Enriching LoD2 Building Models with Façade Openings Using Oblique Imagery

Yitong Xia, Weixiao Gao[*], Jantien Stoter

Faculty of Architecture and the Built Environment, Delft University of Technology, The Netherlands -
xiayitong0630@gmail.com, (w.gao-1,j.e.stoter)@tudelft.nl

Keywords: LoD3 Building Models, 3D Reconstruction, Oblique Aerial Imagery, Semantic Segmentation

Abstract

High-precision 3D urban applications — including emergency response simulation, microclimate analysis, and heritage conservation — demand semantically enriched 3D building representations at Level of Detail 3 (LoD3) with parametric façade components. Current urban digital twins predominantly rely on LoD2 models (as exemplified by the nationwide 3D BAG dataset in the Netherlands) that lack critical architectural features such as windows and doors, constraining their analytical value and their utility for fine-grained applications. This study introduces a novel pipeline to bridge this gap, enabling the enrichment of LoD2 models with accurate opening information using aerial oblique imagery and deep learning. The approach addresses critical challenges in 3D-2D alignment by leveraging perspective projection for comprehensive façade extraction, least-squares registration to rectify systematic offsets, and Mask R-CNN for robust opening detection. Unlike conventional methods, it captures both inward and outward building faces by projecting all 3D façades onto multi-directional images, ensuring complete coverage of visible elements. Geometric scaling integrates detected openings into LoD2 models as watertight, semantically rich components, validated for structural consistency. By overcoming data misalignments and occlusion limitations, this methodology provides a scalable framework for large-scale LoD3 generation, enabling efficient upgrades of existing building models to support detailed spatial analysis in smart city contexts. Our source code and data are available at https://github.com/YitongXia/LOD3-Model.

1. Introduction

The increasing scale and complexity of urbanization intensify the need for sustainable urban planning and smart city development. Smart cities leverage digital technologies to optimize urban systems (e.g., administration, transportation) and enhance livability (Su et al., 2011). Central to this vision are 3D city models, which integrate multi-source geospatial data, including remote sensing images (Singh et al., 2013), point clouds (Peters et al., 2022), and textured meshes (Gao et al., 2021, 2025), to digitally represent urban environments with geometric precision. These models are classified by LoD standards: LoD1 (simplified volumetric blocks) supports city-scale energy simulations and shadow analysis; LoD2 (roof structures and basic façades) enables cadastral management and noise mapping; LoD3 (detailed façades with windows, doors, and balconies) is critical for emergency response planning, heritage preservation, and microclimatic studies (Biljecki et al., 2015). Global initiatives like 3D BAG (Netherlands) (Peters et al., 2022) and Helsinki's digital twin (Airaksinen et al., 2019) exemplify the adoption of LoD1/LoD2 models. However, LoD3's finer granularity remains underexplored, specifically for large areas that require an automated approach, despite its potential for high-precision urban analytics.

Generating LoD3 models faces three key challenges: (1) Data limitations: LiDAR offers geometric accuracy but struggles with texture details (Leberl et al., 2010; Akmalia et al., 2014); oblique imagery contains rich façade textures but suffers from occlusions and resolution variability (Huang et al., 2020; Pantoja-Rosero et al., 2022); BIM provides semantic details but lacks broad coverage (Geiger et al., 2015). (2) Technical complexity: automating façade element extraction (e.g., irregularly distributed windows) requires robust algorithms to handle occlusions, lighting variations, and data misalignments (Zhang et al., 2019; AlHalawani et al., 2013). Existing photogrammetric workflows often rely on manual corrections (Nan et al., 2010) or assume uniform opening patterns (AlHalawani et al., 2013), limiting scalability. (3) Integration gaps: most methods reconstruct LoD3 models independently rather than upgrading existing LoD2 datasets (Gruen et al., 2019), leading to redundant efforts and inconsistencies.

To overcome these challenges, this research contributes three key advancements in LoD3 generation: (1) Misalignment correction for city-scale oblique imagery: We introduce a least-square regression method to resolve systematic offsets between 3D LoD2 models and oblique aerial images, enabling precise façade texture extraction without manual alignment — a critical innovation for scaling LoD3 workflows. (2) Deep learning-based façade parsing with layout regularization: By employing Mask R-CNN for opening detection, followed by a regularization algorithm to optimize irregular opening layouts in 2D space, our method achieves robust façade element extraction even under occlusion/perspective distortions. (3) 2D-to-3D integration via geometric repurposing: Instead of reconstructing full 3D models, we project optimized 2D openings into 3D space using similar-triangle principles and integrate them into LoD2 models as intrusion elements. This BIM-agnostic strategy uniquely leverages existing open datasets to bypass redundant reconstruction, enabling large-scale LoD3 generation.

2. Related work

3D building model reconstruction methodologies are broadly categorized by automation level (fully/semi-automatic), data sources (LiDAR, imagery, topographic data), and approaches

[*] Corresponding author

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-4-W6-2025-225-2025 | © Author(s) 2025. CC BY 4.0 License.

(model-driven vs. data-driven) (Oniga et al., 2022). Model-driven (top-down) methods leverage prior knowledge from predefined libraries to assemble building components, ensuring robustness, but are limited by library diversity. In contrast, data-driven (bottom-up) techniques reconstruct models directly from geometric primitives (e.g., point clouds, planes) without prior assumptions, facing challenges in handling complex structures and ensuring robustness.

