

INCREMENTAL REFINEMENT OF FAÇADE MODELS WITH ATTRIBUTE GRAMMAR FROM 3D POINT CLOUDS

Y. Dehbi^{a,*}, C. Staat^{a,*}, L. Mandtler^{a,*} and L. Plümer^a

^a Institute of Geodesy and Geoinformation, University of Bonn, Meckenheimer Allee 174, Bonn, Germany -
(dehbi, staat, mandtler, pluemer)@igg.uni-bonn.de

Commission III, WG III/4

KEY WORDS: Weighted Attribute Grammar, Parse Tree, Incremental Parser, Kernel Density Estimation, 3D Building Models, UAV

ABSTRACT:

Data acquisition using unmanned aerial vehicles (UAVs) has gotten more and more attention over the last years. Especially in the field of building reconstruction the incremental interpretation of such data is a demanding task. In this context formal grammars play an important role for the top-down identification and reconstruction of building objects. Up to now, the available approaches expect offline data in order to parse an a-priori known grammar. For mapping on demand an on the fly reconstruction based on UAV data is required. An incremental interpretation of the data stream is inevitable. This paper presents an incremental parser of grammar rules for an automatic 3D building reconstruction. The parser enables a model refinement based on new observations with respect to a weighted attribute context-free grammar (WACFG). The falsification or rejection of hypotheses is supported as well. The parser can deal with and adapt available parse trees acquired from previous interpretations or predictions. Parse trees derived so far are updated in an iterative way using transformation rules. A diagnostic step searches for mismatches between current and new nodes. Prior knowledge on façades is incorporated. It is given by probability densities as well as architectural patterns. Since we cannot always assume normal distributions, the derivation of location and shape parameters of building objects is based on a kernel density estimation (KDE). While the level of detail is continuously improved, the geometrical, semantic and topological consistency is ensured.

1 MOTIVATION AND CONTEXT

3D building models are nowadays essential for several tasks such as urban and telecommunication planning (Königer and Bartel, 1998; Knapp and Coors, 2008) and disaster and rescue management (Kolbe et al., 2008) in particular flooding simulations (Schulte and Coors, 2008). Further, the data model CityGML (Gröger et al., 2012) supporting several levels of detail (LoD) is widely used in order to address semantics and exchange city data models. For many applications such as disaster management and visualization in particular in the navigation context, façades (LoD3 models) are important. In order to achieve an interpretation of the observations with good quality models providing prior knowledge are needed. In this context, formal grammars receive increasing attention (Musialski et al., 2012). For the reconstruction of façades Becker (2009) introduced a hybrid approach by the integration of inferred grammar rules into a non-parametric reconstruction process. Ripperda and Brenner (2009) used a probabilistic grammar for the description of façades. The production rules are expanded by the relative frequency of a single rule extracted from a manually collected rule database. Martinić and Van Gool (2013) proposed an approach to learn stochastic attributed grammar rules for two-dimensional façade generation and reconstruction from imagery data. Dehbi et al. (2016) presented a statistical relational based approach for automatic learning of an attribute grammar for building reconstruction.

Formal grammars demand a mechanism that fits data to underlying grammar rules. For this issue, several parse algorithms have been introduced. Han and Zhu (2009) used a two-staged approach in order to parse images based on a graph grammar. They assume that man-made objects are modelled by sub-objects using the grammar rules. Teboul et al. (2011) presented an approach for 2D shape grammar parsing of façades using reinforcement learning. Toshev et al. (2010) introduced a method for the detection and parsing of buildings from 3D laserscan data. Martinić et al. (2012) proposed a three-layered approach for façade

parsing. Schmittwilken (2012) presented an estimator and parser for the identification and interpretation of façades from LIDAR point clouds based on attribute grammars and a-priori probabilities. All these approaches expect offline data in order to parse the underlying grammar rules.

The presented work is part of a project which aims to build 3D building models during the acquisition of data by a lightweight autonomously flying UAV. It uses the visual information from cameras for navigation, obstacle detection, exploration and object acquisition leading to a high resolution dense 3D point cloud. Based on a semantically specified user inquiry the UAV delivers a point cloud stream improving the level of detail. Hence, an incremental model refinement is needed in order to deal with the continuous observation stream and to falsify or verify new model hypotheses.

