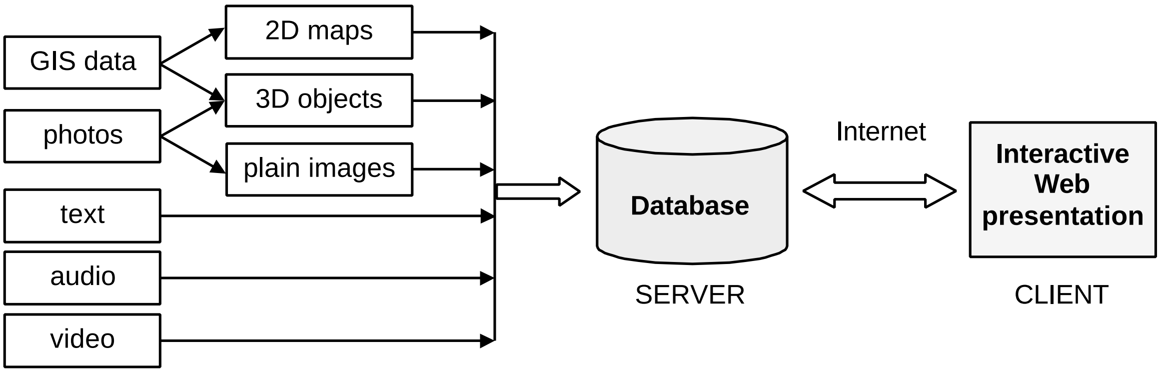

Figure 2: Different kinds of data managed by a virtual city application.

A virtual city application is a complex system incorporating a variety of pre-processed data, storing them in a database and presenting them to clients using special programs and viewers in a client computer (see Fig. 2). A design and a maintenance of a virtual city requires cooperation of people from many fields – computer vision (3D reconstruction), computer graphics (modeling and rendering), databases, networks, multimedia and web applications.

3. Creation of Urban Models

This section describes typical approaches for preparation of 3D city model. A basic element here is a building that can be represented either very simply by textured façade(s) only or as a complex geometrical object with many parts/details. Due to the web limitations, houses should be as simple as possible. The simplicity influences not only a time for data transfer but also a rendering complexity. A virtual city system requires a large volume of data describing photometric (i.e. texture and perhaps reflectance of walls and roofs) and geometric (i.e. shape of individual buildings and position of buildings with respect to a single global coordinate system) properties of the scene. This information must be explicit (an unorganized collection of photographs contains it but only implicitly), i.e. the output of the data acquisition step must already be a 3-dimensional model.

