The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume L-4/W2-2026
21st International 3D GeoInfo Conference 2026, 28–29 September 2026, Sofia, Bulgaria

Towards AI-Generated 3D City Models: Bridging Research and Practice

Lilli Kaufhold[1], Martin Kada[1]

[1] Institute of Geodesy and Geoinformation Science, Technische Universität Berlin, Germany
(lilli.kaufhold, martin.kada)@tu-berlin.de

Keywords: Building Reconstruction, 3D, Deep Learning, Validation, Airborne Laser Scanning (ALS), Face identification.

Abstract

Deep learning methods for 3D building reconstruction from airborne laser scanning point clouds report increasingly strong geometric accuracy on benchmark datasets. However, benchmark performance does not guarantee that the resulting models are usable for city modelling, energy simulation, or flood and noise analysis, which generally require reconstructed buildings to be geometrically valid solids. We comparatively evaluate two architecturally distinct learning-based methods, BWFormer (wireframe prediction) and Point2Building (autoregressive mesh generation), on their own test datasets and on the ISPRS Vaihingen benchmark as a common independent site. We contribute a wireframe-to-solid post-processing pipeline with hard planarity constraints, assess validity with val3dity against the ISO 19107 standard at the polygon and solid level, apply the identical evaluation to the ground truth itself, and measure computational cost. Our results demonstrate that benchmark accuracy and geometric validity are effectively decoupled. BWFormer generates wireframe representations that, by design, do not constitute volumetric solids. Our post-processing pipeline produces geometrically valid solid models in 69.8% of cases. In contrast, Point2Building directly predicts mesh representations that do not require post-processing but produces valid solids in only 0.6% of cases. These shortcomings can be attributed in part to the training data, since none of the evaluated datasets provides valid solids as supervisory signals, implying that geometric validity cannot be explicitly learned from them. We therefore conclude that, for learning-based reconstruction methods to yield standard-compliant city models, data representations, training objectives, and evaluation benchmarks must be redesigned to treat validity as a primary optimisation objective.

1. Introduction

Automated reconstruction of 3D building models from airborne laser scanning (ALS) point clouds is a long-standing challenge in geospatial data processing. While rule-based and semi-automatic methods have been used in production for decades, recent research on deep learning approaches shows strong potential for fully automated reconstruction with improved detail and accuracy. However, reported benchmark performance alone does not guarantee practical usability. For downstream applications and standard-compliant city models, reconstructed buildings must satisfy geometric and topological validity criteria: solids must be watertight, surfaces 2-manifold, and faces planar, consistently oriented, and free of self-intersections. Whether current deep learning methods meet these requirements has received little attention. In this study, we provide a comparative evaluation of two conceptually distinct approaches for 3D building reconstruction from ALS data, representing two prominent research directions in the field: roof wireframe approaches, exemplified by BWFormer (Liu et al., 2025), and autoregressive mesh generation approaches, exemplified by Point2Building (Liu et al., 2024). We discuss both concepts from a theoretical perspective and experimentally evaluate the two example approaches on their respective test datasets, as well as on the ISPRS Vaihingen benchmark (Rottensteiner et al., 2012) as a common independent test site. We assess model validity using val3dity (Ledoux, 2018), analyse computational cost and scalability, and examine the post-processing effort required to obtain closed polyhedral solids. Our findings reveal a substantial gap between reported benchmark metrics and the practical usability of the outputs, highlighting open challenges for learning-based building reconstruction.

2. Related Work

We organise learning-based methods by the output representation of the network, since this determines which geometric and topological guarantees hold by construction and which must be recovered afterwards. We discuss wireframe prediction, direct mesh generation, and neural-guided primitive assembly in turn, followed by non-learning production systems that deliver valid solids, and close with an overview of benchmarks and evaluation practice.

2.1 Wireframe reconstruction

Wireframe approaches represent a building, or more commonly only its roof, as a graph of corner vertices and connecting edges. Point2Roof (Li et al., 2022) established the dominant two-stage paradigm: candidate corners are detected and refined from point features, after which edges are classified from the set of corner pairs. Subsequent work has largely retained this corner-then-edge decomposition whilst improving the feature extraction and classification stages. In order to improve robustness on noisy, real-world point clouds, hypergraph attention and transformer-enhanced edge classification have been proposed (Gong et al., 2025). PBWR (Huang et al., 2023) instead regresses edge parameters directly, without an intermediate corner detection step, followed by an edge non-maximum suppression. BWFormer (Liu et al., 2025) exploits the 2.5D nature of ALS data by detecting corners in a rasterised height map, lifting them to 3D with a Transformer-based module, and predicting edges with an edge attention mechanism. LOD2-Former (Abdelhedi et al., 2026) extends this corner-heatmap strategy with a second modality, fusing aerial imagery with the point cloud to compensate for sparse or incomplete LiDAR coverage. Common to these methods is that edges are classified independently, in a single forward pass. Since no structural constraints couple the decisions, locally ambiguous evidence can result in both of two competing edges being accepted, or neither, yielding dangling edges and non-closing cycles. A more recent line of work generates wireframes autoregressively instead. Ma et al. generate semantically enriched house wireframes as wire sequences (Ma et al., 2025), and BuildingGPT reformulates wireframe reconstruction from point clouds as next-token prediction (Liu et al., 2026). Regardless of how it is predicted, however, a wireframe is not a boundary representation: it carries no surface information, so polygonal faces and closed solids must be recovered in a separate, non-trivial post-processing step.

