Automatic Scan-to-BIM: The Impact of Semantic Segmentation Accuracy on Opening Detection

Jidnyasa Patil[1], Arcot Sowmya[2], Mohsen Kalantari[3]

[1]School of Civil and Environmental Engineering, University of New South Wales, Sydney, 2052, NSW, Australia - jidnyasa.patil@unsw.edu.au
[2]School of Computer Science and Engineering, University of New South Wales, Sydney, 2052, NSW, Sydney, Australia - a.sowmya@unsw.edu.au
[3]School of Civil and Environmental Engineering, University of New South Wales, Sydney, 2052, NSW, Australia - mohsen.kalantari@unsw.edu.au

Keywords: semantic segmentation; deep learning; 3D reconstruction; Scan-to-BIM; BIM; openings.

Abstract

The automation of Scan-to-BIM remains a major challenge within the Architecture, Engineering, and Construction industry, particularly in the detection and geometric characterisation of architectural openings such as doors and windows. Although recent advances in 3D semantic segmentation have improved the classification of architectural elements, the effect of segmentation accuracy on downstream geometric detection and reconstruction is still under study. This work compares five state-of-the-art deep learning models, PointNeXt, PointMetaBase, Point Transformer V1, Point Transformer V3, and Swin3D, on opening detection in Scan-to-BIM. A unified evaluation framework integrating DBSCAN clustering with axis-aligned bounding box fitting is introduced to generate per-instance geometric representations. The models are assessed using semantic metrics and geometric reliability indicators, including centroid error, dimensional deviation and 3D IoU. Experiments on the S3DIS Area 5 dataset, reveal notable performance differences across models. Swin3D achieved the highest door detection rate of 96.9%, followed by PointMetaBase at 92.9%, PointNeXt at 87.4%, PTV3 at 85.0%, and PTV1 at 81.9%. Window detection proved more challenging, with Swin3D and PTV3 both achieving 75.0%, PTV1 at 71.2%, and PointNeXt and PointMetaBase at 67.3%. Notably, PointMetaBase produced strong geometric accuracy for doors despite lower semantic scores. These results suggest that high segmentation accuracy does not always lead to precise geometric reconstruction. To assess generalisation, the trained models were applied to 11 Matterport3D rooms, confirming that the observed patterns extend across different scanning environments. This study concludes that in Scan-to-BIM workflows, greater emphasis should be placed on geometric reconstruction algorithms than segmentation performance alone.

1. Introduction

1.1 Motivation

The digital transformation of the architecture, engineering and construction (AEC) industry has led to increased adoption of Building Information Modelling (BIM) as a standard for managing, designing and maintaining built assets. However, the generation of BIM models for existing buildings remains time-consuming, particularly when relying on manual processes. Laser scanning and photogrammetry have enabled the capture of high-resolution indoor point clouds, however translating these unstructured data into structured, semantically rich models remains an open research challenge. Automating this process, known as Scan-to-BIM, has seen promising developments for structural elements like walls, floors and ceilings, however the detection and reconstruction of architectural openings such as doors and windows is less developed (Adekunle et al., 2022).

Doors and windows play a crucial role in accessibility, ventilation and energy performance. However, their relatively small size, placement variability and frequent occlusion make them more challenging to detect than walls or floors. Moreover, these elements are often co-planar with adjacent surfaces, challenging traditional geometric fitting approaches. Although deep learning-based semantic segmentation has improved point-level classification accuracy, it remains unclear how these improvements influence the detection and geometric reliability of specific architectural elements (Borruso et al., 2023).

This study investigates the relationship between semantic segmentation accuracy and opening detection accuracy. Using five state-of-the-art 3D segmentation models, a unified pipeline is implemented that performs semantic filtering, density-based clustering and axis-aligned bounding box fitting to isolate and characterise individual doors and windows in indoor scenes. The training and evaluation of the methods focus on Area 5 of the Stanford Large-Scale Indoor Spaces (S3DIS) dataset, a widely used benchmark containing diverse room types and architectural conditions (Armeni et al., 2016). Both semantic and geometric metrics are reported, providing a detailed assessment of detection completeness and spatial accuracy. This study seeks to understand how different segmentation models perform across architectural element types and how segmentation accuracy affects geometric instance detection. In doing so, it contributes to a more nuanced understanding of the link between semantic predictions and Scan-to-BIM reconstruction outcomes.

