GRAPHICS TOOLS FOR THE GENERATION OF LARGE SCALE URBAN SCENES

Norbert Haala, Martin Kada, Susanne Becker, Jan Böhm, Yahya Alshawabkeh

University of Stuttgart, Institute for Photogrammetry, Germany
Forename.Lastname@ifp.uni-stuttgart.de

KEY WORDS: Terrestrial, Aerial, Laser scanning, Virtual Reality, Photo-realism

ABSTRACT:

The acquisition of 3D city and landscape models has been a topic of major interest within the photogrammetric community and a number of algorithms became available both for the automatic and semiautomatic data collection. Usually, these 3D models are used for visualisation in order to generate realistic scenes and animations e.g. for planning and navigation purposes. Such real-time visualisations of artificial or real-word scenes are one of the main goals in computer graphics. Due to the rapid developments in that field, realistic visualisations are feasible even for complex and area covering 3D datasets and are integrated in an increasing number of applications. However, the benefit of applying computer graphics tools within photogrammetric data processing is not limited to the visualisation and presentation of 3D geodata. As it will be demonstrated within the paper, also photogrammetric data collection can take considerable advantage of a direct integration of these tools for the generation of large scale urban scenes.

1. INTRODUCTION

Urban models can meanwhile be provided area covering and efficiently based on aerial images or LIDAR. Thus, they have become a standard product of photogrammetric data collection. Usually, these 3D models are used for realistic visualisations and animations in the context of planning and navigation applications based on powerful tools available from computer graphics. As it will be discussed within the paper, the application of these tools is not limited to visualisation purposes. Considerable benefit is also feasible, if they are directly integrated in the modelling and data collection process during the generation of virtual city models. This will be demonstrated for tasks like geometric model refinement and texture image generation.

Airborne data collection efficiently provides complete sets of 3D building models at sufficient detail and accuracy for a number of applications. However, due to the viewpoint restrictions of airborne platforms, the collected building geometry mainly represents the footprints and the roof shapes while information on the facades of the buildings is missing. Thus, in order to allow for high quality visualizations from pedestrian viewpoints, texture mapping based on terrestrial images is usually applied, additionally. As it will be discussed in section 2, this texture mapping process can be supported considerably, if approaches well known from computer graphics are used. As an example, on-the-fly texture mapping from real world imagery is feasible by the application of the so-called Graphics Rendering Pipeline. This enables a direct link between the texture extraction and mapping and the subsequent rendering process. In addition to geometric transformations between the 3D model and the real world texture images, an analysis of the model's visibility within these images has to be implemented. The required detection of occluded areas can become computational expensive especially for complex 3D scenes. However, well known approaches like z-buffering, which were originally developed for rendering purposes can be modified to allow for fast and efficient solutions.

Texture mapping increases the visual realism of building objects, which are represented by relatively simple geometric models. For this purpose, texture maps modify the rendered pixel values for the respective surfaces by the application of additional image information. However, the underlying object geometry is still defined by the coarse 3D model, i.e. the building facades are still represented by planar polygons. While this can be sufficient for moderate 3D structures and orthogonal views, protrusions like balconies or window ledges will disturb the visual impression especially for oblique views. In that case the difference between the available model and the true façade geometry as represented by the texture image will become clearly visible. Additionally, rendered views will appear unrealistic for large differences between the illumination of the 3D model and the texture images. While texture maps from real images represent the illumination at the time of image acquisition, the 3D model can be shaded using light sources at arbitrary locations and directions. Such simulations are for example required if 3D city models are used for architectural applications. Thus, for further visual enhancement of large scale urban scenes, a geometric refinement of the building facades is required.

Very detailed geometric representations of objects like buildings can be generated using terrestrial laserscanning. By these means an area covering acquisition of densely sampled 3D point clouds is feasible. These 3D point clouds can then be converted to polygonal meshes, which are in principle immediately suited for graphic rendering. However, since laser scanning can easily result in several million 3D points, the huge amount of the data makes it unrealistic to derive and visualize such a representation for larger areas or parts of a city. An alternative solution, which is often chosen in computer graphics is to separate the representation of the coarse object geometry from the representation of fine surface details. If for example the object geometry is already available approximately from a given 3D building model, corresponding laser points can be extracted from the available 3D point cloud using a suitable buffer operation. As it will be demonstrated in section 3, this information can then be used for a refined visualization of the façade geometry using standard computer graphic techniques like bump mapping or displacement mapping.