2.2 Direct mesh generation

A second direction predicts polygonal meshes directly. Point2Building (Liu et al., 2024) transfers autoregressive mesh generation in the style of PolyGen (Nash et al., 2020) from object-level shape generation to building reconstruction: conditioned on the ALS point cloud, a Transformer first generates a sequence of quantised vertex coordinates and subsequently a sequence of faces as tuples of vertex indices. The output is a polygonal mesh, so faces are available by construction, and no topology recovery is needed in principle. Sequential generation avoids the independent-edge problem of one-shot wireframe classification, but it introduces failure modes of its own: nothing in the decoding process enforces that the generated face set is watertight, 2-manifold, or free of self-intersections, and errors early in the sequence propagate to all subsequent tokens.

2.3 Neural-guided primitive assembly

A third family of methods is not end-to-end, but combines learning with geometric processing. Chen et al. partition space into a cell complex from detected planar primitives, learn a deep implicit field for occupancy, and extract the building surface by optimisation of the Markov random field (Chen et al., 2022). In PolyGNN (Chen et al., 2024), the authors cast the same selection problem as graph node classification over the polyhedral decomposition. Because the output is the boundary of a union of polyhedral cells, watertightness is guaranteed by construction. These methods depend on upstream plane detection and polyhedral decomposition, and errors in these stages cannot be corrected by the network. Points2Model (Akwensi et al., 2025) inverts the neural-guided assembly: learned point completion and up-sampling enhance the input, after which corners and edges are assembled by hypothesis and selection under geometric constraints.

2.4 Valid solids in production systems

That valid solids at scale are achievable, albeit without deep learning, is demonstrated by geometric optimisation and production pipelines. City3D (Huang et al., 2022) infers the vertical walls missing from ALS data and selects the final model from hypothesised faces via integer programming, building on a hypothesis-and-selection framework. Zang et al. partition compound roofs into regular patches and assemble models from a primitive library, explicitly targeting topological consistency (Zang et al., 2024). Most prominently, the 3DBAG pipeline (Peters et al., 2022) reconstructs LoD1 and LoD2 models for all ten million buildings of the Netherlands from cadastral footprints and the national ALS point cloud, and its outputs are systematically assessed with relative quality metrics including geometric validity (Dukai et al., 2021). These systems set the practical standard against which learning-based outputs must be measured: whatever accuracy a network achieves, downstream use requires the same validity that production pipelines already deliver.

2.5 Benchmarks

Benchmark datasets have shaped the field considerably. The ISPRS Vaihingen benchmark (Rottensteiner et al., 2014) established a common test site and reference-based evaluation for building detection and 3D reconstruction. Building3D (Wang et al., 2023) scaled this up to more than 160,000 buildings across 16 Estonian cities with paired point clouds, meshes, and wireframes, and has become a standard training and evaluation corpus for wireframe methods. More recent efforts address stylistic diversity and standardised metrics (Huang et al., 2025), LoD2 reconstruction from national lidar programmes and footprints (Geniet et al., 2025), roof plane annotation quality (Schuegraf et al., 2023), and the incompleteness of building point clouds (Gao et al., 2024). Comparative studies of data- and model-driven pipelines further note the lack of standardised evaluation metrics (Jarzabek-Rychard et al., 2025). Across all of these, evaluation focuses on geometric accuracy and completeness, such as corner and edge distances or RMSE against reference surfaces. Whether the reconstructed models constitute valid solids in the sense of ISO 19107, as verifiable with val3dity (Ledoux, 2018) and as required by CityGML-based city models, is absent from common evaluation protocols. Exceptions exist on the production side (Dukai et al., 2021). For learning-based reconstruction, this question is systematically addressed in the present study.

3. Methods

We evaluated one representative of each of the two prominent end-to-end reconstruction approaches identified in Section 2: BWFormer (Liu et al., 2025) for wireframe prediction and Point2Building (Liu et al., 2024) for autoregressive mesh generation. Each is evaluated on the test split of its own training data and on a common independent site. The three datasets differ in how they represent their reference geometry, which impacts both what each method can learn and how its predictions can be validated.

3.1 Datasets and their reference formats

Building3D (Wang et al., 2023), on whose Tallinn test split (3,472 buildings) we evaluate BWFormer, annotates roofs as wireframes that do not include vertical edges (Figure 1, top). The point clouds are heavily cleaned, and the resulting gaps are carried into the annotations, so the annotated outlines do not necessarily correspond to the true building footprints. Furthermore, where a dormer structure intersects its host roof face, the intersection is encoded as an inner ring of edges on the host face, so faces must be recovered together with their interior rings.

Zurich (Liu et al., 2024), on whose test split (2,691 buildings) we evaluate Point2Building, pairs ALS point clouds from the Swiss Federal Office of Topography with polygonal meshes from the city model of Zurich (Figure 1, middle), which provide explicit surface information.

