Eurographics Workshop on Urban Data Modelling and Visualisation (2013)
V. Tourre and G. Besuievsky (Editors)

Toward Automated Façades Generation from Oblique Aerial Images

Matthias Vahl and Uwe von Lukas

Fraunhofer IGD, Germany

Abstract

In this paper we introduce an approach for automatic recognition and description of building façades from oblique aerial images. This approach does not use street-side images, like Google StreetView, but aerial images to derive photorealistic city models. The paper starts with a short summary of the state of the art in the domain of image-based modeling, automated recognition and description of façades from (oblique aerial) images. Then the context of the research project is described including the hardware setup and and application requirements for complex façade analysis. Afterwards a novel approach of a grid graph model with traversers is suggested to solve the problem of façade recognition. The framework that implements the approach in a flexible manner is introduced. Finally, the paper ends with a conclusion and an outlook to future works.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—Graphics data structures and data types I.5.1 [Pattern Recognition]: Models—Structural

1. Introduction

High-end car navigation systems already use 3D models to improve the visual quality for the user. This implies big challenges because the data volume of near-realistic 3D models is much bigger than simple plain maps. Currently provided navigation systems neither extend the on-board-memory nor increase visualization power. Therefore it is necessary to reduce the data volume whilst holding a high level of quality.

2. Related Work

Some of today's city models have already derived from aerial images. Most prominent examples are GoogleEarth and Microsoft's Virtual Earth. Due to the fact of poor camera resolution it is very difficult to gather realistic ground models. To solve this, image quality can be improved by street-side shots or by increasing the resolution of the aerial camera. New camera technology could be able to waive expensive street-side data collection by vehicles.

Xiao et al. [XFZ*09] assume that is not possible to acquire photorealistic models at ground level from aerial images. Their approach is to obtain image sequences by car driving and then to start an automated process containing semantic segmentation, block partition, façade modeling and post processing. However, in the domain of car navigation it is not required that users see a photorealistic visualization, the objective should rather be to reach a high degree of recognition. Hence, exact representation is not a requirement. However, street-side shots are much more expensive to acquire and suffer a high risk of occlusions. Furthermore, this approach can result in conflicts with the privacy of the habitants.

The procedural modeling of real buildings is not trivial. Müller et al. [MZWV07] show in their approach how façade textures with different resolutions are first transformed into a semantic 3D model and then rendered in high quality. Undistorted façade images as input data are divided automatically and after that coded into elements. Result is the rendered reconstruction inclusive shadows and reflections. The possible recognition rate for regular façades is given with 90 % and consumes three minutes per image. But it is necessary to set the depth manually.

Using the approach of Müller et al. [MZWV07] Ricard et al. [RRA07] developed another solution for extracting façade structures for real models. For cost reasons only image-based reconstruction is considered. In this approach as well as in the approach of Müller et al. recognition rates are reached for simply structured façades only. Thus the al-

DOI: 10.2312/UDMV/UDMV13/005-008

gorithm is not robust enough and the computation time is not acceptable for industrial generation of city models.

Wenzel et al. [WDF08, WDF*07] present another method for detection of repetitive structures. The method is applied on façade images to gain a description of regularity of façade windows. The result is defined as compact description of façade structure without a-priori knowledge. However, this interesting way is not very robust against noise.

Kreutzmann et al. [KTN09] develop a probabilistic framework which contains a bottom-up, probabilistic approach with dynamic prior probabilities which are taken from the global scene context. In this approach Bayesian Compositional Hierarchies [Neu08] as well as decision trees [TN09] are used.

Ali et al. [ASJ*07] show a possibility for building window detection using Haar wavelet decomposition and the classification by Gentle This approach is able to work well with non-orthogonal views, especially for strong structured historical buildings.

3. Application Context

One of the main requirements for industrial production of 3D city models is to minimize the error of the first kind for façade detection (high specificity). Hence, the operator can be sure that accurately interpreted façades by the algorithm are really correct. Otherwise more errors of the second kind are taken into account (low sensitivity). It is easier for the operator to ignore false positives than to scan all the results for false negatives.

The input data consists of oblique aerial images taken whilst flying in 600 m height. The camera sensors onboard the airplane have a resolution of 4896 x 3264 pixels at 12 bits depth and use a RGGB Bayer pattern. The sensor's resolution, focal length and angle of view to the ground produce a pixel size of about 6 cm.