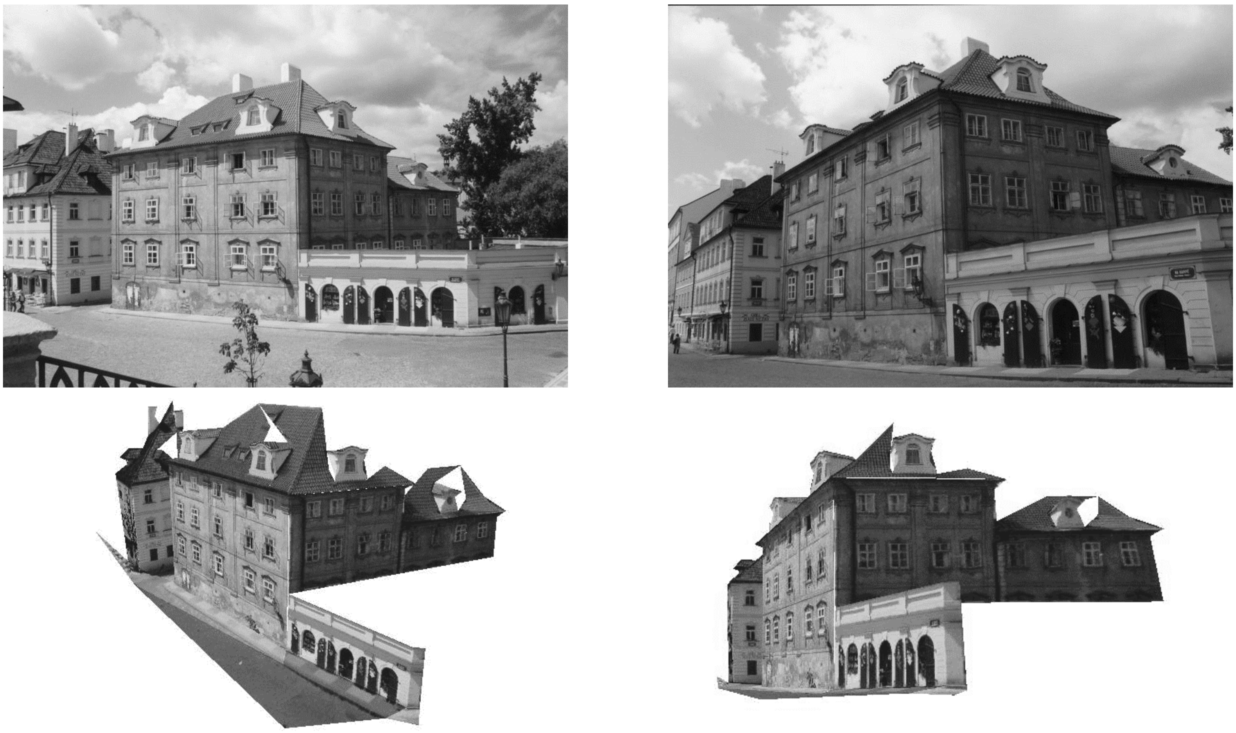

Figure 3: A 3D model of a single building, semi-automatically reconstructed from 7 photographs. Two input photographs and two shots of the output VRML model are shown. Notice that parts invisible on input photographs have not been reconstructed in 3D. (Courtesy of Center for Machine Perception, CTU in Prague)

3.1. Reconstruction

A big collection of methods comes from the field of computer vision, where 3D reconstruction is a traditional research topic. Not considering approaches using laser range finders, there have been intensive research activities in reconstruction models of architecture from photographs. Approaches have been tried using aerial (long-range) or ground (close-range) imagery; passive (only cameras) or active (typically, using a laser range finder) sensing; conventional or panoramic cameras; they can co-operate with GPS/map information or build the models from scratch. Sadly, it has been realized that full automation of the process is extremely difficult. The most difficult problems are those of correspondence (which parts on two or more photographs correspond to a single building part in 3D), registration (put independently reconstructed parts to a common coordinate frame), and recognition (name a part seen on a photograph). Even if prototypes of fully automatic systems do exist, the most promising approach to data acquisition is semi-automatic.

The following projects and the resulting literature are most relevant. From various commercial photogrammetric systems allowing to model buildings manually to some extent, let us name *Photomodeller* [15]. *Facade* was the project of the University in Berkeley aiming at modeling and rendering architecture from photographs interactively [13]. It inspired the successful commercial product *Canoma* by MetaCreations. The *Ascender* system (Automated Site Construction, Extension, Detection and Refinement) [4] is the main result of the *RADIUS* project, allowing automated modeling of architecture from calibrated aerial imagery. It proceeds from line extraction via single-image roof hypotheses to 3-D building models. *CAMERA*, automated acquisition of architectural CAD models of already built environments, is a large EC project covering a broad range of topics, e.g. 3-D models from both range images and photographs, recovery and fusion of features typical for architecture, incremental sensing, and fusing local models to a global one. *MIT City Scanning Project* [5] uses mobile sensor and panoramic images to build a large-scale model of a regular urban environment. *IMPACT* deals with semi-automatic 3-D reconstruction of buildings from aerial images, aiming to full automation [12][1][11][8]. A similar approach is continued for close-range images [18], including modeling fine details as windows.