ISPRS Vaihingen (Rottensteiner et al., 2012) provides manually surveyed reference roof polygons for 110 buildings, given

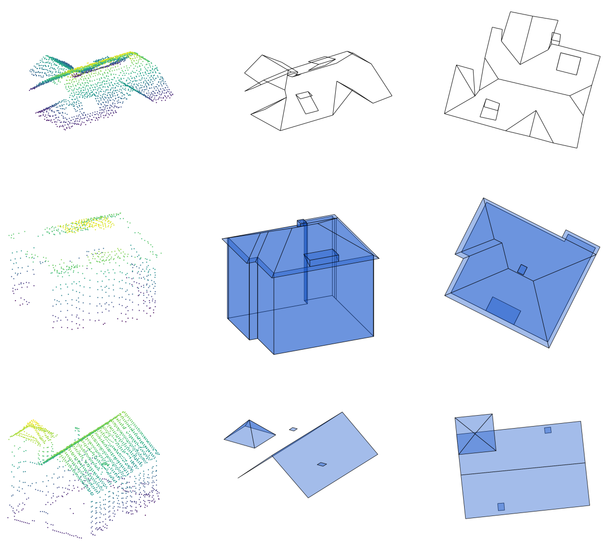

Figure 1. Sample buildings from the three datasets. Each row shows the point cloud, the reference geometry, and its top-down projection. Top: Building3D (Tallinn), wireframe annotation. Middle: Zurich, city model mesh. Bottom: Vaihingen, surveyed roof polygons.

as indexed face sets (Figure 1, bottom). Its ALS point cloud differs in acquisition characteristics from the training data of either method, and thus allows testing generalisation performance for both methods.

3.2 Post-processing towards solids

Wireframes carry no surface information, so validating them as solids requires recovering surfaces first. Our results for BWFormer therefore evaluate the method together with this post-processing, whose failure rate and runtime we report alongside those of the network.

Face extraction. Face loops are only well defined on a graph free of degree-1 vertices and bridges, so the wireframe is first cleaned. We repair degree-1 vertices by greedy nearest-neighbour bridging within a radius and prune remaining dangling chains. The wireframe is split into biconnected components, each treated as an independent closed surface with an expected number of face loops given by Euler's formula. Within each component we extract face loops with an adaption of the crystal growing method (Varley and Company, 2010), which can handle slightly non-planar graphs: starting from a seed on the boundary, faces are grown while keeping track of inside and outside, with a fallback to a depth-first search that favours edges with minimum angle to the previous one (Fan and Chang, 1991). Loops with downward normals, and loops that are both poorly planar and small relative to their 2D hull, are identified as outside face or inner rings and discarded.

Planarity optimisation. The extracted faces exhibit non-planarity inherited from independently predicted corner heights. We solve a constrained Gauss–Newton problem per building, jointly optimising a height per vertex, a plane per face and a horizontal displacement per corner. Planarity is enforced as a hard constraint, requiring that every vertex lies exactly within the plane of each face it belongs to. The objective minimises the vertical deviation from the predicted heights and the horizontal displacements, with ridge regularisation on the plane gradients. Vertices with identical x,y position are assigned a common displacement, thus preserving the verticality of step edges.

Solidification. Each face is triangulated using Triangle (Shewchuk, 1996) and extruded to a common ground plane below the building. The per-face volumes are combined by Boolean union via mesh arrangements (Zhou et al., 2016), as implemented in libigl (Jacobson et al., 2018) on the exact CGAL kernel (The CGAL Project, 2019), bundled by PyMesh (Zhou, 2018). The original face ID of each triangle is tracked through the union. The result is a watertight, triangulated solid per building.

Face merging. Since triangles are trivially planar, evaluating the triangulated solid would render polygon-level planarity assessment meaningless. We therefore merge triangles belonging to the same original face, trace exterior and interior boundary rings with consistent orientation, and remove collinear points introduced by the union. The output is one CityJSON solid per building with semantic surfaces, which is the representation evaluated.

3.3 Experimental setup

We apply BWFormer with the published weights. Since its training data is cropped to the annotated roof structures (Section 3.1), we match this input assumption on Vaihingen by removing facade points before inference: a point is discarded if the slope to any neighbour within a 1 m horizontal radius that is at least 5 cm higher exceeds 70°, which removes near-vertical wall returns while preserving continuously sloped roofs. All post-processing applied to BWFormer outputs runs through the pipeline of Section 3.2, and every intervention it makes (repaired wireframes, rejected buildings, dropped faces) is recorded per building. Since the Building3D test annotations are not publicly released, we use a random sample of the training set of the same size as the ground-truth reference. Point2Building normalises input point clouds with parameters derived from the target mesh, which is unavailable in deployment. For Zurich we therefore evaluate the author-provided predictions; for Vaihingen, where none exist, we recompute predictions with normalisation derived from the input point cloud. All figures and tables follow this split. Point2Building outputs are evaluated without post-processing.

3.4 Validity evaluation

We assess geometric and topological validity with val3dity (Ledoux, 2018), which implements the ISO 19107 definitions of valid solids as used by CityGML-based city models. A building counts as valid if val3dity reports no error; we additionally record all error codes per building. BWFormer outputs are validated on the merged polygonal solids, since validating the triangulated solids would trivialise the planarity criterion, and the merged representation is what a city model stores. Validity rates are reported against the total number of input buildings, so buildings lost at any pipeline stage count as invalid; rates over survivors, and rates excluding buildings with dropped faces, are given alongside.