2.1 3D Building model reconstruction

LoD2 and LoD3 reconstruction rely on diverse data sources. For LoD2, satellite imagery combined with DSM and OSM data enables cost-effective urban-scale reconstruction but struggles with complex geometries (He et al., 2023; Bullinger et al., 2021). Street View Imagery (SVI) offers façade details for single-view or two-view reconstructions, though accuracy depends on image quality (Pang and Biljecki, 2022). Oblique aerial imagery, capturing façades at angles, enhances detail when fused with terrestrial images (Wu et al., 2018). LiDAR data, including Aerial Laser Scanning (ALS), Terrestrial Laser Scanning (TLS), and Mobile Laser Scanning (MLS), supports automated roof detection via Hough transforms (Overby et al., 2004; Chen et al., 2018) and deep learning-based voxel classification (Pirotti et al., 2019), while photogrammetric point clouds from SfM/MVS enable mesh generation (Nan and Wonka, 2017; Pantoja-Rosero et al., 2022). Transitioning to LoD3 requires integrating façade elements (e.g., windows, doors) into the LoD2 models.

Hybrid methods combine LoD2 models with multi-source data: Huang et al. fused aerial/terrestrial imagery using predefined primitives (Huang et al., 2020), while Pantoja-Rosero et al. enriched LoD2 meshes via semantic segmentation and SIFT-based 3D keypoint triangulation (Pantoja-Rosero et al., 2022). Zhang et al. (2019) augmented LoD2 CityGML models by projecting Mask R-CNN-detected openings from façade textures, demonstrating scalability. Challenges persist in automation, irregular pattern handling, and multi-source registration (Wen et al., 2019). In this study, we address these limitations by automating LoD3 generation through oblique imagery and pre-existing LoD2 models, eliminating manual intervention and leveraging geometric optimization for irregular opening integration.

2.2 Façade parsing

Façade element detection employs pixel-based (Yang et al., 2015) and deep learning methods (Liu et al., 2020). YOLOv3 enables real-time detection using Darknet-53 and FPN (Redmon and Farhadi, 2018), whereas Mask R-CNN improves pixel-level accuracy via RoIAlign (He et al., 2017). Layout regularization addresses aesthetic and structural consistency. Hensel et al. (2019) aligned openings using MILP, while Hu et al. (2020) optimized bounding boxes via BIP clustering. Liu et al. (2020) integrated symmetric regularization into CNNs to penalize irregular shapes, and Jiang et al. (2016) enforced alignment, size, and spacing constraints through energy minimization. These methods balance automation with precision but face limitations in handling highly irregular façades (Pantoja-Rosero et al., 2022). Our approach resolves these challenges by combining Mask R-CNN with a regularization algorithm to optimize irregular layouts in 2D space before 3D projection, ensuring both structural consistency and adaptability to complex façade patterns.

3. Methodology

We developed a pipeline to enrich LoD2 building models with opening details. It begins by pre-processing solid building models modelled as LoD2 models, merging co-planar surfaces, and extracting camera parameters (Sec. 3.1). Then, 3D façade corners are projected to 2D space, with (LSR) rectifying offsets for aligned images (Sec. 3.2). The Mask R-CNN framework (He et al., 2017), trained on a combined dataset, detects and segments openings, validated by IoU and accuracy metrics (Sec. 3.3). After normalization, 2D coordinates are converted to 3D and integrated into the models, adding precise openings.

3.1 Pre-processing

The pre-processing stage focuses on preparing two types of data to support accurate façade extraction. The input data are oblique aerial images and LoD2 building models, and the goal is to make these data suitable for subsequent analysis by adjusting camera parameters and refining co-planar surfaces.

Adjusting camera parameters: The inputs for this step are oblique aerial images and their corresponding camera coordinates. The aim is to define the valid extraction regions for building models to be processed. We use Pix4D S.A. (2025) to derive camera parameters. The process involves feature extraction, image matching, and bundle adjustment to reduce re-projection errors, enabling accurate mapping of 2D image points to 3D world coordinates. Given the common absence of Ground Control Points (GCPs) in oblique datasets, we apply back-projection to calculate the 3D coverage of each image. Only buildings fully within the image are selected for façade extraction, bridging the gap between the image space and 3D city models.

Merge co-planar surfaces: Using the 3D BAG LoD2 model (Peters et al., 2022) as input (which consisted of triangular faces at the time of this research), we aim to extract planar façade surfaces. A region-growing algorithm is employed to merge co-planar faces. Starting with random seed faces, adjacent faces meeting the geometric similarity threshold (based on normal vector consistency) are added iteratively. The result is unified, complete façade surfaces, color-coded for validation (see Fig. 1). By calculating surface normal vectors, we retain only vertical wall surfaces, filtering out horizontal roofs and footprints, thus streamlining the dataset for efficient façade analysis.

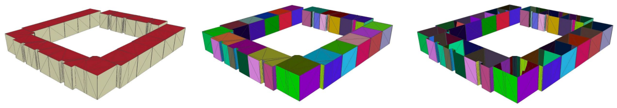


Figure 1. Example of coplanar surface merging. From left to right: 3D BAG LoD2 model, region-growing results (different colors indicate distinct regions), and façade extraction results.

3.2 Façade extraction

Given the challenge of establishing 3D-2D correspondence due to the lack of location information in oblique images, our goal is to develop a reliable method for extracting 2D façade images that accurately match their 3D counterparts (3D façade of the building models). To achieve this, we designed a three-step pipeline involving 3D façade projection, projection optimization, and image rectification (see Fig. 2).

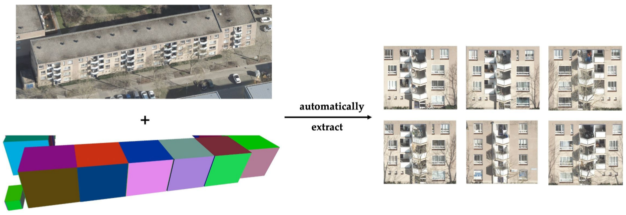


Figure 2. Workflow for automatic façade image extraction.