The following façade properties should be detected: the height between floors, the dominant façade colors and the façade elements and structure. Besides the high quality of the façade, a possibly seldom user interaction by hand during the recognition is aspired. The detected elements should be mapped to a set of predefined façade elements from a library.These predefined façade elements should be parametrizable in color, size etc.

4. Approach and Concept

4.1. Recording

Figure 1 shows the overall process of image-based modeling of city models. Input data is a set of geo-referenced oblique aerial images. As a first step, an automatic exposure compensation is applied to all images. The next step is an optimized demosaicking. For the given raw images and the application

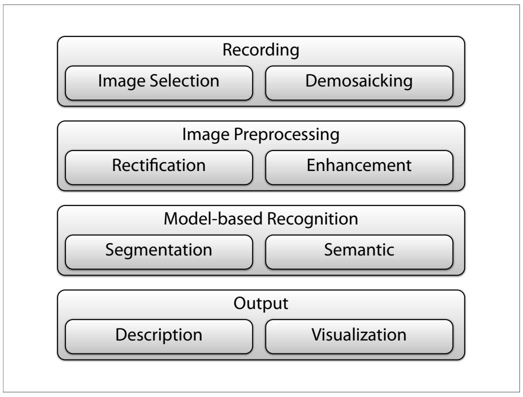


Figure 1: *Overall process*

context the algorithm of Hamilton and Adams [HA97] has the best trade-off between quality and speed.

4.2. Image Preprocessing

The preprocessing step starts with a rectification of the image. This is possible due to the fact that the exact position of the plane and flight height as well as the intrinsic camera parameters are available for each image. With their software package tridicon^{TM} CityDiscoverer, GTA Geoinformatik compute the position of building's roof corners and cut out the façades of buildings using photogrammetry algorithms.

4.3. Model-based Recognition

Model-based recognition is the main step of the whole processing chain. Initial point of this approach is the experience that low-level computer vision is not powerful enough to recognize façades at all. Building façades are man-made structures that are characterized by statical conditions, style, epoch, region and aesthetically considerations. The idea of our work is to use this information to support the recognition process. The core part of this complex façade analysis solution is a semantic model that does not only recognize single elements such as windows and doors but uses structural information and constraints to build a complete model of the façade. This approach significantly improves the quality of the recognition. For example, the axis-parallel alignment of windows on a façade follows certain rules.

The input data contains many errors: occlusions by trees, open and closed windows, different featured balconies etc. This typically leads to errors in the façade recognition. As façades are structures created and made by men that follow certain conditions. These conditions are used for formulating semantic rules. Façade elements such as windows and doors are aligned regularly and these have few different sizes. Hence our approach regards this context information and conditions but those are numerous and complex, partitionally depending on each other and difficult to implement implicitly. Our proposed architecture allows formulating certain aspects, e.g. the regularity of windows. Classical methods, especially segmentation, are used as basis for image analysis but are controlled by the model.

Given the set of all images as I. For a compact, formalized representation of the main properties of a building façade a grid graph serves as the underlying model with following conditions:

1. The whole graph G consists of nodes and edges $G = (N, E)$ and represents a façade part.
2. Each node $n \in N$ corresponds with one façade element, e. g. a window or a door.
3. Each edge $e \in E$ represents a neighborhood relationship between façade elements.
4. Graph, nodes and edges are parametrizable. There are three sets of functions P^N, P^E and P^G and it are given $P^N : N \rightarrow \mathbb{R}^n$, $P^E : E \rightarrow \mathbb{R}^n$ and $P^G : G \rightarrow \mathbb{R}^n$.

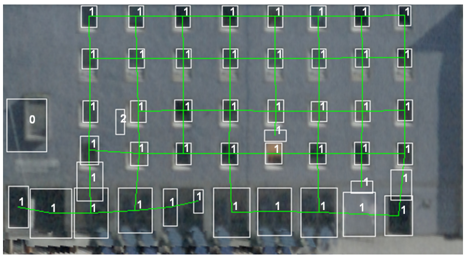
For example the window size is declared by two functions $p_{windowSizeX}$ and $p_{windowsSizeY} \in P^N$ that hold the dimension of a window represented by its node. Typically, the original image is stored as a parameter of the graph: $p_{image} \in P^G$ so gives $p_{image} : G \rightarrow I$ the original image.

