

# COMBINING VISIBILITY ANALYSIS AND DEEP LEARNING FOR REFINEMENT OF SEMANTIC 3D BUILDING MODELS BY CONFLICT CLASSIFICATION

O. Wysocki<sup>1\*</sup>, E. Grilli<sup>3</sup>, L. Hoegner<sup>1,2</sup>, U. Stilla<sup>1</sup>

<sup>1</sup> Photogrammetry and Remote Sensing, TUM School of Engineering and Design, Technical University of Munich (TUM), Munich, Germany - (olaf.wysocki, ludwig.hoegner, stilla)@tum.de

<sup>2</sup> Department of Geoinformatics, University of Applied Science (HM), Munich, Germany - ludwig.hoegner@hm.edu

<sup>3</sup> 3D Optical Metrology (3DOM) unit, Bruno Kessler Foundation (FBK), Trento, Italy - grilli@fbk.eu

## Commission IV, WG IV/9

**KEY WORDS:** 3D reconstruction, MLS point clouds, Semantic 3D building models, CityGML, Deep learning, LoD3 building models, Window and door reconstruction, Building models refinement.

## ABSTRACT:

Semantic 3D building models are widely available and used in numerous applications. Such 3D building models display rich semantics but no façade openings, chiefly owing to their aerial acquisition techniques. Hence, refining models' façades using dense, street-level, terrestrial point clouds seems a promising strategy. In this paper, we propose a method of combining visibility analysis and neural networks for enriching 3D models with window and door features. In the method, occupancy voxels are fused with classified point clouds, which provides semantics to voxels. Voxels are also used to identify conflicts between laser observations and 3D models. The semantic voxels and conflicts are combined in a Bayesian network to classify and delineate façade openings, which are reconstructed using a 3D model library. Unaffected building semantics is preserved while the updated one is added, thereby upgrading the building model to LoD3. Moreover, Bayesian network results are back-projected onto point clouds to improve points' classification accuracy. We tested our method on a municipal CityGML LoD2 repository and the open point cloud datasets: TUM-MLS-2016 and TUM-FAÇADE. Validation results revealed that the method improves the accuracy of point cloud semantic segmentation and upgrades buildings with façade elements. The method can be applied to enhance the accuracy of urban simulations and facilitate the development of semantic segmentation algorithms.

## 1. INTRODUCTION

Semantic 3D building models at levels of detail (LoD)1 and 2 are widespread<sup>1</sup> and commonly applied in urban-related studies (Biljecki et al., 2015). Such 3D models are frequently reconstructed using a combination of 2D building footprints and multi-view stereo (MVS) or airborne laser scanning (ALS) techniques, as in the example of more than eight million reconstructed buildings in Bavaria, Germany (Roschlaub and Batscheider, 2016). This reconstruction strategy enables detailed modeling of roof surfaces but renders generalized façades neglecting openings such as windows and doors, as shown in Figure 1b.

Reconstructing façade elements becomes a key factor enabling automatic 3D building modeling at LoD3, for which an increasing demand has been expressed by numerous applications including estimating heating demand (Nouvel et al., 2013), preserving cultural heritage (Grilli and Remondino, 2019), calculating solar potential (Willenborg et al., 2018), and testing automated driving functions (Schwab and Kolbe, 2019).

Since point clouds are deemed as one of the best data sources for 3D modeling, dense, street-level mobile laser scanning (MLS) point clouds appear to be especially suitable for at-scale façade reconstruction (Xu and Stilla, 2021). For this purpose, however, point clouds require semantic classification, which has been recently approached using machine and deep learning methods yielding promising results (Grilli and Remondino,

2020; Matrone et al., 2020). Yet, these methods can have limited accuracy when classifying objects that are translucent (e.g., windows) or have an inadequate amount of training data (e.g., doors). On the other hand, points' rays intersecting with 3D models can provide geometrical cues about possible façade openings, but without differentiating between classes, such as window, door, or underpass (Tuttas and Stilla, 2013).



Figure 1. Façade in a photo and a 3D building model: a) Oblique image (Google Earth, 2022), b) semantic building model at LoD2 (BayernAtlas, 2022).

In this paper, we present a strategy that combines both ray- and region-based methods for conflict classification (Wysocki et al., 2022a). This approach leads to refinement of both 3D building models and segmented point clouds' accuracy; our contributions are as follows:

- a CityGML-compliant strategy for upgrading LoD2 to LoD3 models by model-driven 3D window and door reconstruction;
- a method classifying conflicts between laser observations and 3D building models using deep learning networks;

\* Corresponding author

<sup>1</sup> <https://github.com/OloOcki/awesome-citygml>

- a method improving the semantic segmentation results of deep learning networks by analyzing ray-traced points and 3D building models.

## 2. RELATED WORK

The internationally used CityGML standard establishes the LoD of semantic 3D city objects (Gröger et al., 2012). One of the chief differences between LoD2 and LoD3 is the presence of façade openings in the latter. In our case, we search for absent façade elements in the input models using point clouds and then carry out their reconstruction. Therefore, we deem methods as related if they deal with detecting missing features (Section 2.1) and façade reconstruction using point clouds (Section 2.2).

### 2.1 Visibility analysis using point clouds

Hebel et al. (2013) employ visibility analysis to detect changes between different point cloud epochs. The method addresses the uncertainty of ALS measurements using the Dempster–Shafer theory (DST). Ray tracing on a voxel grid is introduced to identify *occupied*, *empty*, and *unknown* states per epoch. Based on the epochs comparison, they distinguish: *consistent*, *disappeared*, and *appeared* states.

Visibility analysis is utilized to remove dynamic objects from point clouds, too (Gehring et al., 2017). MLS observations' rays are traced on an efficient octree grid structure introduced by Hornung et al. (2013). Each traced ray provides occupancy probabilities, which are accumulated per voxel using the Bayesian approach. Moving objects are removed based on decreasing occupancy probability of ray-traversed voxels.