These approaches allow for a very efficient geometric enhancement of existing 3D building models. However, a grouping of the laser point measurements into meaningful

entities and different architectural structures like windows, doors or balconies is still necessary for a number of applications. However, the required generation of such detailed CAD representations typically is an interactive and time-consuming process. For this reason within the final part of the paper, the efficient refinement of existing building models based on terrestrial laser measurements will be demonstrated. For this purpose an approach based on so-called half space modeling approach, which is also known in computer graphics will be used.

2. TEXTURE MAPPING

Texture image information is frequently used during surface rendering in order to improve the visual realism of objects, which are represented by relatively simple geometric models. Thus, in the context of virtual city models either artificial texture or real world imagery is mapped to building facades and roofs. Despite of techniques like synthetic image generation based on grammars, which have been proposed for procedural façade texturing (Wonka et al 2003), most applications require the availability of suitable real world imagery of the building objects to be visualised. For this purpose, façade imagery is extracted from terrestrial photographs. These images are then associated with the geo-referenced 3D building model. If the interior and exterior orientation of the respective texture image is available, the image sections are directly mapped to the corresponding 3D surface patches. This process of texture extraction and placement can be realized very efficiently using the functionality of 3D graphics hardware (Kada et al 2005). Within this process also tasks like the elimination of image distortions or the integration of multiple images are solved on-the-fly. Additionally, the generation and storage of intermediate images is avoided and a real-time modification of the input real-world photographs is feasible.

Usually, virtual city models are based on buildings represented by relatively simple polyhedral models. For this reason, tasks like occlusions within the available texture images can be detected without problems. However, the effort for visibility analyses increases considerably for geometrically complex object representations. This is especially true for object representations based on meshed surfaces like they are frequently available for applications lie cultural heritage documentation. Since these meshes are derived from dense 3D point clouds as collected by terrestrial laser scanning a huge amount of surface patches is generated. For this reason, standard approaches like manual texture extraction and mapping to solve the occlusion problem can become very tedious and time consuming. On the other hand, in computer graphics, efficient approaches are available to solve the visibility problem, in order to decide which elements of a rendered scene are visible, and which are hidden. Based on these approaches for occlusion detection, efficient texture extraction and placement has also been realised in the context of cultural heritage representation. As an example, (Grammatikopoulos L. et al 2005) use a modification of z-buffering (Catmull 1974), which computes the visibility of each pixel by comparing the surface depths at each position in image space and selects the one closer to the observer. (Alshawabkeh & Haala 2005) combine such an approach with techniques based on the painter's algorithm. In this approach, also well known in computer graphics for occlusion detection, the model polygons are sorted by the depth in object space with respect to the collected texture image. This process is demonstrated exemplarily in Figure 1 and Figure 2.
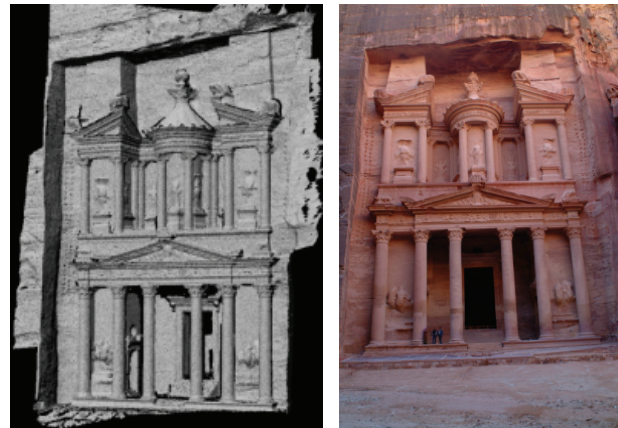

Figure 1: Meshed 3D model from terrestrial laser scanning (left) and texture image from independent camera station (right).