We use val3dity's default tolerances (1 cm distance-to-plane, 20° normal deviation). In order to assess the sensitivity to the planarity tolerance, we report distribution of vertex-to-plane distances over multiple thresholds for every method and dataset (for BWFormer: on the extracted faces before planarity optimisation, since afterwards planarity holds by construction),

Stage		Predictions	Ground truth
Input wireframes (3,472 / 3,472)	well-formed	90.3%	98.7%
	≥1 degree-1 vertex	8.5%	1.1%
	≥1 bridge	9.7%	1.3%
Repair (3,386 / 3,463)	was repaired	7.2%	1.0%
	rejected	2.5%	0.2%
	mean edges added / building	0.013	0.003
	mean edges dropped / building	0.150	0.002
Face extraction (3,386 / 3,463)	faces consistent with Euler total	95.3%	99.6%
Planarisation (3,382 / 3,463)	displacement xy, median / p95 (cm)	1.1 / 19.6	0.9 / 12.4
	displacement z, median / p95 (cm)	3.5 / 35.5	2.8 / 17.7
	not converged (10 iterations)	1.1%	1.1%
Final solids (3,373 / 3,463)	topology changed (repair or dropped face)	10.9%	2.2%

Table 1. BWFormer predictions versus ground-truth wireframes under identical processing (Building3D). The left column names the pipeline stage and the number of buildings surviving it (predictions / ground truth). Well-formedness is measured before repair; a wireframe is well-formed if it has no degree-1 vertices and no bridges. Displacements refer to the planarity optimisation and give the largest movement of any vertex per building (median / 95th percentile across buildings). Topology changed: buildings whose final solid required wireframe repair or lost at least one face during solidification.

and re-validate Point2Building outputs with the planarity check relaxed to 10cm. We apply the identical evaluation to the ground truth of each dataset, which establishes what the supervision provides and separates pipeline-induced from prediction-induced failures. We further record per-building wall-clock times for inference and post-processing, and extrapolate linearly to 100,000 buildings.

4. Results

We report results following the data through the pipeline: we first assess the validity of the ground truth itself (Section 4.1), then the quality of the predicted wireframes and the effort of face extraction and repair (Section 4.2), polygon-level planarity for both methods (Section 4.3), solid-level validity (Section 4.4), and finally computational cost (Section 4.5). All results for BWFormer include our post-processing pipeline (Section 3.2) and characterise structural validity, not geometric accuracy. Unless stated otherwise, percentages are given with respect to the total number of input buildings, so buildings lost at any pipeline stage count as failures.

4.1 Validity of the ground truth

The Building3D wireframe annotations are structurally sound but geometrically imperfect: 98.7% are well-formed as graphs and 99.6% yield face counts consistent with the Euler-implied total (Table 1), yet the faces are not planar, with 37.2% deviating more than val3dity's 1 cm default tolerance from their best-fit plane (mean 1.4 cm, Table 2). The Zurich meshes exhibit the opposite behaviour: they are almost perfectly planar; however, only 2.9% of them constitute valid solid models. The dominant errors, unclosed shells (error 302, 1,330 of 2,691 buildings) and multiple connected components (305, 1,177), follow directly from the modelling convention that represents roof overhangs as single polygons extending beyond the wall tops. Closure is not a property of the training data, and consequently not one the supervision can teach. The Vaihingen reference is a roof model, not a building model. It provides per-face polygon sets, so solid-level validity is not defined on it directly. We therefore pass it through our own solidification pipeline, after which 68.2% of the reference buildings yield valid solids. Repeated consecutive points are the most frequent remaining error, left behind by the triangulation and boolean union and thus produced by our pipeline rather than present in the data. True geometric errors are rare, with non-manifold cases (8 instances) and cases of multiple connected components (6 instances) , in which roof segments at different elevation levels either extrude into distinct components or touch along a single edge. Without the planarity optimisation, only 21.8% of reference buildings are valid, with non-planar polygons dominating (84 of 110): the surveyed polygons themselves are not exactly planar.

None of the three datasets, therefore, provides valid solids as supervision or reference.

4.2 From wireframes to solids

Table 1 contrasts BWFormer predictions with the Building3D ground-truth wireframes under identical processing, and Figure 2 illustrates the pipeline on four test buildings. On ground truth, the pipeline operates almost without intervention: 1.0% of annotation wireframes are repaired, 0.2% rejected, and not a single building is lost downstream, with all 3,463 extracted face sets surviving planarisation and solidification. Failures on predictions are therefore attributable to the predictions, not to the pipeline.

The predictions required more editing. Before repair, 8.5% contain at least one dangling vertex and 9.7% at least one bridge. The dominant repair operation is dropping edges (0.150 per building versus 0.002 for ground truth). Repair recovers most malformed wireframes, while 2.5% of buildings are rejected and count as invalid. Further buildings are lost at planarisation (4) and solidification (9), leaving 3,373 of 3,472 buildings (97.1%) with a final solid. In total, 10.9% of these required some topology intervention (wireframe repair or a face dropped during solidification), rising to 25.3% on Vaihingen. For these buildings the resulting solid no longer accurately reflects the original prediction, regardless of its validity (see Figure 2).

4.3 Planarity

Table 2 reports the distribution of face planarity errors (vertex distance to the best-fit plane of its face) for both methods and