The main contribution of this paper is the incremental parsing of weighted attribute context-free grammar (WACFG) for the automatic 3D building reconstruction. The parser refines models based on new observations. If a refinement is not possible, model hypotheses are falsified and rejected. In this way an adaptation of the available parse tree that has been acquired from previous interpretation steps is enabled. Parse trees are iteratively updated drawn upon ideas of graph transformation systems (Löwe, 1993; Heckel, 2006). Prior knowledge on façade parts is given by probability densities. In order to avoid distribution assumptions, the location and shape parameters of building parts are identified with a non-parametric kernel density estimation (KDE). During the interpretation, geometrical, semantic and topological consistency is ensured Gröger and Plümer (2012) using transaction rules.

The remainder of the article is structured as follows: An introduction of formal grammars is presented in section 2. Our method will be explained in detail in section 3. Hereby, the identification of façade parts is explained in subsection 3.1 while the algorithms for the incorporation of these identified parts based on the incre-

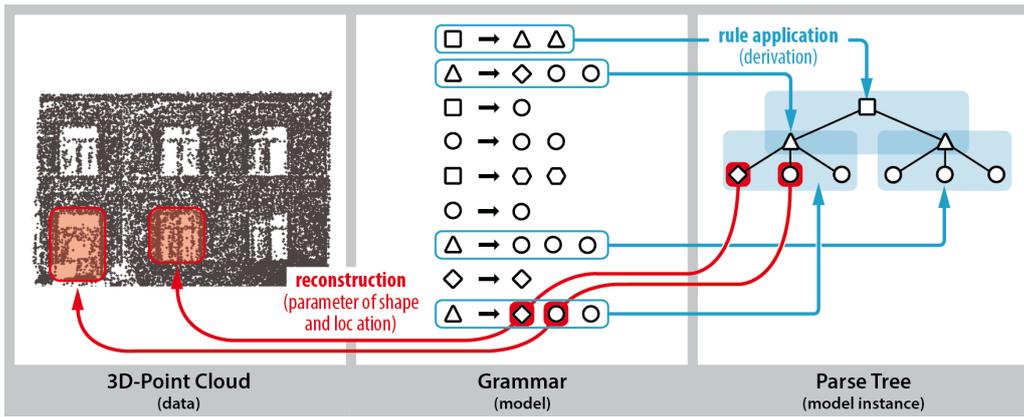


Figure 1: Offline parsing of 3D point clouds using formal grammars.

mental parsing of the attribute grammar rules are presented in subsection 3.2. The paper is summarized and concluded in section 4.

2 FORMAL GRAMMARS

Ever since formal grammars were introduced by Chomsky (1956, 1959) for reconstructing sentences of natural language, they have also been used to generate formal languages. The so-called Chomsky hierarchy distinguishes between four levels (type 0 to type 3) of formal grammars. A formal grammar G can be defined as the quadruple $\{S, N, T, P\}$ of a start symbol S , a set of non-terminals N represented by capitalised initials, a set of terminals T denoted by lower case initials and a set of production rules P .

A special case of formal grammars are context-free grammars which correspond to type 2 according to Chomsky’s hierarchy. Production rules appear in the form $A \rightarrow a$ where A is a non-terminal, and a is a sequence of terminals and non-terminals. This rule implies that each occurrence of the symbol A can be replaced by the string a .

as follows: *Building* as a start symbol is made of *Corpus* and *gableRoof*. In addition to the *Facade*, *Corpus* consists of *left* side, *right* side, and *back* side. *Facade* is built of *window* and *Entrance* which is made of *door* and *Stairs*. For the sake of simplicity of the example we assume that the façade has one single window. *Stairs* can be substituted either by one *Step* and *Stairs* as a sequence of steps or terminates with a *Step* which in turn consists of a *riser* (vertical rectangle) and a *tread* (horizontal rectangle). Likewise, a building can be represented by the following production rules:

<i>Building</i>	\rightarrow	<i>Corpus</i> <i>gableRoof</i>
<i>Corpus</i>	\rightarrow	<i>left</i> <i>Facade</i> <i>right</i> <i>back</i>
<i>Facade</i>	\rightarrow	<i>Entrance</i> <i>WinArray</i>
<i>WinArray</i>	\rightarrow	<i>WinArray</i> <i>window</i>
<i>Entrance</i>	\rightarrow	<i>door</i> <i>Stairs</i>
<i>Stairs</i>	\rightarrow	<i>Stairs</i> <i>Step</i>
<i>Stairs</i>	\rightarrow	<i>Step</i>
<i>Step</i>	\rightarrow	<i>riser</i> <i>tread</i>