Figure 1 (left) shows a meshed 3D model of the Alkasneh monument in Petra, Jordan. The data was collected by terrestrial laser scanning from different stations at an average resolution of 2 cm. This resulted in more than 10 million triangles of the final model. Figure 1 (right) shows a colour image of the monument collected by a digital camera, which is used for texture mapping.
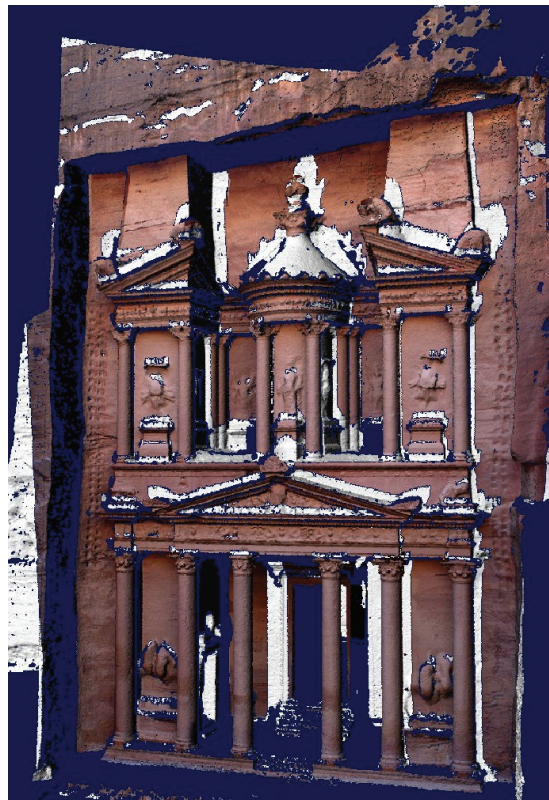

Figure 2: Result of texture mapping with automatic occlusions detection.

The result of this process is depicted in Figure 2. The parts of the 3D model, which are occluded in the available texture image are marked as grey regions. These regions are extracted from the original meshed 3D model and then used as input surfaces for a second texture image. The process for visibility checking and model separation is repeated until the model is textured from all the available images. Finally, the separated parts are merged to generate the overall model.

3. ENHANCED VISUALISATION USING TERRESTRIAL LIDAR DATA

As it is has been discussed in the previous section, terrestrial laserscanning allows an efficient acquisition of densely sampled point clouds of architectural objects. Using state-of-the-art meshing techniques, though not without problems, these point clouds consisting of several million points can be converted to polygonal meshes, immediately suited for graphic rendering. While dense polygon meshes might be a suitable representation for relatively small scenes with a limited number of objects, the large amount of the data makes it unrealistic to derive and efficiently visualize such a representation for even a part of a city. However, efficient model generation and visualization is feasible, if the representation of coarse building geometry as i.e. provided from aerial data collection is separated from the representation of fine surface detail of the facades as provided by terrestrial laser scanning.

This discrepancy in the representation of fine detail and surfaces in a model has long been noted in computer graphics. An important idea in this field was introduced by (Blinn 1978). He observed that the effect of fine surface details on the perceived intensity is "primarily due to their effect on the direction of the surface normal … rather than their effect on the position of the surface". This idea is a first step in separating the coarse and fine geometry of a surface. While the fine details are not modeled as explicit geometry, their perturbation on the direction of the surface normal is recorded and introduced to the computation of surface intensity. This idea of generating normal maps to enhance the realism of computer generated surfaces is known as bump mapping and has become a standard in computer graphics. This feature is implemented in hardware on nearly all up-to-date computer graphics card (see for example (Kilgard 2000) and therefore is ideally suited for high performance rendering and real-time animation.
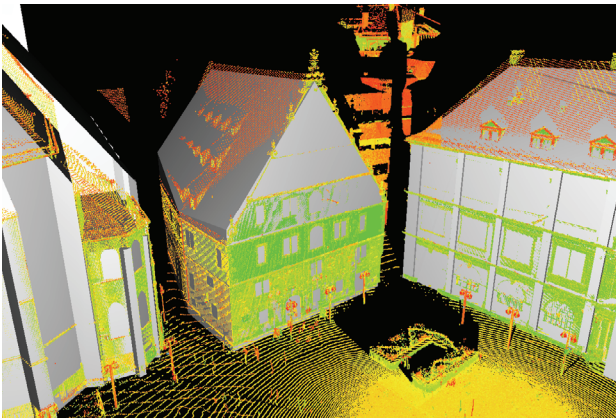

Figure 3: 3D point clouds from laser scanning aligned with a virtual city model.