Besides the compact and formalized representation, the grid graph is helpful to clone the model for alternative calculations. In this way, graph variants that differ in parameters and structure are derived. Each copy can be interpreted as one hypothesis. The rating of the accuracy of different graphs allows comparing and selecting the best characteristics. In this manner the proposed model is open for other methods such as artificial intelligence. For formalizing crucial conditions so-called graph traverser —software modules that traverse the whole graph— are introduced. There are two kinds of traversers. Structure-changing traversers manipulate the structure, e.g. insert or remove missing windows. Parameter-changing traversers manipulate the parameterization of the graph, the nodes or the edges.

Figure 2(a) shows a façade with the associated grid graph. The nodes representing windows contain two parameters: the center point and the size of the windows. The grid graph was initially parameterized by a grid graph generator (e. g. by an adaptive segmentation). The irregularity of the grid is easy to identify due to the incorrect segmentation. Figure 2(b) shows the state of the grid graph model after applying different traversers. All windows of the façade have now the same size and their positions are also corrected.

4.4. Output

After the recognition the second step follows—the generation of a near-realistic façade textures. These textures are applied on the 3D bodies for increasing user's recognition



(a) The initialize state of the grid graph, after applying the grid graph generator on the Façade image

(b) The graph after applying the grid repair and the windows smoother traverser

Figure 2: *Visualization of the grid graph (The grid graph with its nodes is marked with green lines, the recognized window size is represented by a white rectangle.)*

value. Doors, windows and the dominant façade colors are characteristic visualized features. Visualizing the façade by photo textures taken from aerial images or street-side shots mobile systems would quickly reach its limits. Otherwise real pictures contain noise and artifacts (trees, parking trucks etc.) that would destroy the impression of the user.

5. Implementation

Figure 3 describes the architecture of the developed framework. The main part —the processor— processes sequences of commands that are written in an own scripting language and controls the whole process. Furthermore, this concept allows domain experts —that not necessarily have computer vision knowledge— to formulate rules for element detecting by combining commands. Currently the process is only a linear pipeline. Later flow control (loops and conditions) and forks will be added. This allows the calculation of different hypotheses and the integration of voting concepts.

To guarantee modularity and flexibility a plug-in concept is used – every command is encapsulated in a library that contains both the object code and the declaration of the algorithm. A registry holds all possible plug-ins or respectively commands that can be divided into five different types:

1. Image Processor (IP) plug-ins need and give an images ($IP : I \rightarrow I$)

2. Grid Graph Generator (GG) plug-ins build a grid graph model from an image ($GG : I \rightarrow G$)
3. Graph Traverser (GT) plug-ins manipulate the grid graph model ($GT : G \rightarrow G$)
4. Export Traverser (ET) plug-ins exports to a structured data format, e. g. Scalable Vector Graphics (SVG) ($ET : G \rightarrow G \times E$)
5. Feature Traverser (FT) plug-in extract one concrete feature $f \in F$ or property, e.g. dominant façade colors ($FT : G \rightarrow G \times F$).

Therefore, the general formula for a script is:

$$(I \rightarrow I)^*(I \rightarrow G)^1(G \rightarrow G)^*(G \rightarrow F)^*(G \rightarrow E)^*$$

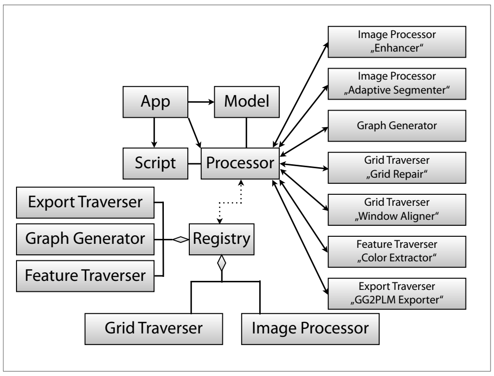


Figure 3: *Architecture of the developed solution*

The registry controls the life-cycle of the plug-ins.The grid graph model is represented by an acyclic bi-directional graph that consists of parametrizable nodes and edges. The framework has been implemented already. The implementation of the grid repair traverser and some other traversers shows that the software architecture is valid. Briefly, two plug-ins are described: The *grid graph generator* builds the initial grid graph from a given façade image. This is realized by using Otsu's [Ots75] adaptive segmentation algorithm with respect to the known real dimensions. An initial grid graph will be produced. The *window smoother traverser* corrects sizes and positions of façade elements. All window corners are projected onto the vertical and horizontal axes. By using k-means clustering and assuming that a façade part usually contains only a few different windows sizes, the sizes and positions of the windows is corrected.