Speaking on the research in Central Europe, a recognizable effort can be seen at VRVis center in Graz, Austria [10][20]. A combined model includes automatically created blocks of buildings (walls and roofs) from GIS data and several tools for reconstruction of houses from set of images and preparation of multiview texturing.

Common problems of 3D reconstruction from images are the big amount of resulting data (triangles, vertices, textures) and difficult post-processing of such non-organized data for interactive purposes, e.g. doors that can be open/closed, specific parts (paintings, reliefs) enriched by hyper-links, etc.

3.2. Modeling

CAD and CAAD systems seem to by highly useful for virtual cities, since they are primarily intended to create precise models of buildings and similar objects. Surprisingly, they play a minimal role in a process of data preparation for a virtual city system. Real architectural models contain too many details to be easy and fast rendered. Already existing models represent modern houses, whereas modeling old buildings is extremely hard, since such houses do not contain prefabricated parts and almost everything must be designed/modeled from scratch. From several attempts for utilizing CAD packages for the reconstruction of historical buildings, let us name the CHARISMATIC project [3] that uses special tools for processing elements (columns, arcs, etc.) typical for old architecture.

General modeling packages like 3D Studio Max are popular in computer graphics community. They can be efficiently used for the design of virtual houses especially when combined with well-prepared ortho photo textures. Exported data save the structure of a house and can be easily incorporated into a virtual reality presentation. The main problem here is to prepare input image(s) that will serve as a texture. Such images should not contain any obstacles like cars, people, dogs, street lamps, etc. Retouching is usually tough work done manually using general raster editors.

Figure 4: In addition to models of buildings, virtual city should incorporate a number of other three-dimensional objects that are typical for urban environment. The example on the left shows trees and benches, the right scene includes even moveable tram in front of the church. Snapshots were taken from the Virtual Old Prague project.

Yet another approach is used in Virtual Old Prague project [22] where manually preprocessed photos are semi-automatically placed on a virtual façade using a special program. The output data represent house model in several levels in details, thus their use in a virtual reality is straightforward.

In contrary to image based reconstruction, modeling with computer graphics tools allows for extension of pure architectural data by variety of other 3D objects that are present in a real city and that improve the natural look for a virtual visitor (see Fig. 4).

4. Data Structures

Large geometrical and texture data have to be carefully organized from several reasons. First, they should be grouped into small chunks suitable for a transfer through the Internet. Such small data packages are later identified on the client side and new chunks are requested as a response to user's activities. Next, a city is a subject of changes, thus various updates should be made possible. Those updates include not only buildings and streets that are added as a city grows, but also changes of hyperlinks attached to different geometrical data, new textures of reconstructed houses already present in a system, etc. And finally, a virtual city is a subject for a searching machine. It should provide hierarchical information about city districts, relations among houses, squares, and streets. The topological information is useful for finding the shortest path connecting two selected places.

The structure of a database holding a virtual city can serve not only for the rendering purposes, but can be oriented towards certain (business) targets. The core of a database has many features common with GIS databases. Our experience gained from the Virtual Old Prague project says that the content of a database should be language independent, e.g. geometrical data (positions, distances, elevations, etc.) should be stored in a general numerical format rather than in DXF or VRML file format. This approach helps to smoothly move to arbitrary final presentation platform when necessary.

Data structures for a virtual city often correspond with a real town layout. Typically, houses are grouped into streets. This natural arrangement is unfortunately not enough for all purposes. Finer granularity is required - long streets have to be split into smaller blocks; otherwise users would wait too long before displaying data. A structure of data should also correspond with structures used by algorithms for visibility preprocessing. Since the visibility computations are considered as a key issue for the speed of rendering large scenes, a city should be internally organized into cells with specific features. Visibility changes inside a cell should have minimal variations, portals between neighboring cells should have simple shape. Here the original historical arrangement of city parts does not correlate with computational requests. A task to find optimal cell decomposition for a given real city layout is still an open question.