(1) 3D façade projection. To bridge the gap between 3D building models and 2D oblique images, the aim of 3D façade projection is to create an initial mapping of 3D façades onto the image plane. Using the intrinsic and extrinsic camera parameters, we project the corner points of 3D façades from the 3D building model into 2D space via perspective projection (Equation 1):

$$\begin{pmatrix} x_u \\ x_u \end{pmatrix} = \begin{pmatrix} \frac{fX^{'}}{Z^{'}} \\ \frac{fY^{'}}{Z^{'}} \end{pmatrix} + \begin{pmatrix} c_x \\ c_y \end{pmatrix} \tag{1}$$

where (x_u, y_u) are pixel coordinates, (c_x, c_y) is the principal point, and (X', Y', Z') are 3D points in camera coordinates. This projection forms rectangular constraints in the 2D image (see Fig. 3(a)), providing a preliminary 3D-2D correspondence. By projecting all 3D façades initially, we can filter out non-visible ones during the subsequent opening detection, simplifying the overall process.

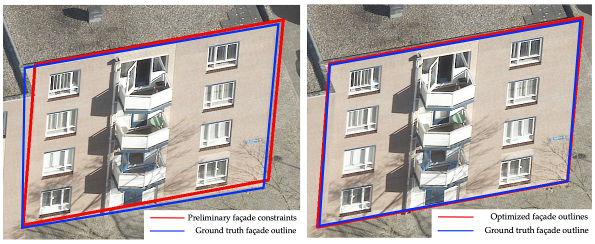
(2) Projection optimization. Due to the absence of GCPs in camera parameter estimation, the initial projection results contain systematic offsets that affect the accuracy of 3D-2D correspondence. The objective here is to correct these offsets to improve the reliability of the extracted façade images. We observe that the offsets exhibit a linear translational relationship for images taken from the same direction. Thus, we employ least-squares registration (LSR) with the linear regression function $y = mx + c$ to minimize the squared residuals between the projected coordinates (x) and the ground truth coordinates (y):

$$m = \frac{\sum(x_i - \bar{x}) \times (y_i - \bar{y})}{\sum(x_i - \bar{x})^2} \tag{2}$$

$$c = \bar{y} - m \times \bar{x} \tag{3}$$

Here, x_i represents the projected coordinate (e.g., from initial 3D-to-2D projection), and y_i represents the corresponding manually annotated ground truth coordinate. The terms $\bar{x}$ and $\bar{y}$ denote the mean values of all projected and ground truth coordinates, respectively. Parameter m quantifies the slope of the linear relationship (how ground truth coordinates change with projected coordinates), while c is the y-intercept, representing the ground truth coordinate when the projected coordinate is zero. Applying LSR to 10 manually annotated façades, we achieve an R-squared value of 0.999, indicating an excellent fit. The derived regression model is then used to optimize all projected points, enhancing the accuracy of 3D-2D correspondence without the complexity of repeated deep learning. This model can be reused for images from the same perspective, balancing efficiency and precision.

(3) Image rectification. After correcting the offsets, the extracted façade images still need to be adjusted to accurately represent the real-world proportions of the façades. The goal of image rectification is to transform the oblique perspective of the extracted images into a frontal view, which is essential for downstream opening detection. Using the optimized rectangular constraints (see Fig. 3(b)), we apply perspective transformation with the homography matrix H (Equations 4-5):


Preliminary façade constraints
Ground truth façade outline
Optimized façade outlines
Ground truth façade outline

(a) Before optimization (b) After optimization

Figure 3. Comparison of projection optimization results.

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} = H \times \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{4}$$

$$(x', y') = (\frac{x}{w}, \frac{y}{w}) \tag{5}$$

where (u, v) are original coordinates, (x, y) are projected coordinates, (x', y') are normalized coordinates in the target image, and w is the normalization factor. This transformation ensures that the rectified images maintain accurate geometric proportions, providing standardized façade representations that are well-aligned with their 3D counterparts.

3.3 Openings detection and optimization

To address irregularities in the detected opening positions and sizes from oblique images and align them with typical architectural layouts, we implement a two-step pipeline: precise detection/segmentation followed by geometric regularization. This ensures that extracted openings not only match their 2D façade representations but also adhere to the regular patterns expected in 3D building models, facilitating accurate integration into LoD2 structures.

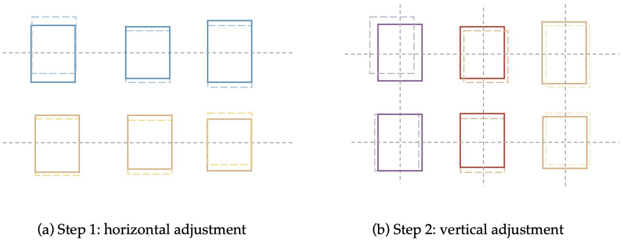
Detection: To accurately identify and segment window and door openings in rectified façade images, we employ the Mask R-CNN framework, which excels in simultaneous object detection and instance segmentation. The ResNet-101 architecture serves as the backbone for feature extraction, leveraging residual connections to mitigate the vanishing gradient problem and enable training of deep networks (He et al., 2016). This choice ensures robust feature representation for complex façade structures. Our training dataset combines 820 images from the public Amsterdam façade dataset (Ams, 2020) and 30 manually annotated images, totaling 850 samples labeled for windows, doors, and sky in MS COCO format (820 for training, 90 for validation). This augmented dataset enhances the model's generalization ability for the diverse opening configurations encountered in 3D BAG building models. After training, the Mask R-CNN model processes each rectified façade image to generate pixel-level segmentation masks and bounding boxes for individual openings, providing precise 2D locations and shapes essential for subsequent layout optimization.

Layout optimization: Architectural openings typically follow regular positional and dimensional patterns for aesthetic and functional consistency, but Mask R-CNN outputs may

https://doi.org/10.5194/isprs-annals-X-4-W6-2025-225-2025

contain variations due to detection noise or perspective effects. The goal of layout optimization is to align detected openings with typical architectural layouts by regularizing their positions and sizes, ensuring geometric coherence and visual plausibility.