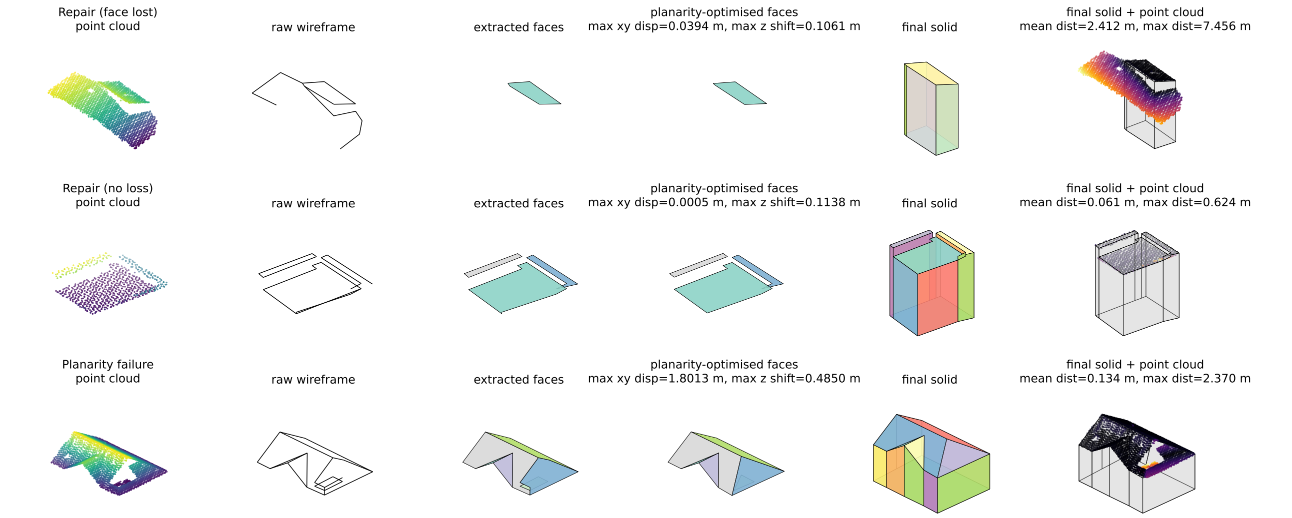

Figure 2. The wireframe-to-solid pipeline on three Building3D test buildings (columns: point cloud, raw wireframe, extracted faces, planarity-optimised faces, final merged solid, solid with point-cloud distances). Rows illustrate distinct behaviours: a pruned dangling edge that removes an important roof face, a bridged dangling edge recovering full topology, and a case where planarity optimisation did not converge.

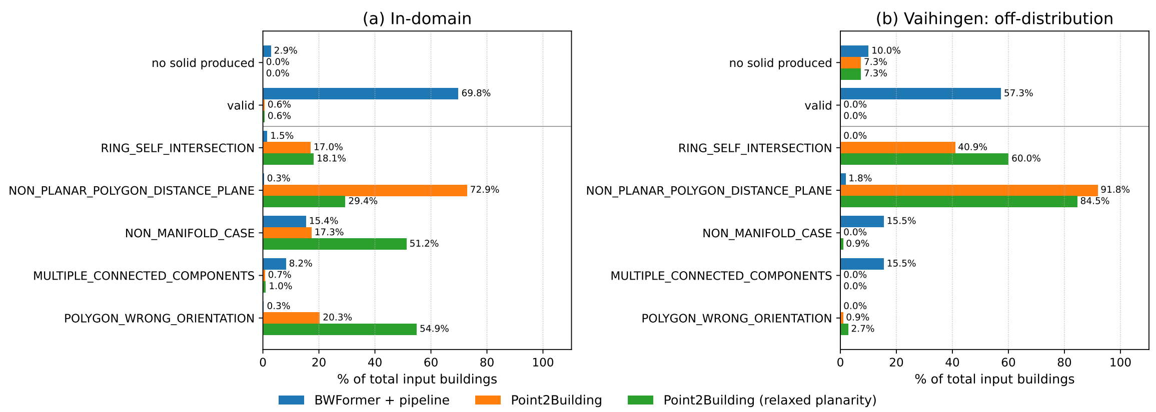

Figure 3. Solid validity and val3dity error codes as the percentage of total input buildings (a building may exhibit several codes, so bars do not sum to the invalid percentages). (a) In-domain (BWFormer: Building3D. Point2Building: Zurich author-provided predictions.). (b) Vaihingen. Relaxed planarity: identical outputs re-validated with the planarity tolerance relaxed to 10 cm (from default 1 cm), which leaves validity unchanged and exposes the errors beneath.

their ground truth at val3dity's default tolerance and two relaxed thresholds.

In-domain, each method approximately reproduces its supervision (Section 4.1): BWFormer exceeds the ground-truth shares at every threshold (42.5% versus 37.2% above 1 cm, 10.6% versus 5.0% above 5 cm), and Point2Building, despite exactly planar training meshes, produces 11.7% of faces above 1 cm. Off-distribution, both perform significantly worse. On Vaihingen, 60.4% of BWFormer faces exceed 1 cm (mean 5.9 cm), and Point2Building reaches a mean of 13.3 cm with a 99th percentile of almost 2 m. Without the facade filtering of Section 3.3, BWFormer planarity errors reach up to 12 m, which is due to edges lost due to lower prediction confidence.
The planarity optimisation resolves the BWFormer deviations at low geometric cost (Table 1): the largest displacement of any vertex has a median of 1.1 cm planimetrically and 3.5 cm in height per building (95th percentile: 20 and 36 cm), barely more than on the ground-truth wireframes themselves (0.9 and 2.8 cm).

4.4 Solid validity

Figure 3 summarises solid-level validity and the val3dity error codes; Figures 4 and 5 show representative cases.
For BWFormer, 69.8% of the 3,472 test buildings yield valid solids (64.2% when buildings with topology changes are excluded). The dominant errors, non-manifold cases (15.4% of buildings) and multiple connected components (8.2%), share one mechanism: adjacent building parts predicted with small gaps or overlaps, which the boolean union turns into touching but unmerged shells (Figure 4). Planarity errors are nearly absent because Stage 2 resolves them by construction; without planarity optimisation, validity collapses to 21.3%.