Currently an interactive evaluation tool based upon the framework allows the evaluation of small test series interactively. This application supports also integrating of feedback and annotations into the data sets. This is very useful for distributed development and testing. The evaluation shows that this approach is suitable for analyzing façades. The basic window and door detection combined with dominant color detection is already in industrial use. Our implementation meets the special industrial requirement for computational time by using CUDA on GPU.

6. Conclusion and Future Work

In this article a concept of an efficient solution for extracting highly compact textures of buildings is introduced. The first implementation shows that model-based recognition can improve computer vision-based analysis of façades substantially. Future work includes the implementation of the concept at full range which means formulating further rules and implementing corresponding traversers. First evaluation shows positive results, but detailed evaluations and comparisons with other algorithm should be done.

7. Acknowledgement

This work was funded by the State Ministry of Economic Affairs of Mecklenburg-Vorpommern, Germany.

References

[ASJ*07] ALI H., SEIFERT C., JINDAL N., PALETTA L., PAAR G.: Window Detection in Facades. In *Proceedings of the 14th International Conference on Image Analysis and Processing* (Washington, DC, USA, 2007), ICIAP '07, IEEE Computer Society, pp. 837–842. 2

[HA97] HAMILTON J., ADAMS J.: Adaptative color plan interpolation in single sensor color electronic camera, 1997. 2

[KTN09] KREUTZMANN A., TERZIĆ K., NEUMANN B.: Context-aware classification for incremental scene interpretation. In *Proceedings of the Workshop on Use of Context in Vision Processing* (New York, NY, USA, 2009), UCVP '09, ACM, pp. 2:1—-2:6. 2

[MZWV07] MÜLLER P., ZENG G., WONKA P., VAN GOOL L.: Image-based procedural modeling of facades. *ACM Transactions on Graphics 26*, 3 (2007), 85 pp. 1

[Neu08] NEUMANN B.: Bayesian Compositional Hierarchies - A Probabilistic Structure for Scene Interpretation. In *Dagstuhl Seminar Proceedings - Logic and Probability for Scene Interpretation* (2008), vol. 08091. 2

[Ots75] OTSU N.: A Threshold Selection Method from Gray-Level Histograms. *Automatica 20*, 1 (1975), 62–66. 4

[RRA07] RICARD J., ROYAN J., AUBAULT O.: From photographs to procedural facade models. *ACM SIGGRAPH 2007 posters*, sap 0520 (2007), 2006. 1

[TN09] TERZIC K., NEUMANN B.: Integrating Context Priors into a Decision Tree Classification Scheme. Department Informatik, Vogt-K{ö}lln-Str. 30, 22527 Hamburg, Germany, 2009. 2

[WDF*07] WENTZEL S., DRAUSCHKE M., FÖRSTNER W., GEB V., REGELM D., SUCHE D., SCHWIERIGKEITEN A., SCHRITT D., ABLEITUNG D., WEITEREN D., ZIEL D.: Detektion wiederholter und symmetrischer Strukturen in Fassadenbildern. In *DGPF Tagungsband - Dreilaendertagung SGPBF, DGPF und OVG* (2007), vol. 16, pp. 119–126. 2

[WDF08] WENZEL S., DRAUSCHKE M., FÖRSTNER W.: Detection of Repeated Structures in Rectified Facade Images. *Pattern Recognition and Image Analysis 18*, 3 (2008), 406–411. 2

[XFZ*09] XIAO J., FANG T., ZHAO P., LHUILLIER M., QUAN L.: Image-based street-side city modeling. In *ACM SIGGRAPH Asia 2009 papers* (New York, NY, USA, 2009), SIGGRAPH Asia '09, ACM, pp. 114:1—-114:12. 1