- *Position regularization.* To enforce horizontal and vertical alignment of openings, we regularize their centroid coordinates to match common architectural patterns (see Fig. 4). First, centroids (c_{ix}, c_{iy}) of detected openings are sorted by their vertical coordinates c_{iy}. Openings with vertical differences within a predefined threshold (allowing for detection errors) are grouped into the same horizontal row, and their vertical coordinates are replaced with the row's average c_y. A similar process is applied horizontally: centroids are sorted by c_{ix}, and horizontal groups (columns) are formed using the same threshold, with c_x values updated to the column average. This two-step adjustment aligns openings into grid-like structures, where dotted lines represent original positions and solid lines denote regularized positions, enhancing visual order and structural consistency.


(a) Step 1: horizontal adjustment

(b) Step 2: vertical adjustment

Figure 4. Two-step position adjustment: vertical and horizontal alignments.

- *Size Regularization.* To standardize opening dimensions while preserving their centroid positions, we use the density-based DBSCAN clustering algorithm (Ester et al., 1996) to group openings with similar initial sizes (see Fig. 5). DBSCAN's ability to handle arbitrary cluster shapes and automatically determine cluster numbers makes it suitable for unsupervised size classification, with parameters set as eps $= 5$ and min_samples $= 1$ to accommodate single-instance sizes. For each cluster, the average width and height are calculated and applied to all members. Given a centroid (c_{ix}, c_{iy}) and target dimensions w (width), h (height), the four corner coordinates of each opening are adjusted symmetrically around the centroid. Specifically, the coordinates for the upper left, lower left, lower right, and upper right corners are: (c_{ix}-$\frac{w}{2}$, c_{iy}-$\frac{h}{2}$), (c_{ix}-$\frac{w}{2}$, c_{iy}+$\frac{h}{2}$), (c_{ix}+$\frac{h}{2}$, c_{iy}+$\frac{h}{2}$), and (c_{ix}+$\frac{h}{2}$, c_{iy}-$\frac{h}{2}$). This ensures uniform sizes within clusters while maintaining geometric consistency with the underlying 3D façade structure, improving both the accuracy and visual appeal of the optimized openings.

3.4 Integration with 3D LoD2 building models to obtain LoD3 models

To ensure geometric consistency and structural integrity between 2D openings and 3D building models, we introduce a two-step methodology to convert 2D opening coordinates to 3D space and integrate them into LoD2 models, addressing the challenge of mapping planar features to volumetric structures for seamless incorporation of detailed opening information.

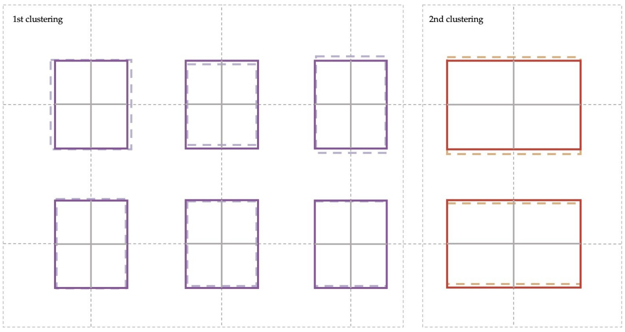


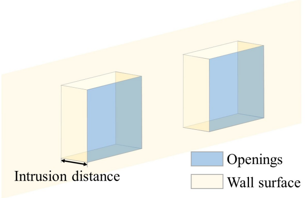
Figure 5. Opening size clustering and regularization.

2D to 3D conversion: To achieve precise alignment between extracted 2D openings and 3D façades while avoiding systematic errors inherent in traditional photogrammetry-based back projection methods, we developed a novel approach rooted in the similar triangle principle. This method ensures the converted 3D openings remain coplanar with façades, crucial for high quality façade and opening extraction. The process unfolds in two key steps as illustrated in Fig. 6.

First, leveraging the guaranteed same aspect ratio between rectified 2D images and 3D façades (due to prior rectification using façade length and width), we calculate the 3D space offsets Δx_{3D} and Δy_{3D} from 2D pixel offsets Δx_{img} and Δy_{img}. Using the proportionality $\frac{\Delta y_{3D}}{\Delta y_{img}} = \frac{H_{3D}}{H_{img}}$, we obtain $\Delta y_{3D} = \frac{H_{3D}}{H_{img}} \times \Delta y_{img}$, then determine the z-value $z_i = z_0 - \Delta y_{3D}$, where H_{3D} and H_{img} are the heights of the 3D façade and 2D image, and z_0 is the initial z-coordinate. Similarly, Δx_{3D} is derived via $\frac{\Delta x_{3D}}{\Delta x_{img}} = \frac{H_{3D}}{H_{img}}$, giving $\Delta x_{3D} = \frac{H_{3D}}{H_{img}} \times \Delta x_{img}$.

Next, for calculating (x_i, y_i), since the façade's footprint is not parallel to XOY - plane axes, we project onto the XOY plane. Using the known 3D distance $F_0 F_1$ (with length W_{3D}) and pixel distance $F_0^{'} F_1^{'}$, we apply scaling relationships. From $\frac{x_i}{x_1 - x_0} = \frac{W_{3D}}{\Delta x_{3D}}$, we get $x_i = \frac{W_{3D}}{\Delta x_{3D}} \times (x_1 - x_0)$, and from $\frac{y_i}{y_1 - y_0} = \frac{W_{3D}}{\Delta x_{3D}}$, we obtain $y_i = \frac{W_{3D}}{\Delta x_{3D}} \times (y_1 - y_0)$. After acquiring all four corner coordinates of an opening, we reevaluate the spatial relationship between the 3D façade and opening to ensure coplanarity, thus maintaining the correct relative positional relationship critical for subsequent integration.