A multimodal approach to visibility analysis is proposed by Tuttas et al. (2015). They investigate how to monitor the progress of a construction site using photogrammetric point clouds and building information modeling (BIM) models. The Bayesian approach and the octree grid structure is employed to analyze the points' rays and vector models. The as-is (point cloud) to as-planned (3D model) comparison differentiates between *potentially built*, *not visible*, and *not built* model parts.

In our previous work (Wysocki et al., 2022a), we introduced visibility analysis to refine semantic 3D building models with underpasses using MLS point clouds. The method compares ray-traced points with building objects on an octree grid in a probabilistic fashion. Contours of underpasses are identified based on an analysis of conflicts between laser observations and building models, supported by vector road features.

### 2.2 Façade openings reconstruction using point clouds

Substantial research effort has been devoted to methods using images for façade segmentation (Szeliski, 2010). Nevertheless, 2D images require additional processing to enable semantic 3D reconstruction. 3D point clouds, however, provide an immediate 3D environment representation, which makes them one of the best datasets for urban mapping (Xu and Stilla, 2021).

When analyzing laser observations, openings are often assumed to represent holes due to their translucent characteristic or face-intruded position (Tuttas and Stilla, 2013; Fan et al., 2021). For example, windows are detected based on building interior points, which imply opening existence (Tuttas and Stilla, 2013).

Borders of openings are delineated based on the ray tracing of interior points and the detected façade plane in point clouds.

Zolanvari et al. (2018) propose a slicing method to identify openings using horizontal or vertical cross-sections. The method finds façade planes using the RANSAC algorithm and removes noisy points based on their deviations from the planes. Gaps occurring in horizontal or vertical cross-sections delineate possible openings.

Layout graphs are proposed by Fan et al. (2021) to identify façade structures. Spatial relations among detected objects are encoded and exploited by the Bayesian framework to deduce the whole façade layout.

Recently, however, data-driven methods based on machine and deep learning approaches have provided promising results for classifying point clouds, especially when using the self-attention mechanism (Zhao et al., 2021). These great strides have influenced façade segmentation of point clouds, too (Grilli and Remondino, 2020; Matrone et al., 2020). Modified versions of the DGCNN deep learning architecture are proposed to classify façade elements in point clouds (Grilli and Remondino, 2020). The method employs features stemming from machine learning approaches to improve deep learning network accuracy.

Little research attention has been given to investigating the automatic upgrade of LoD2 to LoD3 building models using point clouds, except, to the best of our knowledge, our previous works refining overall façade geometry (Wysocki et al., 2021a,b) and reconstructing underpasses (Wysocki et al., 2022a). However, related work is proposed by Hensel et al. (2019) for detecting and reconstructing openings, not by point clouds but by exploiting the textures of semantic city models. They apply the Faster R-CNN deep neural network to identify the bounding boxes of windows and doors on textured CityGML building models. To minimize inaccuracies in the alignment of openings, they apply mixed-integer linear programming. Then, bounding boxes serve as reconstructed opening elements in LoD3 building models.

## 3. METHODOLOGY

In contrast to our previous work devoted to refining building models with underpasses (Wysocki et al., 2022a), in this paper we focus on detecting and reconstructing outstanding façade openings, such as windows and doors. Moreover, our method refines point cloud segmentation by back-projecting classified conflicts onto the input point clouds.

As presented in Figure 2, the method evaluates and assigns uncertainties to the input datasets (Section 3.1). While a neural network is trained on points representing façade elements (Section 3.2), the points ray tracing process performs probabilistic classification of a scene into *occupied*, *empty*, and *unknown* voxels (Section 3.3). Subsequently, labeled voxels are compared to segmented points to derive *static* and remove *dynamic* points in voxels (Section 3.5). The voxels are also compared to vector 3D models to identify *confirmed*, *empty*, and *unknown* voxel labels (Section 3.4). If *conflicted* and *static* features exist, probabilistic classification is carried out, where a Bayesian network identifies *unmodeled openings* and *other objects* (Section 3.6). These are back-projected to the point cloud, refining its segmentation accuracy. If the Bayesian network detects

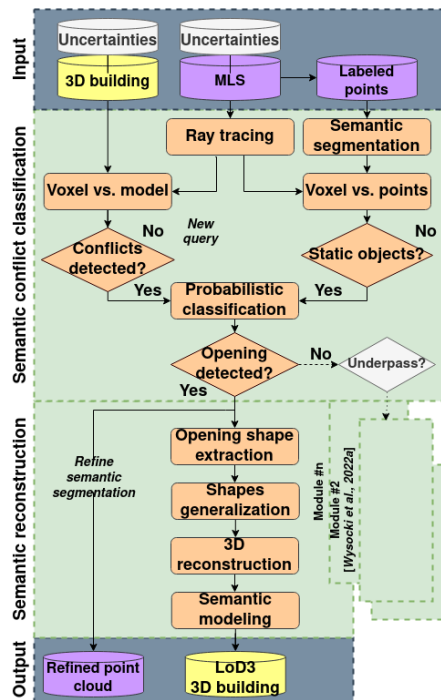


Figure 2. Workflow of the presented method

windows or doors, shape extraction is conducted (Section 3.7); otherwise, another module can be triggered, such as the underpass reconstruction (Wysocki et al., 2022a). Opening shape extraction is followed by shape generalization, which delineates fitting borders for 3D reconstruction (Section 3.7). Window and door 3D models are automatically fitted to shapes based on the respective geometry and opening class (Section 3.9). Afterward, unchanged and new semantics are assigned to geometries, following the CityGML standard for LoD3 (Gröger et al., 2012).