5. Web-based Rendering

Many principles known from rendering large virtual scenes can be directly used in web based rendering of virtual cities. These include utilization of levels of details (both in geometrical and texturing meaning), data streaming [9], culling and visibility preprocessing (general [7] or specific to the urban environment [2]), and imposters [6] for farther objects and a complex background. All such methods are well elaborated in a literature, but most of them must be adapted for the use in

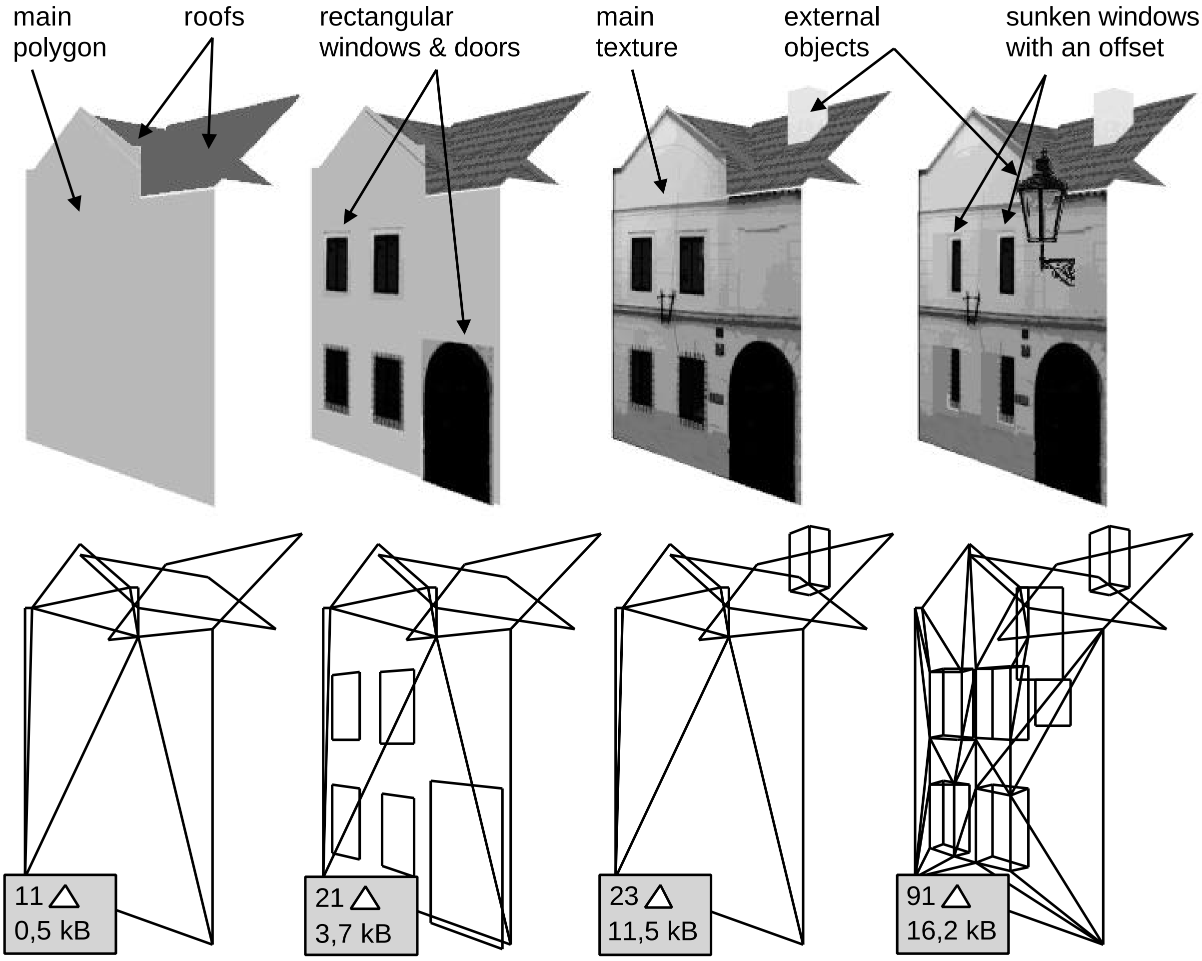

Figure 5: A representation of one house façade in four urban levels of details (Virtual Old Prague project)

a web environment. By our best knowledge, no publication discussing all those topics with respect to the web rendering exists, although it would be highly appreciated.