Integration: To maintain the watertightness of the final 3D model, we extrude the converted 3D openings inward from the façade at a uniform depth, creating seamless connections between the original façade and the new opening structure. This process leverages the counterclockwise vertex ordering of 3D building models (with outward-facing normals) to define the spatial orientation of the façade's inner and outer surfaces. As depicted in Fig. 7, the integration workflow involves: 1) generating a new opening plane parallel to the façade at the specified depth, 2) calculating 3D coordinates for both the original and intruded openings, and 3) constructing connecting walls by preserving the counterclockwise vertex sequence. This approach ensures topological consistency and structural coherence, allowing the detailed opening features to be seamlessly integrated into the LoD2 model while meeting the geometric requirements for downstream applications.



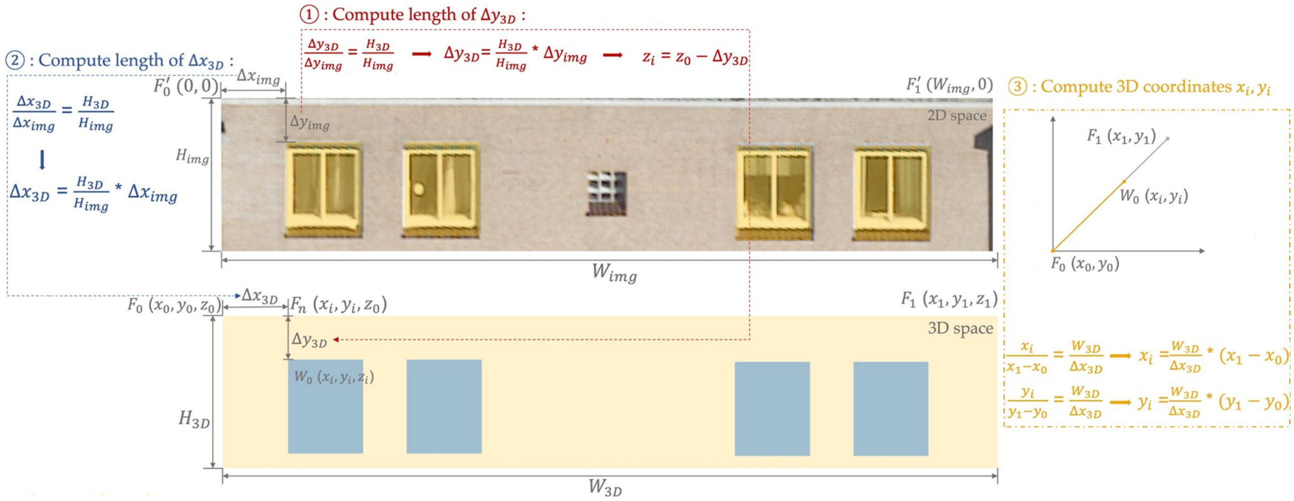

① : Compute length of Δy_{3D} :
$\frac{\Delta y_{3D}}{\Delta y_{img}} = \frac{H_{3D}}{H_{img}} \longrightarrow \Delta y_{3D} = \frac{H_{3D}}{H_{img}} * \Delta y_{img} \longrightarrow z_i = z_0 - \Delta y_{3D}$
② : Compute length of Δx_{3D} :
$\frac{\Delta x_{3D}}{\Delta x_{img}} = \frac{H_{3D}}{H_{img}}$
$\Delta x_{3D} = \frac{H_{3D}}{H_{img}} * \Delta x_{img}$
F_0' $(0,0)$
Δx_{img}
Δy_{img}
H_{img}
W_{img}
F_1' $(W_{img}, 0)$
2D space
③ : Compute 3D coordinates x_i, y_i
F_1 (x_1, y_1)
W_0 (x_i, y_i)
F_0 (x_0, y_0)
F_0 (x_0, y_0, z_0)
Δx_{3D}
F_n (x_i, y_i, z_0)
Δy_{3D}
W_0 (x_i, y_i, z_i)
F_1 (x_1, y_1, z_1)
3D space
H_{3D}
W_{3D}
$\frac{x_i}{x_1 - x_0} = \frac{W_{3D}}{\Delta x_{3D}} \longrightarrow x_i = \frac{W_{3D}}{\Delta x_{3D}} * (x_1 - x_0)$
$\frac{y_i}{y_1 - y_0} = \frac{W_{3D}}{\Delta x_{3D}} \longrightarrow y_i = \frac{W_{3D}}{\Delta x_{3D}} * (y_1 - y_0)$

Figure 6. 2D-to-3D coordinate conversion.

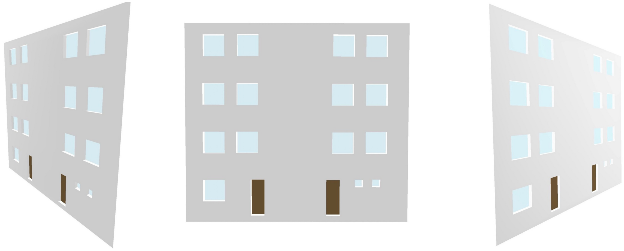


Figure 7. Examples of 3D façade opening integration.

4. Experiments

4.1 Implementation details

Dataset: Our study area is in the northern part of Almere Centrum, a small community with 18 buildings (see Fig. 8). Two primary datasets are employed: the automatically reconstructed nationwide dataset 3D BAG LoD2 building models Peters et al. (2022) and oblique aerial images from the municipality of Almere, captured by CityMapper-2 sensors across four perspectives (forward, back, left, right-looking). After filtering 15 images covering the study area, Pix4D was used to estimate camera parameters (intrinsic matrix, rotation/translation vectors), which are essential for subsequent perspective projection.

Figure 8. Our study area is in the northern part of Almere centrum.