### 3.1 Data with uncertainties

Uncertainties in laser measurements and vector objects can stem from various sources, such as imprecise metadata, data transformations, and acquisition techniques. Uncertainties are application-dependent, too. Therefore, the proposed façades refinement involves uncertainties concerning the global positioning accuracy of point clouds and building models. To quantify these uncertainties, we introduce the confidence interval (CI), which is estimated using the confidence level (CL), its associated z value ( $z$ ), standard deviation ( $\sigma$ ), and mean ( $\mu$ ).

Let  $\sigma_1$  be the location uncertainty of point clouds, and  $\sigma_2$  the location uncertainty of 3D model walls. These are estimated based on the assumed point cloud global registration error  $e_1$  and the global location error of 3D model walls  $e_2$ . Then, the façade’s CI is calculated based on  $\sigma = \sqrt{\sigma_1^2 + \sigma_2^2}$ . The maximum upper and lower bounds are given by  $[\mu_i - 2\sigma_i, \mu_i + 2\sigma_i]$ , when assuming operating in the L1 norm and Gaussian distribution (Suveg and Vosselman, 2000).  $CL_1$  and  $CL_2$  quantify the operator’s confidence level in true-value deviations for laser measurements and 3D model walls, respectively. Depending on the CL value, corresponding  $z_i$  values are assumed. The division of  $\mu_i$  by  $z_i$  estimates the standard deviation  $\sigma_i$  value (Hazra, 2017).

### 3.2 Semantic segmentation

The goal of semantic segmentation is to divide a point cloud into several subsets based on the semantics of the points. Following Wysocki et al. (2022b) and as shown in Figure 3, eight relevant classes for façade segmentation and reconstruction tasks are considered: *arch* (dark blue), *column* (red), *molding* (purple), *floor* (green), *door* (brown), *window* (blue), *wall* (beige), and *other* (gray).

The segmentation is performed using a modified Point Transformer self-attention network (Zhao et al., 2021) extended by the use of geometric features improving the network performance, such as *height of the points*, *roughness*, *volume density*, *verticality*, *omnivariance*, *planarity*, and *surface variation*. The last three mentioned features are based on the normalized eigenvalues  $\lambda_i$  ( $\lambda_1 > \lambda_2 > \lambda_3$ ), which are derived from the 3D point coordinates within a considered spherical neighborhood  $r_i$  (Weinmann et al., 2013; Grilli and Remondino, 2020).

Finally, using a softmax output layer, we obtain an output vector of probabilities for each predicted class, which becomes fundamental for running our conflict classification (Section 3.5).



Figure 3. Semantic segmentation result for the façade in Figure 1.

### 3.3 Ray tracing

Points ray tracing is performed to identify absent structures in existing 3D building models (Figure 4). To enable comparison between these modalities, we employ a 3D occupancy grid. The grid adapts its size to the input data since it utilizes an octree structure. 3D voxels are the octree structure’s leaves, and their size  $v_s$  is selected based on the relative accuracy of laser observations.

Every laser observation is traced from the sensor position  $s_i$ , following the orientation vector  $r_i$ , to the reflecting point  $p_i = s_i + r_i$ . Voxels containing  $p_i$  are labeled as *occupied* (blue),

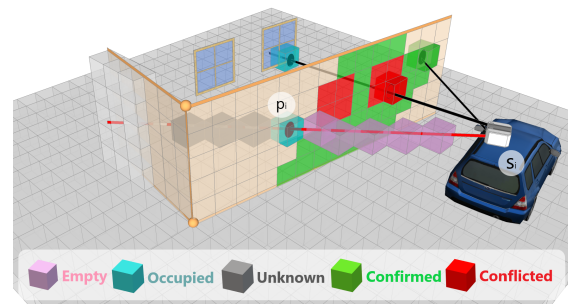


Figure 4. Points ray tracing on a vector-populated octree grid from the sensor position  $s_i$  to the hit point  $p_i$ . Adapted from (Wysocki et al., 2022a).

those traversed by a ray as *empty* (pink), and the untraversed ones as *unknown* (gray). The labels are assigned based on a probability score that considers multiple laser observations  $z_i$ , which are updated using prior probability  $P(n)$  and previous estimate  $L(n|z_{1:i-1})$ . Final score is controlled using log-odd values  $L(n)$  and clamping thresholds  $l_{min}$  and  $l_{max}$  (Hornung et al., 2013; Tuttas et al., 2015):

$$L(n|z_{1:i}) = \max(\min(L(n|z_{1:i-1}) + L(n|z_i), l_{max}), l_{min}) \quad (1)$$

where

$$L(n) = \log\left[\frac{P_n}{1 - P(n)}\right] \quad (2)$$

The grid is vector-populated by inserting 3D model faces and their quantified uncertainties (Section 3.1). Hence, each face has an assigned façade's maximal deviation range (upper CI) and its confidence level (CL). Ultimately, the grid's 3D voxels include attributes such as location, size, as well as state probability stemming from laser observations and a building model.

### 3.4 Voxels to model comparison

As shown in Figure 4, each voxel is analyzed in relation to its intersection with a façade: *Occupied* voxels that intersect with façades are labeled as *confirmed* (green); *empty* voxels that intersect with façades are labeled as *conflicted* (red); *unknown* voxels hold their status, as they represent unmeasured space. Voxels are projected onto the intersected façade, forming the

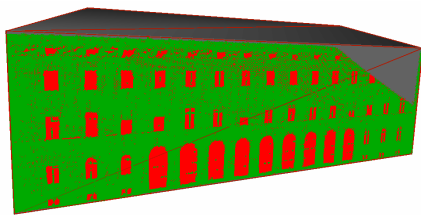


Figure 5. Texture representing *confirmed*, *conflicted*, and *unknown* areas identified on a façade

*model comparison* texture map layer with the respective voxel labels: *confirmed*, *conflicted*, and *unknown* (Figure 5). The cell spacing of a texture map follows the projection of the voxel grid to the plane.