In order to apply this technique for an enhanced visualization of urban scenes, the detailed point cloud from terrestrial laser scanning and the coarse 3D building models from airborne data collection have to be transformed to a common reference coordinate system. This can be solved by an alignment of the laser points to the 3D city model using standard approaches like the Iterative Closest Point approach (Böhm & Haala 2005). The resulting co-registration of both data sets is depicted in Figure 3. After this co-registration step for each planar façade polygon the corresponding laser points can be selected easily by a simple buffer operation. Based on these laser points a distance image with respect to the planar façade polygon is computed.
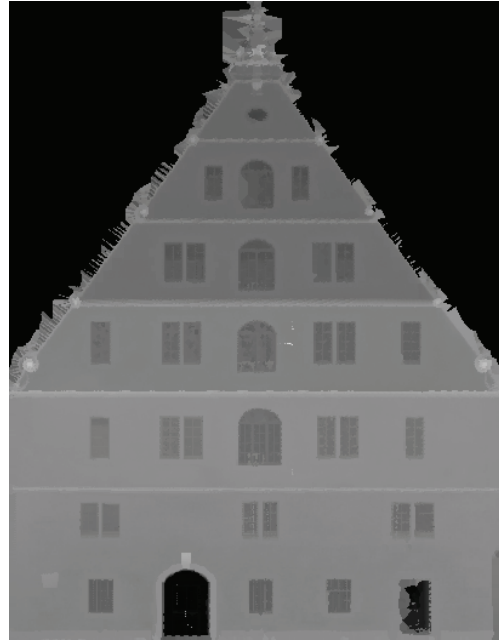

Figure 4: Distance image with respected to a planar building façade as extracted from terrestrial laser scanning.

This image shown in Figure 4 represents the deviations of the true façade to a planar surface. It is computed by a re-interpolation of the originally measured laser points within the facade's buffer region to a regular raster, which is centered on the plane of the facade. This image can then be used to derive modifications of surface normals, which are applied for bump mapping.
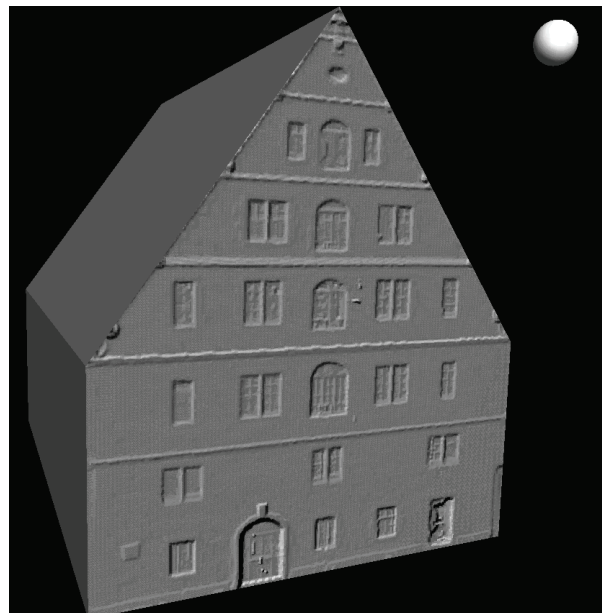

Figure 5: Visualisation of building façade based on bump mapping.