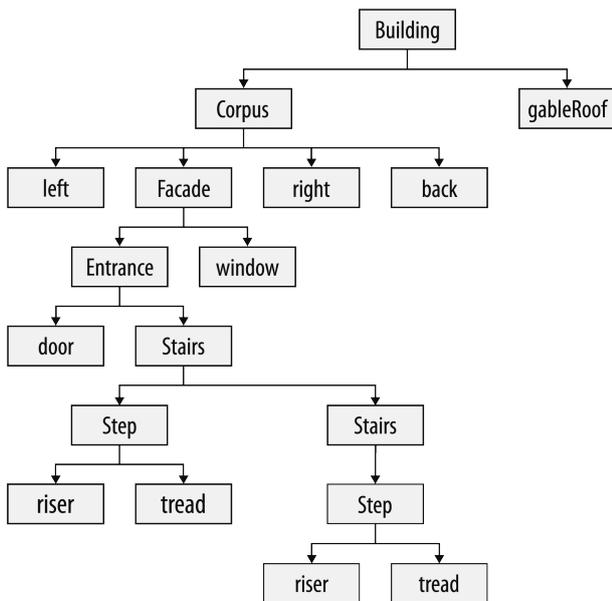


Figure 2: A possible derivation tree of a context-free gable roof building grammar.

As illustrated in figure 2, a possible context-free derivation which describes the structure of a gable roof building can be modelled

The set of non-terminals is $N = \{Building, Corpus, Facade, Entrance, WinArray, Stairs, Step\}$; the set of terminals is $T = \{gableRoof, left, right, back, window, door, riser, tread\}$. In order to express the likelihood of the application of rules, each rule is augmented by a weight or a probability defining a weighted or a probabilistic context-free grammar respectively (Geman and Johnson, 2002).

Motivated by requirements of early programming languages such as ALGOL and in order to cope with a restricted form of context sensitivity Knuth (1968, 1971) extended context-free grammars and introduced *attribute grammars*. Hereby, terminals and non-terminals are expanded by attributes, whereas production rules are extended by "semantic rules". The latter specify the constraints among the attributes. In our example these enable us to state that all steps within the same staircase have the same dimensions which cannot be expressed by context-free grammars. An extract of an attribute grammar of stairs is shown in listing 1.

The grammar symbols at the top are augmented by the attributes *height*, *depth* and *width*, which are used in the semantic rules R2(P2) to R4(P2) at the bottom in order to specify the identity between the shape parameters of risers and treads within the same stairs. Additionally, R1(P1) ensures that the new stairs consist of exactly one more step than the one before. The superscript indices n and $n-1$ are used to differentiate between multiple occurrences of the same symbol.

Listing 1: Excerpt of an attribute grammar for stairs

P1: Stairsⁿ⁻¹ → Step Stairsⁿ
 P2: Stairs → Step
 P3: Step → riser tread

R1(P1): Stairsⁿ.numberOfSteps =
 Stairsⁿ⁻¹.numberOfSteps + 1
 R2(P2): Step.width = Stairs.width
 R3(P2): Step.height = Stairs.height
 R4(P2): Step.depth = Stairs.depth
 ...

3 INCREMENTAL PARSING OF 3D POINT CLOUD

As yet, the parsing of formal grammars expect an offline data processing, especially for 3D point clouds. Based on a given grammar, the parser searches for a most likely parse tree with successively grammar rule application that matches the underlying data well (see figure 1). In this section we introduce a novel approach that enables to deal with changing data, e.g. provided by an UAV. The parser interprets the building parts within the measured point cloud and determines their descriptive location and shape parameters. Based on the detected parts a derivation tree is built that reflects the 3D model of the building. As soon as a new point cloud segment is provided, it is interpreted and the derivation tree of the building reconstruction is updated. This update of the tree is the main discriminative aspect of the incremental parser in comparison to an offline parser.

Algorithm 1: PARSE_FAÇADE – Weighted attribute grammar based incremental parsing of a 3D point cloud stream

Input: *Pts* ... 3D point cloud

Input: *WACFG* ... Weighted attribute context-free grammar

Input: *Tree* ... Parse tree (Can be empty)

Output: *Tree* ... most likely parse tree

```

1 Tree = Tree.getNodes();
2 [Plane, Ppts] =
  ESTIMATE_AND_ROTATE_MAIN_FAÇADE(Pts);
3 [Nodes] = FIND_PARTS(Pts, Plane, Ppts, Nodes);
4 [Tree] =
  TRANSFORM_PARSE_TREE(WACFG, Nodes, Tree);
    
```

The whole algorithm of the incremental parser (see algorithm 1) is subdivided into the two parts *building part detectors* and *tree transformation* which are executed consecutively. An overview of our approach is shown in figure 3. Beforehand, the main façade plane *Plane* is estimated and consequently an orthogonal rectification of the point cloud is performed.