### 3.5 Voxels to point cloud comparison

Ray tracing provides physical, per-voxel occupancy indicators, while semantic segmentation yields educated, per-point semantic classes. Both of these sources provide their semantic information with a probability measure. The fusion of voxels and points is conducted to transfer per-point semantic classes to occupancy voxels and suppress the impact of dynamic points (Figure 6). The rationale behind this fusion is that *static*, occupied voxels (yellow) are building-related; *dynamic*, unoccupied voxels (gray) represent moving objects, such as pedestrians or cars, and can be suppressed by multiple laser observations, as shown by Gehrunig et al. (2017) and in Figure 7.

Semantic points are inserted into the voxel grid to enable comparison between the two representations. Then, the median probability score  $P(B)$  is derived from point classes within each voxel. The occupancy probability  $P(A)$  and the median probability of each class  $P(B)$  are two dependent events, for which the existence probability score  $P_{ex}$  is calculated

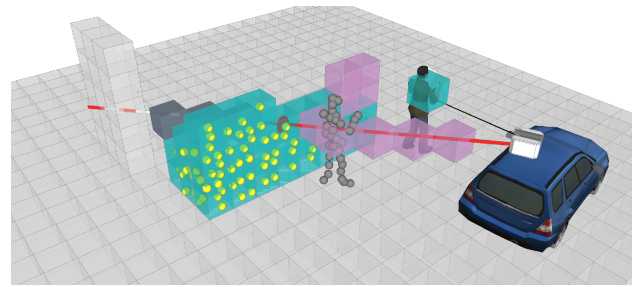


Figure 6. Fusion of voxels with per-point semantic information (yellow), while suppressing dynamic points (gray) using probability score and measurements accumulation.

$P_{ex}(A \cap B) = P(A) \cdot P(B|A)$ . Voxels are deemed as *static* if the existence probability score  $P_{ex}$  is greater than the static threshold probability:  $P_{ex} \geq P_{static}$ ; otherwise, voxels represent the *dynamic* state. Points within *dynamic* voxels are re-

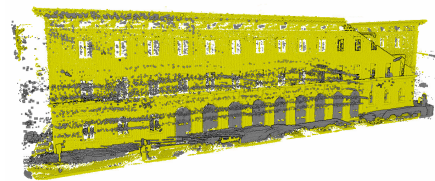


Figure 7. *Dynamic*, noisy points (gray) separated from *static*, building-related points (yellow).

labeled to the *other* class and are back-projected to the input point cloud. The *static* voxels obtain the point class that scores the greatest probability  $P(B)$  within a voxel (Figure 7).

*Static* voxels with semantics are projected onto the façade, forming the *points comparison* texture map layer with labels corresponding to the classes, as shown in the example of windows (orange) in Figure 8. As in the *model comparison* layer (Section 3.4), the cell spacing of a texture map follows the projection of the voxel grid to the plane.

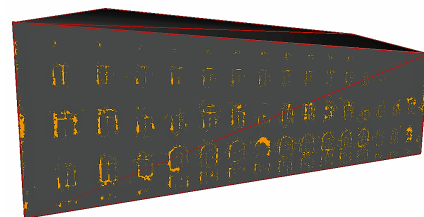


Figure 8. Texture showing one of the *static* voxel classes *window* on a façade

### 3.6 Probabilistic classification: the Bayesian approach

*Model comparison* and *points comparison* textures are utilized to identify façade openings using a Bayesian network (BayNet). The network estimations are also back-projected onto semantic point clouds to enhance their segmentation accuracy.

As shown in Figure 9, the designed BayNet comprises: one target (red), two input (yellow), one decision (blue), and two output nodes (green). Each directed link represents a causal relationship between the  $X$  and the  $Y$  nodes. The conditional probability table (CPT) prescribes weights for each state and node combination (gray). The target, *opening* state is calculated



using the joint probability distribution  $P(X, Y)$  and the CPT. The marginalization process is used to calculate the probability of the target node  $Y$  being in the *opening* state  $y$ . The process sums conditional probabilities of the states  $x$  stemming from parent nodes  $X$  (Stritih et al., 2020). Since the network con-

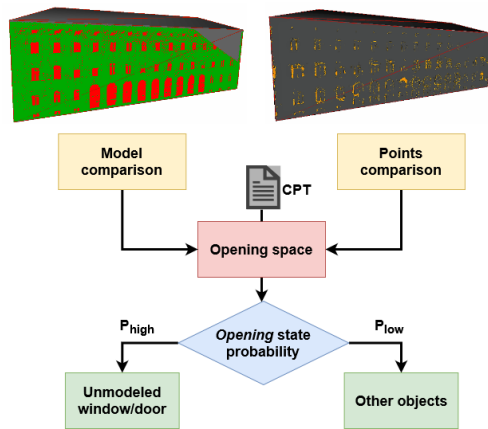


Figure 9. Input nodes (yellow) and CPT estimate the probability of opening space (red) in BayNet: if (blue) the probability is high, doors or windows are unmodeled; otherwise, areas indicate other objects (green).

sists of texture layers with state probabilities, the data evidence represents the so-called soft evidence (Stritih et al., 2020). In an inference process, soft evidence is added to update the joint probability distribution. This process provides the most likely node states by estimating the posterior probability distribution (PPD).

Pixel classes from the *model comparison* and *points comparison* textures form clusters if they have a neighbor in any of the eight directions of the pixel. The co-occurring *conflicted*, *window*, and *door* cluster classes, lead to a high probability of unmodeled openings. This output is used for further opening 3D modeling and is back-projected onto segmented point clouds as either the *window* or *door* class. On the other hand, co-occurring *confirmed*, *window*, and *door* clusters, lead to a low probability of existing openings. These clusters are also back-projected to improve the accuracy of semantically segmented point clouds: either as the *molding* class, if close to an opening; or otherwise as the *wall* class. The low probability  $P_{low}$  and the high probability  $P_{high}$  labels are assigned to clusters based on the probability threshold  $P_t$ :  $P_{high} > P_t \geq P_{low}$ .