For Point2Building on Zurich, 0.6% of the provided predictions are valid. Under the deployment normalisation, validity drops further to 0.2% (6 of 2,691), so the provided predictions represent the method's best case. The dominant errors are non-planar polygons (72.9% of buildings), wrong polygon orientation (20.3%), non-manifold cases (17.3%) and ring self-intersections (17.0%) (Figure 5). Although non-planarity is the

Dataset	Faces	$>$ 1 cm	$>$ 5 cm	$>$ 10 cm	Mean	Median	P99
BWFormer/Building3D test (pre-opt)	13,791	42.51%	10.56%	4.29%	2.31 cm	0.72 cm	23.8 cm
Building3D train subset (GT)	15,481	37.23%	5.00%	1.36%	1.39 cm	0.54 cm	12.9 cm
BWFormer/Vaihingen (pre-opt)	376	60.37%	26.06%	15.69%	5.86 cm	2.00 cm	51.8 cm
Vaihingen reference (GT)	591	38.24%	10.32%	2.37%	1.82 cm	0.29 cm	14.8 cm
Zurich ground truth	45,927	0.00%	0.00%	0.00%	0.00 cm	0.00 cm	0.0 cm
Point2Building/Zurich (provided)	36,136	11.66%	4.84%	2.74%	1.23 cm	0.00 cm	25.6 cm
Point2Building/Vaihingen	2,287	37.43%	27.94%	22.30%	13.33 cm	0.00 cm	196.6 cm

Table 2. Face planarity errors as vertex distance to the face's best-fit plane: share of faces exceeding val3dity's default tolerance (1 cm) and two relaxed thresholds, with mean, median and 99th percentile. BWFormer faces are measured before planarity optimisation, since afterwards planarity holds by construction. Point2Building meshes include floors and walls, which are easier to predict planar than roof faces, so its shares are not directly comparable to the roof-only rows.

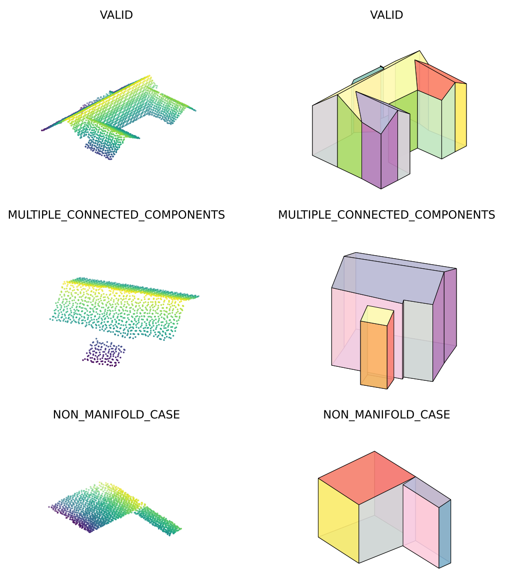

Figure 4. BWFormer cases (point cloud and final solid through our pipeline). Valid meshes (top) against the dominant error patterns.

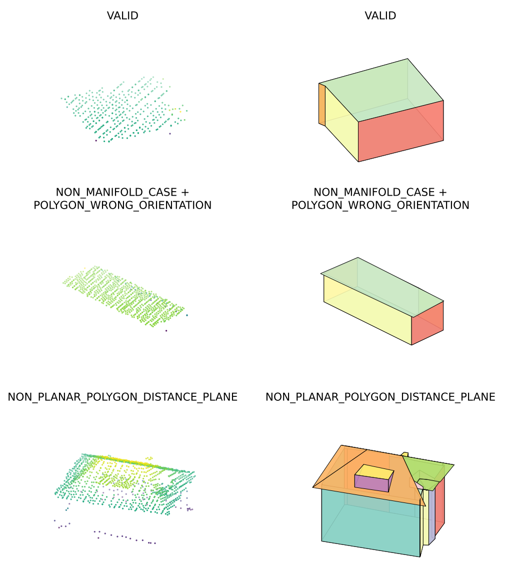

Figure 5. Point2Building cases (point cloud and mesh as provided). Valid meshes (top) against the dominant error patterns.

most common issue, it is not the limiting factor for validity as relaxing the planarity requirement in the re-validation does not change which buildings are valid. Instead, it reveals additional underlying errors, with non-manifold cases increasing to 51.2% and wrong orientation affecting 54.9% of buildings, since val3dity's hierarchical procedure would otherwise stop at the polygon stage.

On Vaihingen, BWFormer reaches 57.3% validity against the 68.2% that the surveyed reference achieves through the same pipeline: the site and pipeline admit substantially more than the predictions deliver, and the gap carries the in-domain error profile of non-manifold contacts and multiple components between building parts (15.5% each).

None of the Point2Building outputs generated for the Vaihingen dataset are valid, irrespective of whether the planarity constraint is enforced. The majority of building instances contain self-intersecting rings. Figure 6 shows the reconstructed scene. While BWFormer is able to reconstruct generally recognisable building structures with less detail compared to the reference geometries, Point2Building fails to produce coherent reconstructions.

4.5 Computational cost

All timings were measured on a single machine with an NVIDIA GeForce RTX 4060 Ti (16 GB, CUDA 12.6), an AMD Ryzen 9 7900X and 64 GB RAM. Of the pipeline, only face extraction parallelises the buildings across CPU cores (16 workers); inference, planarity optimisation, solidification, and triangle merging were processed one by one. BWFormer inference takes 0.053 s per building on average (Vaihingen: 0.058 s). Post-processing adds 0.138 s per building, dominated by the boolean union, for a full-pipeline total of 0.19 s per building, or roughly 5.3 h extrapolated to 100,000 buildings. Point2Building requires 1.44 s per building on Zurich and 3.67 s on Vaihingen, with no post-processing. Its autoregressive decoding with re-sampling on failed plausibility checks makes runtime highly variable, ranging from 0.23 to 21.7 s per building on Zurich. Extrapolated to 100,000 buildings, this corresponds to roughly 40 h.