The first part of the algorithm is responsible for the detection of the building parts within the measured point cloud *Pts*. Herewith, the descriptive location and shape parameters of each building part is determined. These parts build a set of nodes *Nodes*. This identification step is the core of section 3.1. The results of the interpreted dataset from the first part are employed to incorporate the observed building part in the model derived or predicted so far. A predicted tree is deduced based on a reasoning process as described in Loch-Dehbi and Plümer (2015). If there is no tree available, a new model is constructed otherwise. In both cases, the model consists in a parse tree. It describes the taxonomy and partonomy of a given façade reflecting semantics of each building part with respect to an a-priori learned weighted attribute

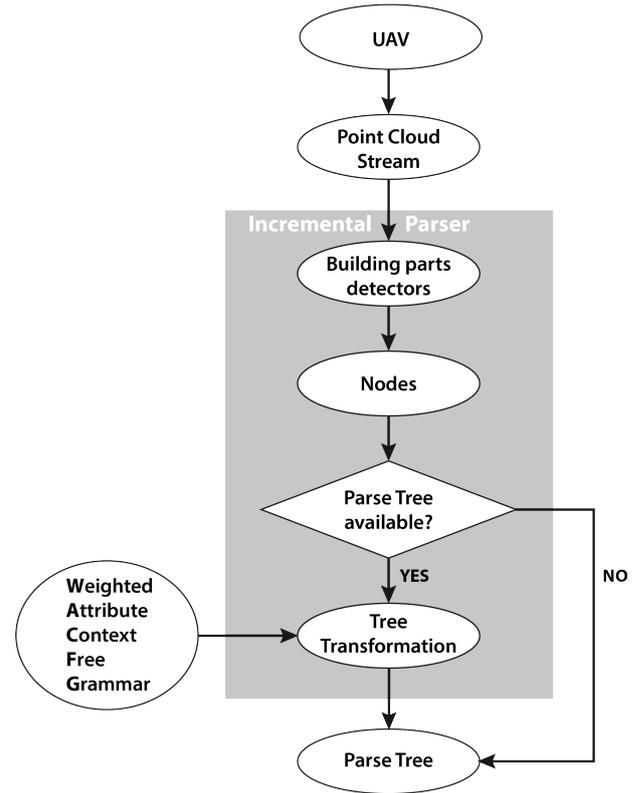


Figure 3: An overview of the incremental attribute grammar parsing of a 3D point cloud stream.

context-free grammar (*WACFG*) (Dehbi et al., 2016). The interpretation is performed as long as new observations (point cloud stream) are received. This incremental interpretation is discussed in section 3.2.

3.1 Identification of façade parts

This section describes the process of the determination of the façade parts. This is based on building part detectors tailored for each building part of interest. At first, the main façade is detected using a façade detector. This is performed with the use of a RANSAC estimation (Fischler and Bolles, 1981) based on the observed and geo-referenced points *Pts*. RANSAC delivers the estimated plane *Plane* together with the consensus set that supports the plane model. We assume that the input point cloud is taken from a fronto-parallel view of the world plane. Otherwise, an orthogonal rectification of the point cloud is performed according to the transformation in equation 1 leading to plane points *Ppts* (Förstner and Wrobel, 2016). Herewith, a rotation $R(N_p, N_{xz})$ using the vector pair (N_p, N_{xz}) along the axis $N_p \times N_{xz}$ takes place. N_p represents the normal of *Plane*, whereas N_{xz} is the normal of the xz plane of the world coordinate system. I , \tilde{N}_p and \tilde{N}_{xz} denote the identity matrix, the normalization of N_p and N_{xz} respectively.

$$R(N_p, N_{xz}) = I + 2\tilde{N}_p\tilde{N}_{xz}^T - \frac{1}{1 + \tilde{N}_{xz}\tilde{N}_p}(\tilde{N}_p + \tilde{N}_{xz})(\tilde{N}_p + \tilde{N}_{xz})^T \quad (1)$$

where $\tilde{N}_p = N(N_p)$, $\tilde{N}_{xz} = N(N_{xz})$ and $N_p \cdot N_{xz} \neq -1$.

In this manner an accurate detection of façade parts is enabled. To this end, further building part detectors are required. Openings such as windows or doors in the façades are interpreted as holes

in the point cloud. This corresponds to areas with a low point density. Therefore, a kernel density estimation (KDE) (Wand and Jones, 1994; Wang and Suter, 2004) is appropriate for the determination of the location and shape parameters of windows and doors. KDE enables a non-parametric estimation of a probability density function avoiding to make data distribution assumptions. The bandwidth of the used Gaussian kernel is adapted and optimized for each estimation depending on the underlying data (Benoudjit et al., 2002).