### 3.7 Openings shape extraction

The high probability clusters  $P_{high}$  are extracted from a Bayesian probability texture as opening shape candidates. Adding to existing shape indices (Basaraner and Cetinkaya, 2017), we introduce the completeness index, which measures the  $r_{cp}$  ratio of outer shape area to inner-holes area. The candidates are rejected if their area is smaller than the chosen area threshold value  $b_s$  and if their completeness index score  $r_{cp}$  is smaller than  $r_{cpt}$ .

### 3.8 Openings shape generalization

Yet, the extracted candidates can still display distorted, noisy shapes. Morphological opening operation is applied to minimize the effect of spiky and weakly connected contours. Subsequently, these shapes are generalized to minimum bounding

boxes, for which a modified rectangularity index (Basaraner and Cetinkaya, 2017) is calculated. The modification considers relation of the bounding box sides  $a$  to  $b$ , where outliers are rejected based on the upper  $PE_{up}$  and lower  $PE_{lo}$  percentiles of the index score.

### 3.9 Model-driven 3D reconstruction

Identified bounding boxes are used as fitting boundaries for window and door 3D models, which are loaded from a pre-defined library. The opening models' coordinate origin is erased and then placed in the bottom left corner of a model. The offset to global coordinates is calculated between the opening model origin and the bottom left corner of the respective bounding box. After the shift, the rotation is performed as a difference between the façade's face orientation and opening model orientation. Aligned 3D models are scaled to fit bounding box boundaries, as presented in Figure 10 and in Figure 11.

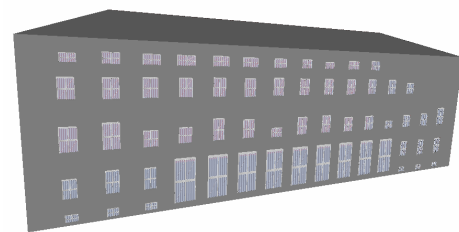


Figure 10. Reconstructed 3D windows for façade A

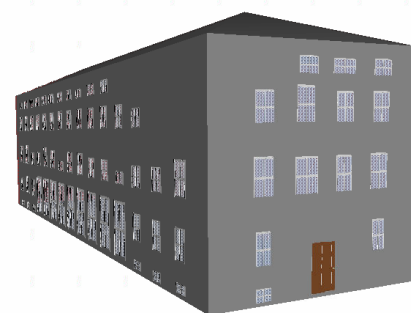


Figure 11. Reconstructed 3D windows and doors for façade B

### 3.10 Semantic modeling

Since 3D solid libraries of openings are employed for 3D reconstruction, we opt to model them as solid geometries, too, following the CityGML encoding recommendation (Special Interest Group 3D, 2020). Based on the identified opening class, windows and doors are assigned to the respective CityGML *Window* and *Door* classes; as such, they link to the building entity (Gröger et al., 2012). The unchanged semantics of input elements is preserved, except for the LoD which is upgraded to LoD3.

## 4. EXPERIMENTS

### 4.1 Datasets

The method was tested using MLS point clouds and governmental CityGML building models at LoD2 representing the Technical University of Munich (TUM) main campus, Munich, Germany.

The acquired LoD2 building models were created using 2D cadastre footprints and aerial measurements (Roschlaub and Batscheider, 2016). LoD3 door and window models were extracted from the manually modeled, open LoD3 city model of Ingolstadt, Germany<sup>2</sup>. The open TUM-MLS-2016 dataset (Zhu et al., 2020) was transformed into the global coordinate reference system (CRS) and used to perform point cloud ray tracing. The TUM-FAÇADE dataset was deployed for training, as it comprises façade-annotated point clouds (Wysocki et al., 2022b). For computational reasons, we subsampled the original dataset removing all the redundant points within a 5 cm distance. In this way, we compressed an initial dataset of about 118 million points to a still resolute but lightweight version of about 10 million points. The subsampled point cloud was divided into 70% training and 30% validation sets (Figure 12). Additionally, 17 available classes were consolidated into seven

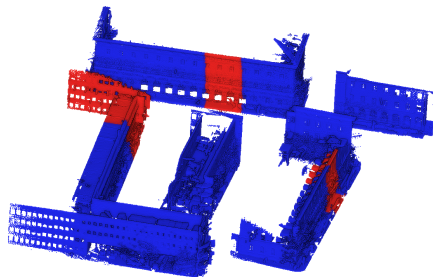


Figure 12. The TUM-FAÇADE benchmark divided into training (blue) and validation (red) sets for the segmentation experiments.

representative façade classes: *molding* was merged with *decoration*; *wall* included *drainpipe*, *outer ceiling surface*, and *stairs*; *floor* comprised *terrain* and *ground surface*; *other* was merged with *interior* and *roof*; *blinds* were added to *window*; while *door* remained intact.

#### 4.2 Parameter settings

The uncertainties of the true façades location were estimated considering the global registration error of MLS point clouds and building models: For point clouds these were set to  $e_1 = 0.3 m$ ,  $\mu = 0.15 m$ ,  $CL_1 = 90\%$ , and  $z_1 = 1.64$ ; for building models were set to  $e_2 = 0.03 m$ ,  $\mu = 0.015 m$ ,  $CL_2 = 90\%$ , and  $z_1 = 1.64$ . This yielded the façades' upper CI score of 0.2 m and  $CL = 90\%$ .