As it is shown in Figure 5, the bump map of the façade allows for a realistic shading of the surface with respect to a pre-defined direction of the illumination. For demonstration purposes, the respective direction of illumination is also visible on the sphere added in the upper right corner. Bump mapping can improve the visual appearance during high performance rendering and real-time animation. Still, it is limited to the basic shading of the respective objects. Since only the surface

normals are affected, effects of the refined geometry on shadows, occlusions or object silhouettes are not considered. This resulted in the development of displacement mapping (Cook 1984), which adds small-scale detail to the respective surfaces by actually modifying the positions of the surface elements. Usually, displacement mapping is implemented by subdividing the original surface into a large number of smaller polygons. The vertices of these polygons are then shifted in normal direction.
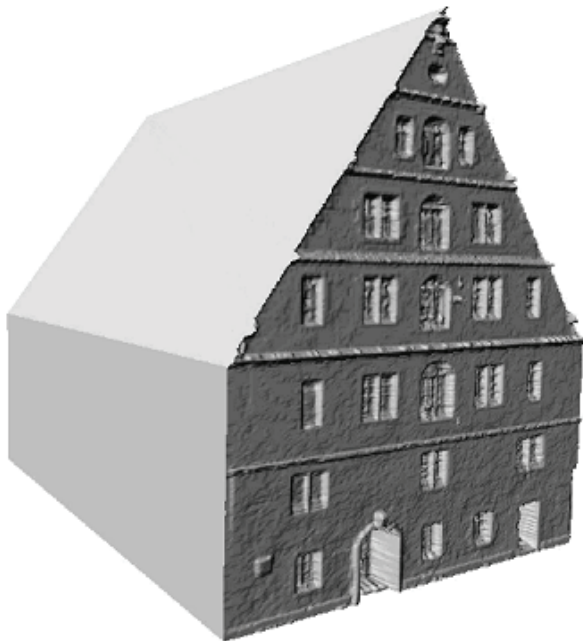

Figure 6: Refined visualisation of building façade using displacement mapping.

An exemplary visualisation based in displacement mapping is depicted in Figure 6. Again the distance image already shown in Figure 4 was used to provide the required geometric information.

4. MODEL REFINEMENT OF BUILDING FAÇADES BY TERRESTRIAL LIDAR

As it has been demonstrated, bump and displacement mapping allow for a visual improvement of building facades. However, the underlying distance image is limited to the representation of 2½ D structures. In contrast, true 3D structures, which are required for visualization of the highest possible quality, require a modification of the original 3D building model. Within the required reconstruction step also constraints between different object parts like co-planarity or right angles should be guaranteed. This will be demonstrated in the following by an approach for the integration of window objects in an existing 3D building model.

Usually, tools for the generation of polyhedral building models are either based on a constructive solid geometry (CSG) or a boundary representation (B-Rep) approach. In contrast, the presented model refinement is based on cell decomposition which is also well known in solid modelling. Cell decomposition is a special type of decomposition models, which subdivides the 3D space into relatively simple solids. Similar to CSG, these spatial-partitioning representations describe complex solids by a combination of simple, basic objects in a bottom up fashion. In contrast to CSG, decomposition models are limited to adjoining primitives,

which must not intersect. The basic primitives are thus 'glued' together, which can be interpreted as a restricted form of a spatial union operation.

4.1 Data pre-processing

Similar to the generation of a distance image discussed in section 3, terrestrial LIDAR data has to be selected for the respective building facades by a buffer operation.
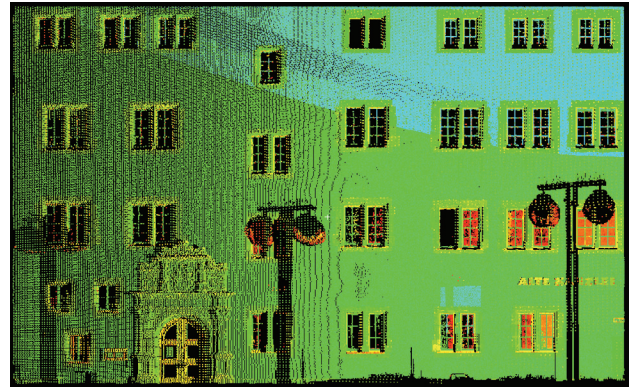

Figure 7: Laser point cloud to be used for the geometric refinement of the corresponding building façade.