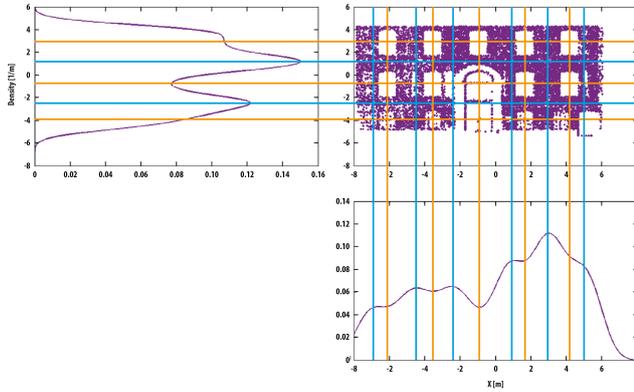


Figure 4: KDE of the consensus set of the façade main plane of the Frankenforst manor in Vinxel/ Germany. The extrema are incorporated for the floor-wise and column-wise segmentation of the façade and for the identification of openings.

The identification of the façade openings in algorithm 2 starts with a kernel density estimation of the x - and z -coordinates of the points Pts . Consequently the extrema along both directions are identified. The openings in the façade correspond to the intersection of the horizontal and vertical orange lines in figure 4. These denote the orthogonal projection of single minima mi_x and mi_z on the according axis leading to an approximative centroid and hence a location of an opening. Furthermore, in order to determine the shape parameters (width and height) a bounding box BB is defined over each centroid. Each bounding box BB is expanded in four directions (left, right, up, down) until a minimal point number is reached. Besides, the maxima ma_x and ma_z in the according direction are used for the robust determination of the shape parameters such as shown in figure 5. Further background knowledge – probability density functions – about shape parameters of building parts derived from a relational building database (RBDB) is also used in order to acquire a good size of the bounding box BB . The latter corresponds to the shape parameters of a single façade part and hence a leaf of a parse tree. These nodes are collected in a list $TempNodes$. As can be stated for the windows of the ground floor in figure 4, due to structured walls the derivation of the shape parameters is in general not straightforward. In this case, a local floor-wise re-estimation of the parameters is needed. To this end, a segmentation of the point cloud is performed based on the maxima values of the coordinates. This enables a recursive application of the previous steps for each floor based on a new bandwidth leading to a list $Nodes$ and their improved parameters. In order to discriminate between windows and doors, common architectural patterns as well as probability density functions derived from RBDB are used.

3.2 Incremental refinement of parse trees

This section deals with parse tree transformations based on the building parts detected and acquired from the first stage described in section 3.1. These parts are integrated in a parse tree with respect to the grammar rules of the $WACFG$. If there is no tree

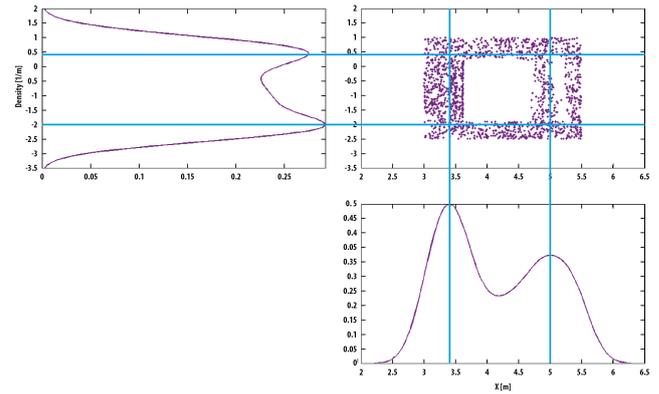


Figure 5: Local KDE in the neighbourhood of an opening in the main plane of the façade from figure 4. The maxima are incorporated for the identification of shape parameters of a window.

Algorithm 2: FIND_PARTS – KDE-based identification of façade parts

Input: Pts
Input: $Plane$
Input: $PPts$
Input: $Nodes$
Output: $Nodes$

- 1 Perform a KDE in x and z directions;
- 2 Calculate extrema in x and z directions;
- 3 Identify openings using minima mi_x and mi_z ;
- 4 Initialize bounding boxes BBs over centroids of the openings;
- 5 **for** $bb_i \in BBs$ **do**
- 6 Expand bb_i in 4 directions till minimal points or next (ma_x and ma_z in according direction) reached;
- 7 Define a $node_i$ using $size(bb_i)$;
- 8 Add $node_i$ to a list $TempNodes$;
- 9 $[FPts, Floors] = SEGMENT_IN_FLOORS(Pts, ma_x, ma_z, TempNodes)$;
- 10 **for** $f_i \in Floors$ **do**
- 11 $[FPlane, CFpts] = ESTIMATE_FLOOR_PLANE(FPts)$;
- 12 $[F_iNodes] = FIND_PARTS(Pts, FloorPlane, CFpts, TempNodes)$;
- 13 Update $TempNodes$ using F_iNodes ;
- 14 Let $Nodes = TempNodes$;