To show the typical difference between distributed and web-oriented rendering algorithms, let us mention the method for streaming complex 3D scenes proposed by Teler and Lischinski [14]. Their method is planned for remote walking. A server is continuously informed about user's movement in 3D space. A server is fully responsible for a selection of proper data to be rendered. This solution requires high-performance server and evidently limits the number of concurrent users. A successful web solution has to have almost opposite configuration – computations are performed in a client computer, the number of concurrent users is very high.

While some sort of troubles is caused by the nature of the web rendering, several already developed algorithms can be adapted into a virtual city system without problems. A positive example is shown on a principle of LOD widely used in a virtual reality. The idea is to define several representations for one object and to switch among them with respect to a distance and/or a current rendering speed. Thanks to the similarities in visual look of all buildings (the most important distant feature is a silhouette, windows and doors start to be visually important from the medium distance, and three-dimensional details are visible from the short distance only), representations can be designed in a uniform manner, even taking the web specific properties into account. One such example is shown in Fig. 5, where so-called *urban LOD* consists of four steps. Number of triangles together with typical size of geometry and texture data for each level is presented. The first number influences speed of rendering, the second one plays a role when transferring data through the Internet. Urban LOD can be thus adjusted for users with various Internet connections and computer performances. Popping effects caused by switching among levels can be suppressed by smooth geometrical transition between last two levels (scaling depth of windows) as proposed e.g. by Willmott [19].

A specific question is a selection of a viewer (renderer) on the client side. The most common choices are: VRML (or X3D in a future) compliant browser, an application written in Java/Java3D, and a special-purpose application written in another language. While the first two options have been originally designed for the web, the last one must solve the network issues and installation into a web browser environment. Thus the platform independence is limited. All options also differ in performance. Java is usually considered as slow. Also VRML browsers are not very fast because of complexity and a general functionality of the VRML specification. Special viewers make use of graphics hardware (Open GL, Direct X) and can be well tuned for the task of city presentation. Even engines for network gaming can be used here. A selection of proper 3D viewer thus remains open.

6. Conclusion

A complex virtual city presentation includes many types of data and algorithms from the field of computer vision, computer graphics, computational geometry, networking and databases. A common framework for processing input data, storing them in a database and presenting on the web is missing, but several software tools and attempts already exist. We expect that number of informatively rich virtual cities will soon appear on the web.

Problems to be solved are not principal and can be overcome either by implementing robust automatic algorithms (3D reconstruction, urban LOD creation, area visibility preprocessing) or by adapting existing approaches to a web environment (utilizing imposters, design of light weighted and platform independent 3D browsers). One of the most important conditions for a successful city presentation is an existence of a functional business model satisfying both providers and users.

7. Acknowledgement

This project has been partly supported by the Ministry of Education, Youth and Sports of the Czech Republic under research program No. Y04/98: 212300014 (Research in the area of information technologies and communications).

Our thanks belong to Tomas Werner from the Center for Machine Perception at CTU in Prague for his concise introduction to image based reconstruction methods, and to Kamil Ghais for the continuous improvement of data and algorithms used in the Virtual Old Prague web site.

8. References

[1] BAILLARD, C., ZISSERMAN, A., Automatic reconstruction of piecewise planar models from multiple views, in: CVPR99, pages II: 559 – 565, 1999.

[2] BITTNER, J., WONKA, P., WIMMER, M., Visibility Preprocessing in Urban Scenes using Line Space Subdivision, in: Proceedings of Pacific Graphics (PG'01), pages 276 – 284, Tokyo, Japan, October 2001.