Ray casting was employed on a grid with the voxel size set to  $v_s = 0.1 m$  considering: opening size, the point clouds density, and their relative accuracy. The voxels were initialized with a uniform prior probability of  $P = 0.5$ . Log-odd values were set to  $l_{occ} = 0.85$  for *occupied* and  $l_{emp} = -0.4$  for *empty* states, corresponding to  $P_{occ} = 0.7$  and  $P_{emp} = 0.4$ , respectively. Clamping parameters were set to  $l_{min} = -2$  and  $l_{max} = 3.5$ , corresponding to  $P_{min} = 0.12$  and  $P_{max} = 0.97$ , respectively, following (Tuttas et al., 2015; Hornung et al., 2013); an exemplary implementation is provided in our repository<sup>3</sup>. For the fusion of voxels and points, the static threshold was set to  $P_{static} = 0.7$ , while the *empty* voxels occupancy probability was fixed to 0.4 for processing acceleration.

As regards the semantic segmentation procedure, taking into consideration the main characteristic of the buildings, the classes to be detected, and following Grilli et al. (2019), we

<sup>2</sup> <https://github.com/savenow/lod3-road-space-models>

<sup>3</sup> <https://github.com/OloOcki/conflict-mls-citygml-detection>

identified 0.8 m as optimal neighborhood search radius  $r_i$  for the features *roughness*, *volume density*, *omnivariance*, *planarity*, and *surface variation*, while 0.4 m for *verticality*.

The proposed BayNet has two input soft evidence layers: *points comparison* and *model comparison* textures. These had associated confidence levels, which scored 70% and 90% for *point* and *model comparison* layers, respectively. The opening state probability was defined by the probability threshold:  $P_t = 0.7$ .

The opening candidates' area threshold value  $b_s$  was set to  $0.3 m^2$ , while completeness threshold score  $r_{com_t}$  was set to 0.1, to suppress noisy, patchy clusters. The over-elongated bounding boxes were suppressed by calculating the modified rectangularity index, where the upper  $PE_{up}$  and lower  $PE_{lo}$  were set to the 95th and 5th percentile, respectively.

#### 4.3 Validation of improved semantic segmentation

Semantic segmentation results were validated on unseen ground-truth point clouds of the TUM-FAÇADE dataset. For evaluation, we use the overall accuracy (OA); F1 score per class; and average: precision ( $\mu P$ ), recall ( $\mu R$ ), F1 score ( $\mu F1$ ), and intersection over union ( $\mu IoU$ ). The *arch* and *column* classes were omitted in the validation, since they were absent in the ground-truth building. As shown in Table 1, for the baseline of the validation served the Point Transformer (PT) network (Zhao et al., 2021). The presented feature-extended version of the PT network (PT+Ft.) served as an input for the proposed conflict classification (CC) method.

Method	OA	$\mu P$	$\mu R$	$\mu F1$	$\mu IoU$	molding	floor	door	window	wall	other
PT	63.4	58.5	53.2	53.8	41.4	48.2	84.8	1.5	48.7	81.5	58.4
PT+Ft.	72.6	<b>66.4</b>	66.7	63.3	52.0	68.3	92.9	5.1	54.6	86.6	72.7
CC	<b>75.3</b>	65.9	<b>71.9</b>	<b>65.4</b>	<b>52.9</b>	67.6	86.3	13.6	59.8	85.9	79.1

Table 1. Overall accuracy (OA), average recall ( $\mu R$ ), average F1 ( $\mu F1$ ), average intersection over union ( $\mu IoU$ ), and F1 scores per class, given in percents.

#### 4.4 Validation of openings reconstruction

Reconstructed openings were validated using manually modeled ground-truth building openings (Table 3). Detection rate was calculated based on the on-site inspection of all existing façade openings (AO) and measured openings (MO) by laser scanner (Table 2). The validation was performed for façades A, B, and C, shown in Figure 10, Figure 11, and Figure 15, respectively.

Façade	A	B	C	Total
AO	66	17	20	<b>101</b>
MO	60	17	10	<b>87</b>
D	60	15	6	<b>81</b>
TP	60	15	5	<b>80</b>
FP	0	0	1	<b>1</b>
FN	6	2	12	<b>21</b>
DR-AO	90.1%	88.2%	27.8%	<b>79.2%</b>
FR-AO	0%	0%	16.7%	<b>1.2%</b>
DR-MO	100%	88.2%	50%	<b>91.9%</b>
FR-MO	0%	0%	16.7%	<b>1.1%</b>

Table 2. Detection rate for all openings (DR-AO) and measured openings (DR-MO) and the respective false alarm rate (FR-AO and FR-MO) for façades A, B, and C (D = detections, TP = true positives, FP = false positives, FN = false negatives, for all openings).

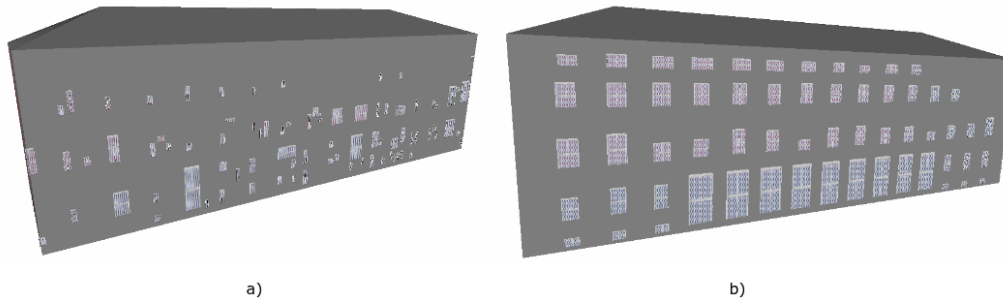


Figure 13. Windows reconstructed using: a) only neural networks output, b) the proposed method.