already available, e.g. at the beginning of a UAV mission, a parse tree $Tree$ is predicted using a stochastic and geometric reasoning based on few observations such as the façade width (Loch-Dehbi et al., 2013). If a parse tree already exists, the resulting building parts $Nodes$ from the building part detectors are regarded as leaves of the current parse tree. They represent together with the $WACFG$ and the parse tree $Tree$ the input of algorithm 3 for the parse tree transformation. Since the parse tree represents a directed acyclic graph, our approach is inspired by the ideas of *Graph transformation* (Löwe, 1993; Heckel, 2006) in order to adapt the tree structure based on the concept of transformation rules. The latter consist of a start and a target graph that are mapped using transitional rules.

The first step in order to integrate the new observations in the model $Tree$ is a diagnostic step. In this stage, nodes of the current tree $Tree$ that deviate from the new observed façade parts $Nodes$ are identified and labelled. To this end, the discrepancy is measured based on a *Hausdorff* distance based analysis (Edgar, 2007). In this step, only nodes in the new observed region are considered. The remaining nodes out of this scope remain unchanged and are maintained in the new tree. During the node comparison, it is differentiated between four states of nodes:

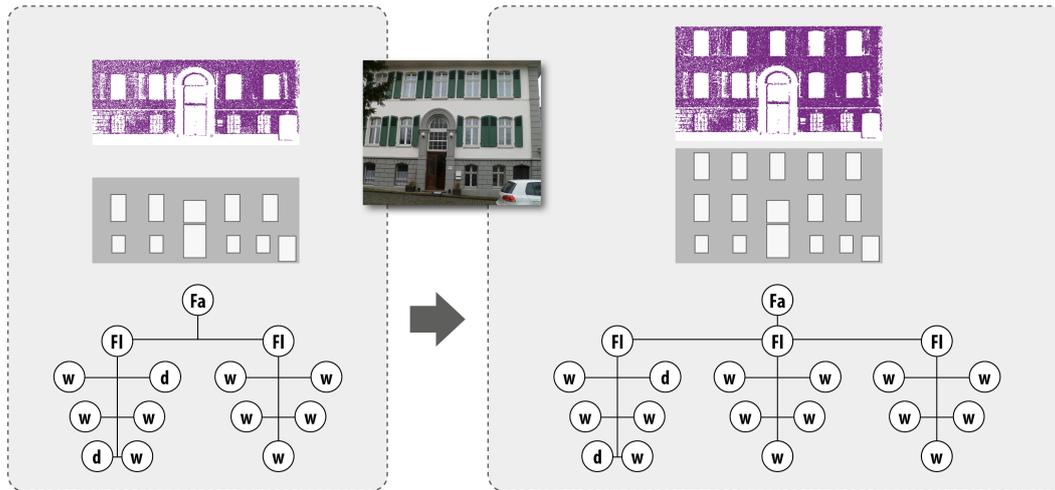


Figure 6: An incremental interpretation of the façade (Fa) of the Frankenforst manor in Vinxel/ Germany. On the left: Parse tree and the belonging resulted interpretation of two observed floors (Fl) with windows (w) and doors (d). On the right: An updated parse tree after the observation of a further floor.

Algorithm 3: TRANSFORM_PARSE_TREE – Construct or update a parse tree

Input: *WACFG* ... Weighted attribute context-free grammar

Input: *Nodes* ... Observed and interpreted building parts

Input: *Tree* ... Parse tree (Can be empty)

Output: *Tree* ... most likely parse tree

- 1 [*CNodes*] = FIND_CORRUPTED_NODES(*Tree*, *Nodes*);
- 2 [*CorrNodes*] = CORRECT_ATTRIBUTES(*Tree*, *CNodes*);
- 3 DELETE_NOT_SUPPORTEDED_NODES(*Tree*, *CNodes*);
- 4 [*Tree*] = PROPAGATE_ERROR_ON_DELETE(*AG*, *Tree*);
- 5 [*Tree*] = ADD_NODES(*Tree*, *CorrNodes*);
- 6 [*Tree*] = PROPAGATE_ERROR_ON_ADD(*WACFG*, *Tree*);

error-free, *error-prone*, *erroneous* and *new*. *Error-free* nodes are those still having the same location and shape parameters taking the observations uncertainty into account. In this case old nodes are confirmed by new observations and retained in the so far constructed tree. *Error-prone* nodes are characterised with nearby location values and one or more significantly different shape parameters. Nodes that are no longer supported by the new observations, are denoted as erroneous. Furthermore, *new* nodes are those without a correspondence in the tree.