The contributions of this study may be summarised as follows. First, it presents a cross-model benchmark comparing five state-of-the-art 3D semantic segmentation architectures on the task of opening detection. Second, it introduces a lightweight pipeline for extracting per-instance door and window geometry directly from semantically segmented point clouds. Finally, it provides a detailed comparative analysis that examines how semantic segmentation accuracy influences the quality of instance detection and geometric characterisation across models and element types.

The paper is structured as follows: section 1.1 outlines the motivation for this work, while section 1.2 reviews prior research. In section 2 the dataset, models and detection pipeline are described, while section 3 presents the results and analysis. The implications across semantic and geometric domains are discussed in section 4, and section 5 concludes with directions for future work.

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-XI-1-2026-437-2026

1.2 Previous Research Work

Scan-to-BIM consists of two major tasks: semantic segmentation and 3D reconstruction. Semantic segmentation is the process of classifying point cloud data into pre-defined classes. Semantic segmentation of point clouds has evolved considerably in recent years, particularly for indoor scene understanding. Early approaches relied on voxel-based representations and 3D convolutional neural networks (CNN) such as VoxNet (Maturana and Scherer, 2015) and OctNet (Riegler et al., 2017), which processed spatial data on regular grids. These models enabled initial breakthroughs in object recognition but were computationally expensive and often lost geometric detail, particularly in large-scale or cluttered environments.

The introduction of PointNet (Charles R Qi et al., 2017) and PointNet++ (Charles R. Qi et al., 2017) marked a significant turning point by enabling direct processing of unordered point sets. These models used shared multi-layer perceptron (MLP) and hierarchical feature abstraction to capture both global and local context without voxelisation. Follow-up models such as PointCNN (Li et al., 2018), DGCNN (Wang et al., 2019) and RandLA-Net (Hu et al., 2020) improved on this foundation by incorporating local neighbourhood aggregation and improved spatial encoding.

More recently, PointNeXt (Qian et al., 2022) and PointMetaBase (Lin et al., 2023) have re-examined the MLP-based paradigm, introducing architectural enhancements such as residual blocks, deeper hierarchies and modular design. These modifications have led to increased segmentation accuracy while maintaining computational efficiency. In parallel, attention-based architectures have emerged as powerful alternatives. Point Transformer (Engel et al., 2021) and Point Transformer V3 (Wu et al., 2023) introduced vector self-attention and patch-based spatial grouping, enabling models to learn long-range geometric dependencies. Swin3D (Yang et al., 2023), a voxel-transformer hybrid, combines sparse voxelisation with shifted window attention and contextual signal encoding, achieving strong performance on indoor benchmarks such as S3DIS and ScanNet (Armeni et al., 2016; Dai et al., 2017).

Despite these architectural advances, a critical gap remains in how these models are evaluated. While significant gains have been made on pointwise classification, most segmentation models are evaluated using class-agnostic metrics such as overall accuracy, mean class accuracy, or mean Intersection-over-Union (mIoU). These metrics, although useful for benchmark comparisons, do not directly assess the geometric consistency, instance-level grouping or reconstruction capability of segmented outputs. These are the key requirements for downstream BIM modelling (Patil and Kalantari, 2025). Large structural elements such as walls often dominate performance metrics due to their size and regularity, while minority classes like doors and windows remain challenging due to occlusions, co-planarity and inconsistent labelling in training datasets.

Several prior studies have focused on surface-based reconstruction of architectural elements. For example, plane fitting techniques such as RANSAC (Mahmoud et al., 2024), MLESAC (Torr and Zisserman, 2000) and region growing (Hojjatoleslami and Kittler, 1998) have been employed to extract dominant wall surfaces after semantic segmentation. While wall reconstruction has received substantial attention, the reconstruction of architectural openings such as doors and windows remains a relatively underexplored area. Traditional detection methods rely on identifying voids or geometric discontinuities within reconstructed wall surfaces, often using simple heuristics or bounding box rules (Pexman et al., 2021; Previtali et al., 2018). These approaches typically assume clean wall reconstructions and fail in the presence of clutter, incomplete scans or co-planar features. Furthermore, many do not incorporate semantic cues from deep learning outputs, making them susceptible to false positives in complex environments (Macher et al., 2017).

Recent research has begun to integrate semantic segmentation with object-level detection. Approaches such as Scan2BIM-Net (Perez-Perez et al., 2021), BIM-modules for IFC (Industry Foundation Classes) parametric reconstruction (Mahmoud et al., 2024) and automated frameworks for complex indoor modelling combine segmentation with post-processing strategies such as clustering, procedural modelling or rule-based filtering (Zbirovský and Nežerka, 2025). However, these efforts often focus on large-scale elements or assume simplified opening geometries.