5. Discussion

Both methods were published with strong accuracy: BWFormer reports 88.4 % corner and 79.4 % edge F1 with a mean corner offset of 0.20 m on the Tallinn test split, Point2Building 86.8 % vertex F1 with a mean distance error of 0.25 m on the Zurich test split. These metrics measure distances between predicted and reference primitives, and no distance metric captures whether

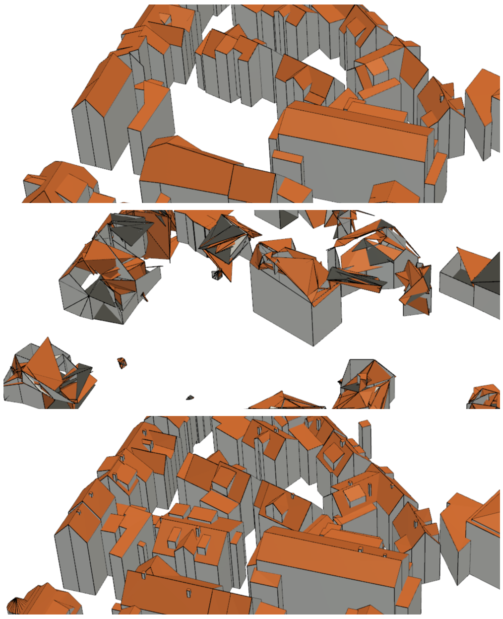

Figure 6. Reconstructed Vaihingen scene. Top: Solids derived from BWFormer predictions. Middle: Point2Building predictions. Bottom: Solids derived from the reference model. For the reference and BWFormer we extrude to a fixed ground plane 10 m below, as only roofs are available; Point2Building estimates ground height from the point cloud.

the primitives assemble into a closed, manifold, consistently oriented solid: the same predictions yield 69.8 % and 0.6 % valid solids. Benchmark evaluation as currently practised is blind to the properties downstream applications require, and validity rates should be reported alongside accuracy.

Neither output space is constrained to valid solids. BWFormer classifies edges independently, so nothing enforces that they close into face loops: 8.5 % of predicted wireframes contain dangling vertices and 9.7 % bridges, and its residual solid errors are gaps between building parts. Where our pipeline intervenes topologically, the result is a solid that validates but no longer reflects the prediction. Practitioners should treat repaired outputs as flagged for review rather than repair them. Point2Building's sequential decoding yields closed polygons, but they are not prevented from producing a self-intersecting ring, an inward-facing polygon, or a non-manifold contact, which are the dominant errors that persist when the planarity check is relaxed (Section 4.4).

Even perfect learning would not have produced valid solids, because the training data does not contain them: only 2.9 % of the Zurich meshes are valid, with closure broken by the overhang modelling convention, and the Building3D wireframes are a roof-only format whose faces are not even planar (Section 4.1). Dataset conventions are silently baked into what the models reproduce. Training on valid data would likely help, but by the previous paragraph it would not suffice: validity would remain a statistical tendency rather than a guarantee.

Every vertex-predicting representation faces a dilemma: exact vertices make faces beyond triangles non-planar in general, while exact planes make intersections miss their common corners, so the apex of a pyramid roof dissolves into nearby points. For wireframes the dilemma is resolvable after the fact, since roof planes admit a height-function parametrisation with displacements shared per corner: our optimisation restores exact planarity with median displacements below the method's reported corner accuracy of 0.20 m, provided the prediction is plausible to begin with. The topological errors of both methods admit no such treatment.

Beyond what the models learned, serving them requires reproducing their training conditions. BWFormer performs better when facade returns are removed to match its roof-cropped training data. Point2Building performs poorly under slight changes of normalisation: replacing the target-mesh-derived parameters with input-derived ones, as deployment requires, drops validity from 0.6 % to 0.2 %.

6. Conclusion

Evaluated on their own test data, BWFormer yields 69.8 % valid solids after post-processing, Point2Building 0.6 % without; neither training dataset contains valid solids, and neither architecture constrains its output space to them. Validity therefore cannot be expected from current end-to-end methods, but it can be measured: unlike accuracy, it requires no reference data and is verifiable per building from the output alone, as are the structural interventions of a post-processing pipeline such as ours. We conclude that validity should be reported alongside accuracy in benchmarks, enforced where possible through representations or training objectives, and verified at production time where it is not. Until learning-based methods meet these requirements, geometric post-processing with structural guarantees remains necessary for standard-compliant city models.

References

Abdelhedi, Y., Panangian, D., Amrullah, C., Chaabouni-Chouayakh, H., Bittner, K., 2026. LoD2-Former: Multi-Modal Transformer-Based 3D Building Wireframe Reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XI-2-2026, 187–195.

Akwensi, P. H., Bharadwaj, A., Wang, R., 2025. Points2Model: a neural-guided 3D building wireframe reconstruction from airborne LiDAR point clouds. *International Journal of Digital Earth*, 18(1), 2458682.

Chen, Z., Ledoux, H., Khademi, S., Nan, L., 2022. Reconstructing compact building models from point clouds using deep implicit fields. *ISPRS Journal of Photogrammetry and Remote Sensing*, 194, 58–73.