The diagnostic step leads to a list of corrupted nodes *CNodes* and their attributes that will be corrected in a subsequently step leading to a list *CorrNodes* together with the new nodes. The corrected attributes that affect other parse tree nodes are considered, the according changes are then propagated based on the semantic rules of *WACFG*. The *WACFG* describes the taxonomic and partonomic structure of façades by a weighted context-free grammar (WCFG) and the substantial topological as well as geometric constraints which are described using Statistical Relational Learning (SRL) methods namely Markov Logic Networks (MLNs) (Dehbi et al., 2016).

In the next step, erroneous nodes from *CNodes* and possibly belonging parse tree parts are deleted. That would lead to conflicts between the current tree structure and the underlying grammar rules. In order to deal with such cases, a rule selection is performed based on the rule weights as well as the floor information derived during the identification stage in section 3.1 leading to a most likely rule. If the conflict cannot be resolved, the according parse tree part is deleted from *Tree*. The belonging

nodes are retained in a separate node list giving the possibility for a later consideration. Geometric and topological errors caused by deletion of nodes or tree parts are again considered and corrected using the attributed part of the *WACFG*. Constraints such as alignments of windows in floors or attributes identity are ensured using MLNs.

The list *CorrNodes* is now considered. Based on the floor-wise segmentation and the rule's likelihood, a rule selection is performed. Hereby, rules with higher node support are preferred. Based on this deduced parse tree structure, constraints are ensured in a same manner as in the deletion case using the semantic rules of the *WACFG*.

Figure 6 illustrates a scenario of an incremental interpretation of the main façade of the Frankenforst manor in Vinxel / Germany. At first, two floors have been observed. The according interpretation so far is shown as a parse tree on the left side of the figure. The incorporation of a new observed floor triggers an adaptation of the tree structure as shown in the parse tree on the right. The shape parameters are also adequately updated.

4 CONCLUSION

This paper introduced a novel approach for the automatic incremental refinement of façade models. The refinement process is based on a weighted attribute context-free grammar. In contrast to other approaches that expect an offline data interpretation, the presented method deals with continuous observations coming from an unmanned aerial vehicle (UAV). Prior knowledge on façade parts (e.g. windows) is given by non-parametric distributions using a kernel density estimation (KDE). These distributions are iteratively adapted using new observations. Accordingly, available parse trees acquired from previous interpretation steps are adapted with respect to the rules of the grammar. The falsification or rejection of model hypotheses is supported as well. As long as the level of detail is continuously improved, the geometrical, semantic and topological consistency is ensured.

Up to now, the presented approach paid attention to façades that can be split into floors. The underlying *WACFG* supports several splitting schemes, e.g. column-wise. In order to take more sophisticated façades into consideration an extension of the rule selection strategy is required.

ACKNOWLEDGEMENTS

This work was supported by the German Research Foundation (DFG) as part of the project "Mapping on Demand" (FOR 1505, PL 188/10-1). We thank Michael Kneuper for his support in preparing the illustrations. Our special acknowledgements are due to the anonymous reviewer.