[3] BROWNE, S. P., WILLMOTT, J., WRIGHT, L. I., DAY, A. M., ARNOLD, D. B., Modelling & Rendering Large Urban Environments. http://www.charismatic-project.com/MODELL.PDF

[4] COLLINS, R. T., JAYNES, C. O., CHENG, Y. Q., WANG, X. G., STOLLE, F. R., RISEMAN, E. M., HANSON, A. R., The ascender system: Automated site modeling from multiple aerial images. CVIU, 72(2): 143 – 162, November 1998.

[5] COORG, S., TELLER, S., Extracting Textured Vertical Facades from Controlled Close-Range Imagery, in: Proceedings of CVPR, pages 625 – 632, 1999.

[6] DECORET, X., SCHAUFLER, G., SILLION, F., DORSEY, J., Multi-layered impostors for accelerated rendering, in: Computer Graphics Forum 18/3 (Proceedings of Eurographics '99), pages 61-73, 1999.

[7] DURAND, F., A multidisciplinary survey of visibility. ACM SIGGRAPH Course notes. Visibility, Problems, Techniques, and Applications, July 2000.

[8] FRADKIN, M., ROUX, M., MAITRE, H., LELOGLU, U. M., Surface reconstruction from multiple aerial images in dense urban areas, in: CVPR99, pages II: 262 – 267, 1999.

[9] HOPPE, H., Progressive meshes, in: Computer Graphics Proceedings, Annual Conference Series, pp. 99-108, 1996.

[10] MAYER, H., BORNIK, A., BAUER, J., KARNER, K., LEBERL, F., Multiresolution Texture for Photorealistic Rendering, in: Proceedings of the Spring Conference on Computer Graphics SCCG, Budmerice, Slovakia, pages 174 – 183. 2001.

[11] MOONS, T., FRERE, D., VANDEKERCKHOVE, J., VANGOOL, L., Automatic modelling and 3d reconstruction of urban house roofs from high resolution aerial imagery, in: ECCV98, pages 410 – 425, 1998.

[12] SCHMID, C., ZISSERMAN, A., Automatic line matching across views, in: CVPR97, pages 666 – 671, 1997.

[13] TAYLOR, C. J., DEBEVEC, P. E., MALIK, J., Reconstructing polyhedral models of architectural scenes from photographs, in: ECCV96, pages II: 659 – 668, 1996.

[14] TELER, E., LISCHINSKI, D., Streaming of Complex 3D Scenes for Remote Walkthroughs. Computer Graphics Forum, 20(3), September 2001.

[15] The PhotoModeler 3D Modelling Tool. http://www.photomodeler.com.

[16] Virtual Paris (based on blaxxun technology). http://www.2nd-world.fr/

[17] Virtual Old Prague Project home page. http://www.ms.mff.cuni.cz/vsp/

[18] WERNER, T., ZISSERMAN, A., New techniques for automated architecture reconstruction from photographs. Springer-Verlag, 2002.

[19] WILLMOTT, J., WRIGHT, L. I., ARNOLD, D. B., DAY, A. M., Rendering of Large and Complex Urban environment for Real Time Heritage Reconstructions, in: Proceedings of VAST 2001 (Virtual Reality, Archaeology, and Cultural Heritage), Athens, Greece.

[20] ZACH, CH., KLAUS, A., BAUER, J., KARNER, K., From Real Cities to Virtual Habitats, VRVis Technical Report, TR-VRVis-2002-009. Austria, 2002.

[21] ZARA, J., CHROMY, P., CIZEK, J., GHAIS, K., HOLUB, M., MIKES, S., RAJNOCH, J., A Scaleable Approach to Visualization of Large Virtual Cities, in: Proceedings of the Fifth International Conference on Information Visualisation. London, pages 639 – 644, 2001.

[22] ZARA, J., Concise Tour to the Virtual Old Prague, accepted for the EUROGRAPHICS 2002 conference, Saarbrucken, Germany, 2002.