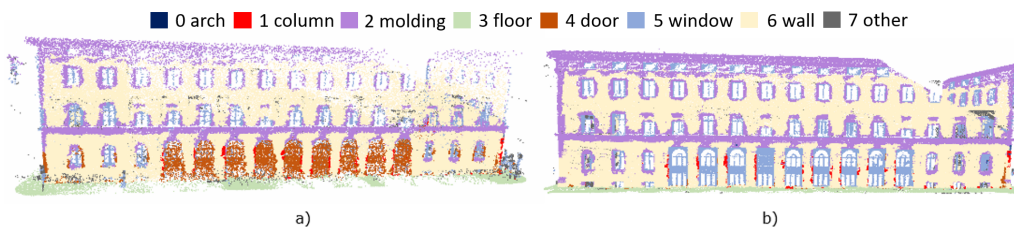


Figure 14. Semantic segmentation based on: a) only a neural network, b) the proposed method.

Façade	A	B	C	Total
<i>mIoU</i>	88.4%	94.8%	97.2%	<b>89.6%</b>
$\mu$ IoU	79.1%	83.4%	85.7%	<b>80.3%</b>

Table 3. Validation of reconstructed openings using median (*mIoU*) and average intersection over union ( $\mu$ IoU).

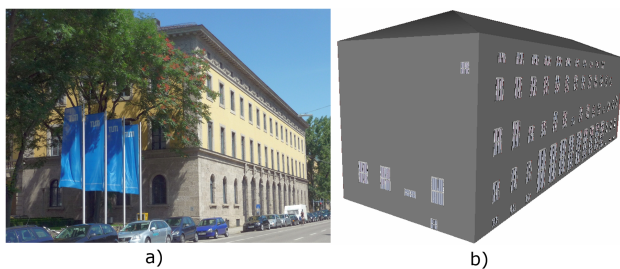


Figure 15. Openings reconstruction for façade C in the presence of occluding objects: a) Photo, b) refined façade.

## 5. DISCUSSION

Experiments revealed promising results for refining of both building models and classified point clouds. As presented in Table 2, façade openings were correctly detected with an estimated 92% detection rate to total measured openings (DR-MO) and 79% detection rate for all openings (DR-AO). Roughly 1% false alarm rate for both measured (FR-MO) and all openings (FR-AO) was noted.

The experiments corroborate that DR was dependent on the density of measurements per façade: for the densely covered façade A it estimated 90% DR-AO and detected 100% of measured openings (DR-MO); for the highly occluded side-façade C it estimated 28% and 50%, respectively (see Table 2 and Figure 15).

The method improves significantly reconstruction performance in comparison to the one conducted only on segmented point clouds of the baseline PT architecture, as shown in Figure 13. When compared to the ground-truth openings, the proposed reconstruction reached roughly 90% accuracy (Table 3); yet, the

method is limited when windows are partially measured (e.g., blinds before windows), as exemplified by several windows in the third row in Figure 13b.

The back-projected, classified conflicts increased accuracy of semantic point cloud segmentation by approximately 12% (Table 1). Note that the precision and intersection over union score for CC remained similar to the PT+Ft. score, while F1 score for *floor* dropped by about 6%. Remarkably, the proposed CC method improves segmentation of *window*, *door*, and *other* classes by approximately 11%, 12%, and 21%, respectively.

## 6. CONCLUSION

Our work has led us to the conclusion that refinement is a promising alternative to a from-scratch reconstruction. The refinement preserves input semantics, minimizes model-specific planarity issues, and enables consistent city model updates. Moreover, existing LoD3 elements can be extracted and directly employed as refinement features for buildings at lower LoDs.

The validation presents that the method reaches a high accuracy of 92% in detecting observable windows and a low false alarm rate score of approximately 1%. Refined point clouds also score low false negative rate, which is indicated by a high recall score of 79%. This trait of our method could be of particular importance for feature-dependent applications, where robustness is favored over visualization, such as in simulations of automated driving functions (Schwab and Kolbe, 2019). On the other hand, façade occlusions and a laser range could limit the method's applicability for visualization-oriented purposes, where a further prediction of unseen objects could be employed.

Experiments corroborate that combining visibility analysis with a region-based approach improves segmentation accuracy. In the future, we plan to embed occupancy information directly into the training of the deep neural network. Furthermore, including radial point cloud features (e.g., intensity) in training datasets could facilitate detecting windows covered by blinds.

Tested façades presented challenging, varying measuring conditions; for similar façade and opening styles, the method is

expected to provide comparable results. Yet, testing sample size implies that caution must be exercised. It is worth noting that *static* objects, which do not contribute to façades elements and are adjacent (e.g., traffic signs, bus shelters), can negatively influence the semantic back-projection results. To further our research, we plan to test the method on a higher number of façades.

## ACKNOWLEDGMENTS

This work was supported by the Bavarian State Ministry for Economic Affairs, Regional Development and Energy within the framework of the IuK Bayern project *MoFa3D - Mobile Erfassung von Fassaden mittels 3D Punktwolken*, Grant No. IUK643/001. Moreover, the work was conducted within the framework of the Leonhard Obermeyer Center at the Technical University of Munich (TUM). We gratefully acknowledge the Geoinformatics team at the TUM for the valuable insights and for providing the CityGML datasets.