REFERENCES

- Becker, S., 2009. Generation and application of rules for quality dependent faade reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing* 64(6), pp. 640 – 653.
- Benoudjit, N., Archangeau, C., Lendasse, A., Lee, J. A., Verleyesen, M. et al., 2002. Width optimization of the gaussian kernels in radial basis function networks. In: *ESANN*, Vol. 2, Citeseer, pp. 425–432.
- Chomsky, N., 1956. Three models for the description of language. *IRE Transactions on Information Theory* 2, pp. 113–124.
- Chomsky, N., 1959. On certain formal properties of grammars. *Information and Control* 2(2), pp. 137–167.
- Dehbi, Y., Hadiji, F., Gröger, G., Kersting, K. and Plümer, L., 2016. Statistical relational learning of grammar rules for 3d building reconstruction. *Transactions in GIS*.
- Edgar, G., 2007. *Measure, topology, and fractal geometry*. Springer Science & Business Media.
- Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* 24 (6), pp. 381–395.
- Förstner, W. and Wrobel, B., 2016. *Photogrammetric Computer Vision: Geometry, Orientation and Reconstruction*. Geometry and Computing, Springer International Publishing.
- Geman, S. and Johnson, M., 2002. Probabilistic grammars and their applications. *International Encyclopedia of the Social & Behavioral Sciences* 2002, pp. 12075–12082.
- Gröger, G. and Plümer, L., 2012. Transaction rules for updating surfaces in 3d gis. *ISPRS Journal of Photogrammetry and Remote Sensing* 69, pp. 134–145.
- Gröger, G., Kolbe, T. H., Nagel, C. and Häfele, K. H. (eds), 2012. *OGC City Geography Markup Language (CityGML) Encoding Standard, Version 2.0.0.*, Open Geospatial Consortium: OGC Doc. No. 12-019.
- Han, F. and Zhu, S.-C., 2009. Bottom-up/top-down image parsing with attribute grammar. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31(1), pp. 59–73.
- Heckel, R., 2006. Graph transformation in a nutshell. *Electron. Notes Theor. Comput. Sci.* 148(1), pp. 187–198.
- Knapp, S. and Coors, V., 2008. The use of eparticipation in public participation: The veps example. In: V. Coors, M. Rumor, E. M. Fendel and S. Zlatanova (eds), *Urban and regional data management, BALKEMA - proceedings and monographs in engineering, water and earth sciences*, Taylor & Francis, London, pp. 93–104.
- Knuth, E. D., 1968. Semantics of context-free languages. *Theory of Computing Systems* 2(2), pp. 127–145.
- Knuth, E. D., 1971. Top-down Syntax Analysis. *Acta Informatica* 1(2), pp. 79–110.
- Kolbe, T. H., Gröger, G. and Plümer, L., 2008. Citygml - 3d city models for emergency response. In: S. Zlatanova and J. Li (eds), *Geospatial information technology for emergency response*, International Society for Photogrammetry and Remote Sensing book series, Vol. 6, Taylor & Francis, London, pp. 257–274.
- Königer, A. and Bartel, S., 1998. 3d-gis for urban purposes. *GeoInformatica* 2(1), pp. 79–103.
- Loch-Dehbi, S. and Plümer, L., 2015. Predicting building facade structures with multilinear gaussian graphical models based on few observations. *Computer, Environment and Urban Systems* 54, pp. 68–81.
- Loch-Dehbi, S., Dehbi, Y. and Plümer, L., 2013. Stochastic reasoning for uav supported reconstruction of 3d building models. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Science XL-1 W 2*, pp. 257–261.
- Löwe, M., 1993. Algebraic approach to single-pushout graph transformation. *Theoretical Computer Science* 109(1), pp. 181–224.
- Martinović, A. and Van Gool, L., 2013. Bayesian grammar learning for inverse procedural modeling. In: *Computer Vision and Pattern Recognition (CVPR)*, 2013 IEEE Conference on, pp. 201–208.
- Martinović, A., Mathias, M., Weissenberg, J. and Van Gool, L., 2012. A three-layered approach to facade parsing. In: *Computer Vision–ECCV 2012*, Springer, pp. 416–429.
- Musialski, P., Wonka, P., Aliaga, D. G., Wimmer, M., van Gool, L. and Purgathofer, W., 2012. A Survey of Urban Reconstruction. In: *EUROGRAPHICS 2012 State of the Art Reports*, Eurographics Association, pp. 1–28.
- Ripperda, N. and Brenner, C., 2009. Application of a formal grammar to facade reconstruction in semiautomatic and automatic environments. In: *Proceedings of 12th AGILE Conference on GIScience*, Hannover, Germany.
- Schmittwilken, J., 2012. *Attributierte grammatiken zur rekonstruktion und interpretation von fassaden*. In: *Dissertation, Universität Bonn, IAPRS*, Vol. 39.
- Schulte, C. and Coors, V., 2008. Development of a citygml ade for dynamic 3d flood information,. In: *Joint ISCRAM-CHINA and GI4DM Conference on Information Systems on Information Systems for Crisis Management*, Harbin, China, 2008.
- Teboul, O., Kokkinos, I., Simon, L., Koutsourakis, P. and Paragios, N., 2011. Shape grammar parsing via reinforcement learning. *Computer Vision and Pattern Recognition (CVPR)* in Colorado Spring, USA.
- Toshev, A., Mordohai, P. and Taskar, B., 2010. Detecting and parsing architecture at city scale from range data. In: *Computer Vision and Pattern Recognition*, pp. 398–405.
- Wand, M. P. and Jones, M. C., 1994. *Kernel smoothing*. Crc Press.
- Wang, H. and Suter, D., 2004. Robust adaptive-scale parametric model estimation for computer vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26(11), pp. 1459–1474.