Chen, Z., Shi, Y., Nan, L., Xiong, Z., Zhu, X. X., 2024. PolyGNN: Polyhedron-based graph neural network for 3D building reconstruction from point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 218, 693–706.

Dukai, B., Peters, R., Vitalis, S., Van Liempt, J., Stoter, J., 2021. Quality Assessment of a Nationwide Data Set Containing Automatically Reconstructed 3D Building Models. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVI-4/W4-2021, 17–24.

Fan, K.-C., Chang, C.-Y., 1991. Surface extraction from line drawings of a polyhedron. *Pattern Recognition Letters*, 12(10), 627–633.

Gao, W., Peters, R., Stoter, J., 2024. Building-PCC: Building Point Cloud Completion Benchmarks. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-4/W5-2024, 179–186.

Geniet, F., Séguin, E., Le Bihan, E., Duan, L., Girard, N., Peters, R., Vallet, B., 2025. A new benchmark on LoD 2 building reconstruction from aerial lidar and footprints. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-1/W6-2025, 83–90.

Gong, H., Liu, J., Tong, R., Tian, F., Wang, D., 2025. Robust building wireframe reconstruction: a hypergraph and transformer-enhanced framework for large-scale and real-world urban point clouds. *International Journal of Remote Sensing*, 46(24), 9565–9596.

Huang, J., Stoter, J., Peters, R., Nan, L., 2022. City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds. *Remote Sensing*, 14(9), 2254.

Huang, S., Wang, R., Guo, B., Yang, H., 2023. PBWR: Parametric Building Wireframe Reconstruction from Aerial LiDAR Point Clouds.

Huang, S., Wang, R., Wang, X., 2025. BuildingWorld: A Structured 3D Building Dataset for Urban Foundation Models.

Jacobson, A., Panozzo, D. et al., 2018. libigl: A simple C++ geometry processing library. Version 2.1.0.

Jarzabek-Rychard, M., Rigon, S., Boguslawski, P., Remondino, F., 2025. Automating 3D Building Modeling: A Comparative Study of Data- and Model-Driven Methods. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-G-2025, 701–707.

Ledoux, H., 2018. val3dity: validation of 3D GIS primitives according to the international standards. *Open Geospatial Data, Software and Standards*, 3(1), 1.

Li, L., Song, N., Sun, F., Liu, X., Wang, R., Yao, J., Cao, S., 2022. Point2Roof: End-to-end 3D building roof modeling from airborne LiDAR point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 193, 17–28.

Liu, Y., Obukhov, A., Wegner, J. D., Schindler, K., 2024. Point2Building: Reconstructing buildings from airborne LiDAR point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 215, 351–368.

Liu, Y., Zhu, L., Ye, H., Huang, S., Gao, X., Zheng, X., Shen, S., 2025. BWFormer: Building Wireframe Reconstruction from Airborne LiDAR Point Cloud with Transformer. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 22215–22224.

Liu, Y., Zhu, L., Ye, H., Liu, Y., Huang, S., Gao, X., Wang, R., Shen, S., 2026. Buildinggpt: Auto-regressive building wireframe reconstruction model with reinforcement learning. *Proceedings of the Computer Vision and Pattern Recognition Conference*, 36400–36410.

Ma, X., Liu, Y., Zhou, W., Wang, R., Huang, H., 2025. Generating 3D House Wireframes with Semantics. *Computer Vision – ECCV 2024*, 15080, Springer Nature Switzerland, Cham, 223–240.

Nash, C., Ganin, Y., Eslami, S. M. A., Battaglia, P. W., 2020. PolyGen: An Autoregressive Generative Model of 3D Meshes. *ICML*.

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 & Remote Sensing*, 88(3), 165–170.

Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J. D., Breitkopf, U., Jung, J., 2014. Results of the ISPRS benchmark on urban object detection and 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, 256–271.

Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Benitez, S., Breitkopf, U., 2012. The ISPRS Benchmark on Urban Object Classification and 3D Building Reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, I-3, 293–298.

Schuegraf, P., Fuentes Reyes, M., Xu, Y., Bittner, K., 2023. ROOF3D: A Real and Synthetic Data Collection for Individual Building Roof Plane and Building Sections Detection. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-1/W1-2023, 971–979.

Shewchuk, J. R., 1996. Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator. G. Goos, J. Hartmanis, J. Van Leeuwen, M. C. Lin, D. Manocha (eds), *Applied Computational Geometry Towards Geometric Engineering*, 1148, Springer Berlin Heidelberg, Berlin, Heidelberg, 203–222.

The CGAL Project, 2019. *CGAL User and Reference Manual*. 5.0 edn, CGAL Editorial Board.

Varley, P. A. C., Company, P. P., 2010. A new algorithm for finding faces in wireframes. *Computer-Aided Design*, 42(4), 279–309.

Wang, R., Huang, S., Yang, H., 2023. Building3d: A urban-scale dataset and benchmarks for learning roof structures from point clouds. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 20076–20086.

Zang, Y., Mi, W., Xiao, X., Guan, H., Chen, J., Li, D., 2024. Compound 3D building modeling with structure-aware partition and primitive assembly from airborne laser scanning point clouds. *International Journal of Digital Earth*, 17(1), 2375112.

Zhou, Q., 2018. PyMesh — geometry processing library for Python. https://pymesh.readthedocs.io/en/latest/. Accessed: 2026-06-20.

Zhou, Q., Grinspun, E., Zorin, D., Jacobson, A., 2016. Mesh arrangements for solid geometry. *ACM Transactions on Graphics*, 35(4), 1–15.