## References

- Basaraner, M., Cetinkaya, S., 2017. Performance of shape indices and classification schemes for characterising perceptual shape complexity of building footprints in GIS. *Int. J. Geogr. Inf. Syst.*, 31(10), 1952–1977.
- BayernAtlas, 2022. BayernAtlas, TUM Campus, Theresienstr., Munich, Germany. 3D Buildings data layer. <https://geoportal.bayern.de/bayernatlas/>. Accessed: 2022-01-25.
- Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., Çöltekin, A., 2015. Applications of 3D city models: State of the art review. *ISPRS Int. J. of Geo-Inf.*, 4(4), 2842–2889.
- Fan, H., Wang, Y., Gong, J., 2021. Layout graph model for semantic façade reconstruction using laser point clouds. *Geo-Spat. Inf. Sci.*, 24(3), 403–421.
- Gehring, J., Hebel, M., Arens, M., Stilla, U., 2017. An approach to extract moving objects from MLS data using a volumetric background representation. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-1/W1, 107–114.
- Google Earth, 2022. Google Earth Pro, TUM Campus, Theresienstr., Munich, Germany. 3D Buildings data layer. <http://www.google.com/earth/index.html>. Accessed: 2022-01-25.
- Grilli, E., Farella, E. M., Torresani, A., Remondino, F., 2019. Geometric features analysis for the classification of cultural heritage point clouds. *ISPRS - Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-2/W15, 541–548.
- Grilli, E., Remondino, F., 2019. Classification of 3D digital heritage. *Remote Sens.*, 11(7), 847.
- Grilli, E., Remondino, F., 2020. Machine learning generalisation across different 3D architectural heritage. *ISPRS Int. J. of Geo-Inf.*, 9(6), 379.
- Gröger, G., Kolbe, T. H., Nagel, C., Häfele, K.-H., 2012. OGC City Geography Markup Language CityGML Encoding Standard. Open Geospatial Consortium: Wayland, MA, USA, 2012.
- Hazra, A., 2017. Using the confidence interval confidently. *J. Thorac. Dis.*, 9(10), 4125.
- Hebel, M., Arens, M., Stilla, U., 2013. Change detection in urban areas by object-based analysis and on-the-fly comparison of multi-view ALS data. *ISPRS J. Photogramm. Remote Sens.*, 86, 52–64.
- Hensel, S., Goebbels, S., Kada, M., 2019. Facade reconstruction for textured LoD2 CityGML models based on deep learning and mixed integer linear programming. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-2/W5, 37–44.
- Hornung, A., Wurm, K. M., Bennewitz, M., Stachniss, C., Burgard, W., 2013. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Auton. Robots*, 34(3), 189–206.
- Matrone, F., Grilli, E., Martini, M., Paolanti, M., Pierdicca, R., Remondino, F., 2020. Comparing machine and deep learning methods for large 3D heritage semantic segmentation. *ISPRS Int. J. of Geo-Inf.*, 9(9), 535.
- Nouvel, R., Schulte, C., Eicker, U., Pietruschka, D., Coors, V., 2013. CityGML-based 3D city model for energy diagnostics and urban energy policy support. *IBPSA World*, 2013, 1–7.
- Roschlaub, R., Batscheider, J., 2016. An INSPIRE-conform 3D building model of Bavaria using cadastre information, LiDAR and image matching. *ISPRS - Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLI-B4, 747–754.
- Schwab, B., Kolbe, T. H., 2019. Requirement analysis of 3D road space models for automated driving. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-4/W8, 99–106.
- Special Interest Group 3D, 2020. Modeling Guide for 3D Objects - Part 2: Modeling of Buildings (LoD1, LoD2, LoD3) - SIG3D Quality Wiki EN. <https://en.wiki.quality.sig3d.org/>. Accessed: 2021-10-30.
- Strith, A., Rabe, S.-E., Robaina, O., Grêt-Regamey, A., Celio, E., 2020. An online platform for spatial and iterative modelling with Bayesian networks. *Environ. Model. Softw.*, 127, 104658.
- Suveg, I., Vosselman, G., 2000. 3D reconstruction of building models. *ISPRS - Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XXXIII, 538–545.
- Szeliski, R., 2010. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.
- Tuttas, S., Stilla, U., 2013. Reconstruction of façades in point clouds from multi aspect oblique ALS. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, II-3/W3, 91–96.
- Tuttas, S., Stilla, U., Braun, A., Borrmann, A., 2015. Validation of BIM components by photogrammetric point clouds for construction site monitoring. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, II-3/W4, 231–237.
- Weinmann, M., Jutzi, B., Mallet, C., 2013. Feature relevance assessment for the semantic interpretation of 3D point cloud data. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, II-5/W2, 313–318.
- Willenborg, B., Pültz, M., Kolbe, T. H., 2018. Integration of semantic 3D city models and 3D mesh models for accuracy improvements of solar potential analyses. *ISPRS - Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-4/W10, 223–230.
- Wysocki, O., Hoegner, L., Stilla, U., 2022a. Refinement of semantic 3D building models by reconstructing underpasses from MLS point clouds. *Int. J. Appl. Earth Obs. Geoinf.*, 111, 102841.
- Wysocki, O., Hoegner, L., Stilla, U., 2022b. TUM-FAÇADE: Reviewing and enriching point cloud benchmarks for façade segmentation. *ISPRS - Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLVI-2/W1-2022, 529–536.
- Wysocki, O., Schwab, B., Hoegner, L., Kolbe, T., Stilla, U., 2021a. Plastic surgery for 3D city models: A pipeline for automatic geometry refinement and semantic enrichment. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, V-4, 17–24.
- Wysocki, O., Xu, Y., Stilla, U., 2021b. Unlocking point cloud potential: Fusing MLS point clouds with semantic 3D building models while considering uncertainty. *ISPRS - Int. Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, VIII-4/W2, 45–52.
- Xu, Y., Stilla, U., 2021. Towards building and civil infrastructure reconstruction from point clouds: A review on data and key techniques. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 14, 2857–2885.
- Zhao, H., Jiang, L., Jia, J., Torr, P. H., Koltun, V., 2021. Point transformer. *Proc. IEEE Int. Conf. Comput. Vis.*, 16259–16268.
- Zhu, J., Gehring, J., Huang, R., Borgmann, B., Sun, Z., Hoegner, L., Hebel, M., Xu, Y., Stilla, U., 2020. TUM-MLS-2016: An annotated mobile LiDAR dataset of the TUM City Campus for semantic point cloud interpretation in urban areas. *Remote Sens.*, 12(11), 1875.
- Zolanvari, S. I., Laefer, D. F., Natanzi, A. S., 2018. Three-dimensional building façade segmentation and opening area detection from point clouds. *ISPRS J. Photogramm. Remote Sens.*, 143, 134–149.