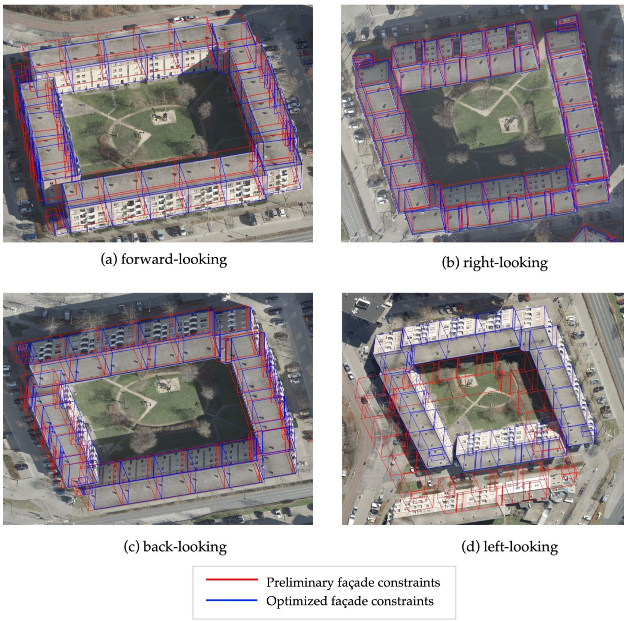
Tools: Key libraries include CGAL (Oesau et al., 2023) for 3D surface region growing, Detectron2 (Wu et al., 2019) (built on PyTorch (Paszke et al., 2019)) for Mask R-CNN implementation, OpenCV (Bradski, 2000) for image processing, and Scikit-learn (Pedregosa et al., 2011) for DBSCAN clustering and least-squares regression. Software tools comprise COCO-annotator (Brooks, 2019) for dataset labeling, Pix4D for photogrammetric parameter extraction, Azul (Arroyo Ohori, 2020) for 3D model visualization, and Val3dity (Ledoux, 2018) for validating the resulting LoD3 models.

Parameter settings: Key parameters were optimized through empirical experiments to balance accuracy and efficiency. For the region-growing algorithm resolving co-planar surfaces, the optimal parameters after iterative testing are set as a maximum vertex-to-plane distance of 10, a maximum normal angle difference of 10 degrees, and a minimum region size of 2 faces, ensuring effective surface merging without misclassification. For Mask R-CNN, extensive hyperparameter tuning was conducted using the Amsterdam façade dataset. After evaluating backbone networks (ResNet-50 vs. ResNet-101) and iteration counts, the best performance was achieved with ResNet-101 (101-layer depth), a learning rate of 0.00025, and 5,000 training iterations. This configuration outperformed ResNet-50 and Faster R-CNN in segmenting windows and doors, as validated by average precision (AP) metrics. In the DBSCAN clustering for opening size regularization, parameters were optimized to handle detection errors and size variations. An *eps* value of 50 was selected to tolerate minor size discrepancies while distinguishing distinct opening groups, with *min_samples* set to 1 to accommodate unique-sized openings commonly found in architectural façades.

4.2 Results

Extracted façades: The projection, registration, and rectification pipeline effectively bridges 3D building models and 2D oblique images to extract accurate façade representations. Using camera parameters from Pix4D, initial perspective projection of 3D BAG façades onto four-oriented images revealed systematic offsets, particularly in left-looking orientations. These were resolved via least-squares registration, as shown in Fig. 9, where optimized projections (blue lines) align tightly with true façade boundaries, eliminating initial misalignments (red lines). Rectification based on 3D dimensions ensured correct aspect ratios, enabling extraction of all visible façades from corresponding image orientations. Statistical analysis confirmed 100% completeness in capturing visible façades across all four directions, leveraging oblique imagery's unique ability to access both exterior and interior building faces—an advantage over traditional street-view methods.

Detected openings: Openings detection using the optimized Mask R-CNN (ResNet-101 backbone, 5,000 iterations) yielded


(a) forward-looking

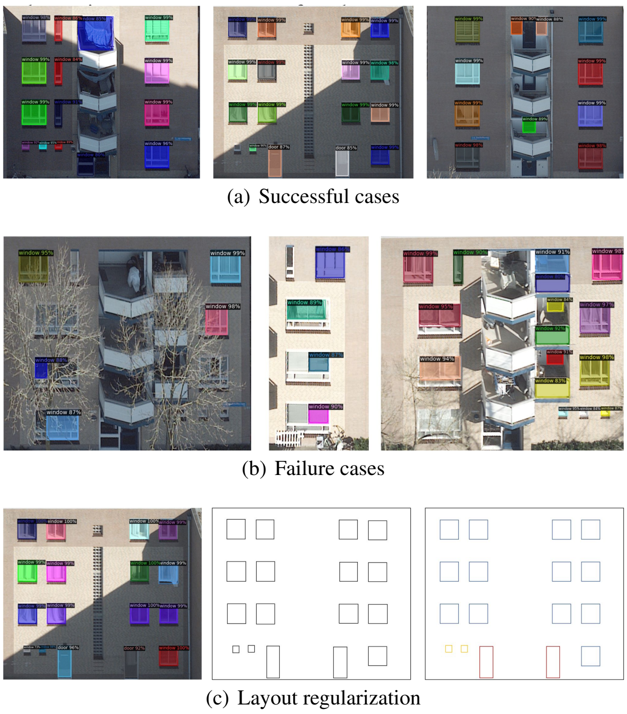
(b) right-looking

(c) back-looking

(d) left-looking

Figure 9. Least-squares projection optimization results for an entire building.

robust results, achieving 75.49% average precision (AP) for openings, including 67.93% for windows and 61.71% for doors. Successful detections (see Fig. 10(a)) clearly identified windows and doors, while failure cases (see Fig. 10(b)) were mostly due to occlusions or complex window structures. Subsequent layout optimization via DBSCAN clustered openings by size, regularizing their positions and dimensions to align with architectural norms. Fig. 10(c) demonstrates improved geometric consistency, with color-coded clusters representing homogeneous size groups, enhancing the plausibility of detec-


(a) Successful cases

(b) Failure cases

(c) Layout regularization

Figure 10. Mask R-CNN façade opening detection results.