Figure 7 shows this point cloud after transformation to a local coordinate system as defined by the façade plane. After mapping of the 3D points to this reference plane, further processing can again be simplified to a 2.5D problem. Thus algorithms originally developed for filtering of airborne LIDAR (Sithole & Vosselman 2004) can be applied. However, in contrast to the distance image in Figure 4, no re-interpolation to a raster data set is computed. In contrast, the original 3D information from the laser point cloud is preserved.

4.2 Generation of façade cells

As it is visible for the façade in Figure 7, usually no 3D points are measured at window areas. Either no measurement is feasible at all due to specular reflections of LIDAR pulses at the glass, or the available points refer to the inner parts of the building and are eliminated due to their large distance to the façade. Thus, in our point cloud segmentation algorithm, window edges are given by no data areas. In principle, such holes can also result from occlusions. However, this is avoided by using point clouds from different viewpoints. In that case, occluding objects only reduce the number of LIDAR points since a number of measurements are still available from the other viewpoints.
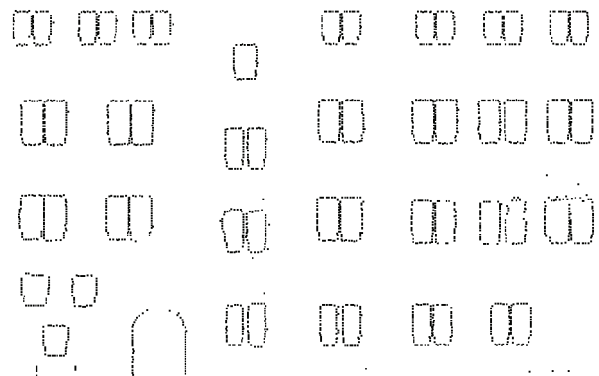

Figure 8: Detected edge points at horizontal and vertical window structures.
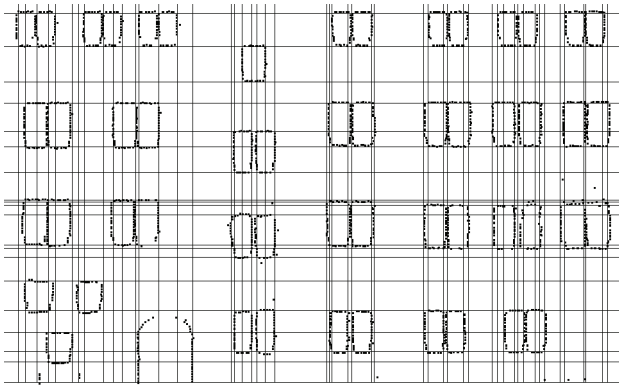
Figure 9: Detected horizontal and vertical window lines.

During segmentation of edge points, four different types of window borders are distinguished: horizontal structures at the top and the bottom of the window, and two vertical structures that define the left and the right side. As an example, to extract edge points at the left border of a window, points with no neighbour measurements to the right have to be detected. In this way, four different types of structures are detected in the LIDAR points of Figure 7.

These extracted points are shown in Figure 8. Figure 9 then depicts horizontal and vertical lines, which can be estimated from non-isolated edge points in Figure 8. Each line depicted in Figure 9 can be used to define the basic structure of potential window cells within the building façade. These 3D cells are generated based on theses lines as well as the façade plane and an additional plane behind the façade at window depth. This depth is available from LIDAR measurements at window cross bars. The points are detected by searching a plane parallel to the façade, which is shifted in its normal direction.

4.3 Cell separation

All the generated 3D cells are then separated into building and non-building fragments based on the availability of measured laser points. For this purpose the point availability map depicted in Figure 10 is used. Similar to the detection of edge points shown in Figure 8, it is assumed within this process, that no laser points are measured at window positions. This is clearly demonstrated in Figure 10. Black pixels are grid elements where LIDAR points are available. In contrast, white pixels define raster elements with no 3D point measurements

Figure 10: Point-availability-map.