Reconstructed LoD3 buidling models: The final LoD3 model reconstruction successfully integrated detailed openings into 3D BAG building models, validated via val3dity (Ledoux, 2018) and visualized in Azul (Arroyo Ohori, 2020) (see Fig. 11). The pipeline upgraded 18 Almere building blocks from LoD2 to LoD3, maintaining watertightness and semantic information (e.g., WallSurface) by recessing openings inward and ensuring connected polygons between windows and walls. Unlike traditional methods limited to outward-facing façades, this approach captures both interior and exterior features without complete model rebuilds. Testing on a larger dataset (see Fig. 11) confirmed robustness, with results adhering to LoD3 standards, highlighting the approach's scalability for large-scale 3D city model enrichment with detailed, se-

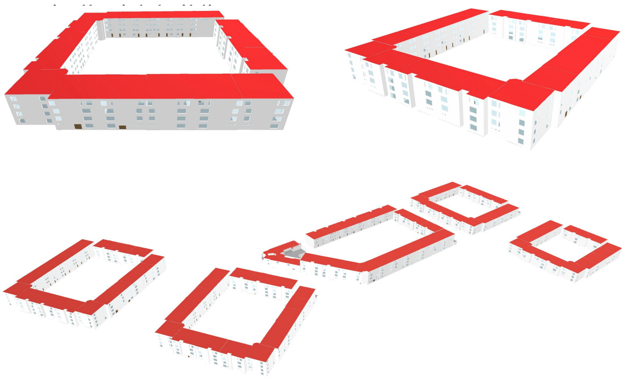


Figure 11. LoD3 building model reconstruction results.

5. Conclusion

We introduce a novel pipeline to upgrade LoD2 building models to semantically enriched LoD3 models, leveraging oblique aerial imagery and deep learning. The approach innovatively bridges 3D-2D spaces through perspective projection and least-squares registration, enabling comprehensive extraction of both inward and outward building façades—an advancement over traditional methods limited to external views. By projecting all 3D façades onto multi-directional images and applying data registration, the pipeline ensures complete capture of visible façades, while Mask R-CNN-based detection accurately identifies openings. A similarity-scaling method integrates 2D detections into 3D models, generating watertight LoD3 structures validated for semantic consistency. Key contributions include a robust framework for large-scale façade extraction, scalable registration across data sources, and the first systematic method to enrich LoD2 models with detailed opening information from multi-view aerial imagery, paving the way for nationwide 3D city model upgrades.

Limitations: Despite these advancements, the pipeline has limitations: it does not handle complex occlusions from trees/balconies in oblique images; it relies on region-specific training data for Mask R-CNN; and it requires manual intervention in data registration. Detection optimization does not account for potential missed openings, and image quality remains a critical factor affecting accuracy.

Future work: Future work will focus on integrating oblique and street-view imagery to mitigate occlusions, incorporating additional LoD3 elements (e.g., balconies, dormers) into a unified pipeline, developing algorithms to pre-select camera-visible façades for efficiency, and automating the registration process to eliminate manual steps. These enhancements aim to improve model completeness, scalability, and automation, enabling more detailed and realistic 3D city model reconstruction for urban planning and analysis.

Acknowledgements

We extend our gratitude to AeroVision for their high-quality aerial imagery and to the Municipality of Almere for granting permission to use their urban dataset images.

References

Airaksinen, E., Bergström, M., Heinonen, H., Kaisla, K., Lahti, K., Suomisto, J., 2019. The Kalasatama digital twins project—The final report of the KIRA-digi pilot project. Technical report, City of Helsinki.

Akmalia, R., Setan, H., Majid, Z., Suwardhi, D., Chong, A., 2014. TLS for generating multi-LOD of 3D building model. *IOP Conference Series: Earth and Environmental Science*, 18(1), 012064.

AlHalawani, S., Yang, Y.-L., Liu, H., Mitra, N. J., 2013. Interactive Facades Analysis and Synthesis of Semi-Regular Facades. *Computer Graphics Forum*, 32(2pt2), 215–224.

Amsterdam, 2020. Amsterdam facade dataset. `https://drive.google.com/file/d/1nkZXSTCM019HGX1QtG2z3sZ3jLoXVL3f/view`. Accessed on July 1, 2025.

Arroyo Ohori, G., 2020. Azul: A fast and efficient 3D city model viewer for macOS. *Transactions in GIS*, 24(5), 1165–1184.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., Çöltekin, A., 2015. Applications of 3D City Models: State of the Art Review. *ISPRS International Journal of Geo-Information*, 4(4), 2842–2889.

Bradski, G., 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Brooks, J., 2019. COCO Annotator. `https://github.com/jsbroks/coco-annotator/`. Accessed on July 1, 2025.

Bullinger, S., Bodensteiner, C., Arens, M., 2021. 3D surface reconstruction from multi-date satellite images. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2021, 313–320.

Chen, K., Lu, W., Xue, F., Tang, P., Li, L. H., 2018. Automatic building information model reconstruction in high-density urban areas: Augmenting multi-source data with architectural knowledge. *Automation in Construction*, 93, 22-34.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, AAAI Press, 226–231.

Gao, W., Nan, L., Boom, B., Ledoux, H., 2021. SUM: A Benchmark Dataset of Semantic Urban Meshes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 179, 108-120.

Gao, W., Nan, L., Ledoux, H., 2025. SUM Parts: Benchmarking part-level semantic segmentation of urban meshes. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE/CVF.

Geiger, A., Benner, J., Haefele, K. H., 2015. *Generalization of 3D IFC Building Models*. Springer International Publishing, Cham, 19–35.

Gruen, A., Schubiger, S., Qin, R., Schrotter, G., Xiong, B., Li, J., Ling, X., Xiao, C., Yao, S., Nuesch, F., 2019. Semantically enriched high-resolution LoD 3 building model generation. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 11–18.