Of course, the already extracted edge points in Figure 8 and the resulting structures in Figure 9 are more accurate than this rasterized image. However, this limited accuracy is acceptable since the binary image is only used for classification of the 3D

cells as they are already created from the detected horizontal and vertical window lines.
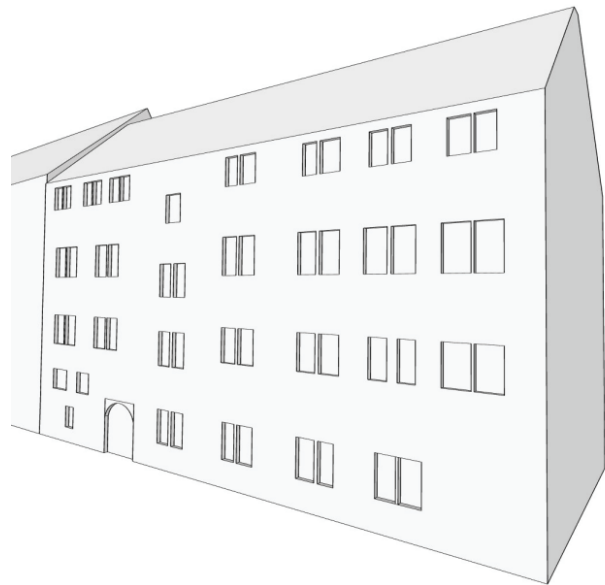

Figure 11: Building model with refined facade structure.

The final result of the building façade reconstruction from terrestrial LIDAR is depicted in Figure 11. For this example the window areas were cut out from the coarse model, which was provided from aerial data. While the windows are represented by polyhedral cells, also curved primitives can be integrated in the reconstruction process. This is demonstrated exemplarily by the round-headed door of the building.

5. CONCLUSION

Real-time visualizations of area covering large scale urban scenes can be realized based on standard hard- and software components. However, the application of tools from computer graphics is not only profitable for photorealistic visualization of urban environments. Major benefit is also feasible, if these tools are integrated in the data collection step. Within the paper this was demonstrated by techniques aiming at the enhancement of 3D scenes and building models. For this purpose image and terrestrial laser data was integrated for texture mapping and shape refinement, respectively. By these means the benefit of closely linking the data collection and presentation steps, i.e. combining techniques for photogrammetric data collection with tools available from computer graphics for the generation of large scale urban scenes could be demonstrated.

6. REFERENCES

Alshawabkeh, Y. & Haala, N. [2005]. Automatic Multi-Image Photo-Texturing of Complex 3D Scenes. CIPA 2005 IAPRS Vol. 34-5/C34, pp.pp. 68-73.

Blinn, J.F. [1978]. Simulation of wrinkled surfaces. *ACM SIGGRAPH Computer Graphics* **12**(3), pp.286-292.

Böhm, J. & Haala, N. [2005]. Efficient Integration of Aerial and Terrestrial Laser Data for Virtual City Modeling Using LASERMAPS. IAPRS Vol. 36 Part 3/W19 ISPRS Workshop Laser scanning 2005 , pp.192-197.

Catmull, Ed. A. [1974] Subdivision Algorithm for Computer Display of Curved Surfaces. PhD thesis, Computer Science Department, University of Utah, Salt Lake City, UT, Report

UTEC-CSc-74-133.

Cook, R.L. [1984]. Shade trees. *ACM SIGGRAPH Computer Graphics* **18**(3), pp.223-231.

Grammatikopoulos L., Kalisperakis I., Karras G. & Petsa E. [2005]. Data Fusion from Multiple Sources for the production of Orthographic and Perspective Views with Automatic Visibility Checking. CIPA 2005 IAPRS Vol. 34-5/C34.

Kada, M., Klinec, D. & Haala, N. [2005]. Facade Texturing for rendering 3D city models. ASPRS Conference 2005, pp.78-85.

Kilgard, M. [2000]. A Practical and Robust Bump-mapping Technique for Today's GPUs. GDC 2000: Advanced OpenGL Game Development.

Sithole, G. & Vosselman, G. [2004]. Experimental comparison of filter algorithms for bare-earth extraction from airborne laser point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing* **59**(1-2), pp.pp. 85-101.

Wonka, P., Wimmer, M., Sillion, F. & Ribarsky, W. [2003]. Instant Architecture . *ACM Transactions on Graphics (TOG) Special issue: Proceedings of ACM SIGGRAPH 2003* **22**(3), pp.669-677 .