He, K., Gkioxari, G., Dollar, P., Girshick, R., 2017. Mask r-cnn. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

He, Y., Liao, W., Hong, H., Huang, X., 2023. High-Precision Single Building Model Reconstruction Based on the Registration between OSM and DSM from Satellite Stereos. *Remote Sensing*, 15(5).

Hensel, S., Goebbels, S., Kada, M., 2019. Façade reconstruction for textured LOD2 CityGML models based on deep learning and mixed integer linear programming. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-2/W5, 37–44.

Hu, H., Wang, L., Zhang, M., Ding, Y., Zhu, Q., 2020. Fast and regularized reconstruction of building façades from street-view images using binary integer programming. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-2-2020, 365–371. https://isprs-annals.copernicus.org/articles/V-2-2020/365/2020/.

Huang, H., Michelini, M., Schmitz, M., Roth, L., Mayer, H., 2020. LOD3 building reconstruction from multi-source images. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 43.

Jiang, H., Nan, L., Yan, D.-M., Dong, W., Zhang, X., Wonka, P., 2016. Automatic Constraint Detection for 2D Layout Regularization. *IEEE Transactions on Visualization and Computer Graphics*, 22(8), 1933-1944.

Leberl, F., Irschara, A., Pock, T., Meixner, P., Gruber, M., Scholz, S., Wiechert, A., 2010. Point clouds: Lidar versus 3D Vision. *Photogrammetric Engineering & Remote Sensing*, 76(10), 1123–1134.

Ledoux, H., 2018. Val3dity: validation of 3D GIS primitives according to the international standards. *Open Geospatial Data, Software and Standards*, 3(1), 1–12.

Liu, H., Xu, Y., Zhang, J., Zhu, J., Li, Y., Hoi, S. C. H., 2020. DeepFacade: A Deep Learning Approach to Facade Parsing With Symmetric Loss. *IEEE Transactions on Multimedia*, 22(12), 3153-3165.

Nan, L., Sharf, A., Zhang, H., Cohen-Or, D., Chen, B., 2010. SmartBoxes for Interactive Urban Reconstruction. *ACM SIGGRAPH 2010 Papers*.

Nan, L., Wonka, P., 2017. PolyFit: Polygonal surface reconstruction from point clouds. *Proceedings of the IEEE International Conference on Computer Vision*, 2353–2361.

Oesau, S., Verdie, Y., Jamin, C., Alliez, P., Lafarge, F., Giraudot, S., Hoang, T., Anisimov, D., 2023. Shape detection. *CGAL User and Reference Manual*, 5.5.2 edn, CGAL Editorial Board.

Oniga, V.-E., Breaban, A.-I., Pfeifer, N., Diac, M., 2022. 3D Modeling of Urban Area Based on Oblique UAS Images – An End-to-End Pipeline. *Remote Sensing*, 14(2).

Overby, J., Bodum, L., Kjems, E., Iisoe, P., 2004. Automatic 3D building reconstruction from airborne laser scanning and cadastral data using Hough transform. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34(01).

Pang, H. E., Biljecki, F., 2022. 3D building reconstruction from single street view images using deep learning. *International Journal of Applied Earth Observation and Geoinformation*, 112, 102859.

Pantoja-Rosero, B., Achanta, R., Kozinski, M., Fua, P., Perez-Cruz, F., Beyer, K., 2022. Generating LOD3 building models from structure-from-motion and semantic segmentation. *Automation in Construction*, 141, 104430.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems 32*, Curran Associates, Inc., 8024–8035.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

Peters, R., Dukai, B., Vitalis, S., van Liempt, J., Stoter, J., 2022. Automated 3D reconstruction of LoD2 and LoD1 models for all 10 million buildings of the Netherlands. *Photogrammetric Engineering and Remote Sensing*, 88(3), 165–170.

Pirotti, F., Zanchetta, C., Previtali, M., Della Torre, S., 2019. Detection of building roofs and facades from aerial laser scanning data using deep learning. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(2), 975–980.

Pix4D S.A., 2025. Pix4Dmapper Photogrammetry Software. `https://www.pix4d.com/product/pix4dmapper-photogrammetry-software`. Accessed: April 21, 2025.

Redmon, J., Farhadi, A., 2018. Yolov3: An incremental improvement.

Singh, S. P., Jain, K., Mandla, V. R., 2013. Virtual 3D city modeling: techniques and applications. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-2/W2, 73–91.

Su, K., Li, J., Fu, H., 2011. Smart city and the applications. *2011 International Conference on Electronics, Communications and Control (ICECC)*, 1028–1031.

Wen, X., Xie, H., Liu, H., Yan, L., 2019. Accurate Reconstruction of the LoD3 Building Model by Integrating Multi-Source Point Clouds and Oblique Remote Sensing Imagery. *ISPRS International Journal of Geo-Information*, 8(3). https://www.mdpi.com/2220-9964/8/3/135.

Wu, B., Xie, L., Hu, H., Zhu, Q., Yau, E., 2018. Integration of aerial oblique imagery and terrestrial imagery for optimized 3D modeling in urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, 139, 119-132.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., Girshick, R., 2019. Detectron2. `https://github.com/facebookresearch/detectron2`.

Yang, X., Qin, X., Wang, J., Wang, J., Ye, X., Qin, Q., 2015. Building Façade Recognition Using Oblique Aerial Images. *Remote Sensing*, 7(8), 10562–10588. https://www.mdpi.com/2072-4292/7/8/10562.

Zhang, X., Lippoldt, F., Chen, K., Johan, H., Erdt, M., Zhang, X., Lippoldt, F., Chen, K., Johan, H., Erdt, M., 2019. A Data-driven Approach for Adding Facade Details to Textured LoD2 CityGML Models. 294–301.