As seen from the literature, significant strides have been made in the development of deep learning models for semantic segmentation over the past decade, enabling more precise and efficient point cloud analysis. However, for BIM reconstruction, extracting measurements of architectural elements from segmented point clouds and 3D reconstruction of these elements are two critical processes for accurate model building of existing structures. Given these limitations in existing approaches, a fundamental question arises: how does semantic segmentation accuracy influence the correctness of reconstructed models?

A recent study investigated this relationship by evaluating how varying levels of segmentation accuracy affect the accuracy of 3D reconstruction, focusing specifically on wall reconstruction, as walls are fundamental structural elements that define the framework of interior spaces (Patil and Kalantari, 2025). Their findings revealed an interesting pattern: even when semantic segmentation accuracy was lower, reconstruction accuracy could still remain high, suggesting that the relationship between segmentation performance and reconstruction quality is not straightforward. Moreover, different deep learning architectures demonstrated varying impacts on reconstruction outcomes, with some models achieving better reconstruction despite weaker segmentation performance.

To determine whether this pattern extends beyond walls, the present study aims to investigate whether similar patterns exist for doors and windows. These architectural elements present distinct challenges compared to walls. Unlike walls, which are large planar surfaces, doors and windows are smaller openings with varied geometries, specific dimensional constraints and frequent occlusions. They represent minority classes in typical indoor point cloud datasets, often comprising a much smaller proportion of labelled points compared to dominant classes like walls, floors and ceilings. Additionally, doors may appear in different states (open, closed, or partially open), while windows exhibit diverse configurations (single-pane, multi-pane, with or without frames) and material properties (glass reflectance, varied frame materials) that complicate accurate detection and reconstruction.

This study evaluates how varying levels of segmentation accuracy for doors and windows affect the accuracy of their 3D reconstruction. We compare the performance of five deep learning models: PointNeXt, PointMetaBase, Point Transformer V1, Point Transformer V3 and Swin3D, examining their influence on opening reconstruction. Our goal is to examine how the architecture of the deep learning algorithms impacts door and window reconstruction, extending the understanding established for walls to these more challenging architectural elements. This would enable future researchers to select appropriate algorithms for improving the Scan-to-BIM process, particularly for applications where accurate opening reconstruction is critical for operations, maintenance, accessibility analysis or energy modelling.

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-XI-1-2026-437-2026 | © Author(s) 2026. CC BY 4.0 License.

2. Method

This section describes the dataset, semantic segmentation model training, and the pipeline used to extract and evaluate door and window instances from 3D point clouds.

2.1 Dataset and Preprocessing

All experiments were conducted on the Stanford Large-Scale Indoor Spaces (S3DIS) dataset, a widely used benchmark for indoor point cloud understanding. The dataset comprises six large areas scanned from three different educational buildings. Each point includes 3D spatial coordinates (x, y, z), colour information (RGB) and a semantic label from one of 13 original categories: ceiling, floor, wall, beam, column, window, door, table, chair, sofa, bookcase, board, and clutter.

For this study, a relabelling strategy was employed to reduce noise and enhance the focus on architectural elements. Following established protocols from prior Scan-to-BIM research (Patil and Kalantari, 2025), all furniture-related classes (table, chair, sofa, bookcase, and board) were merged into the "clutter" category. This resulted in a simplified 8-class schema: ceiling, floor, wall, beam, column, window, door, and clutter. This relabelling reduces the class imbalance and increases model generalisability for reconstruction tasks. Models were trained to segment only these eight classes, allowing for more accurate downstream detection of architectural elements such as doors and windows.

For evaluation, Area 5 of S3DIS was held out as the test set, while Areas 1 through 4 and Area 6 were used for training. Area 5 includes a diverse mix of spaces such as offices, conference rooms, hallways and storage areas, and exhibits realistic scanning challenges including partial occlusions, non-uniform point density and clutter. For opening detection, only door and window classes were used, with Area 5 containing 127 annotated door instances and 52 window instances. Each opening is uniquely identified in the dataset and used as ground truth for evaluating detection and geometric accuracy.

2.2 Deep Learning Models

To evaluate architectural opening detection, five state-of-the-art semantic segmentation networks were selected. These represent a range of point cloud processing strategies: pointwise MLP, attention-based feature aggregation and voxelised transformer architectures. Each model was trained from scratch on the relabelled 8-class S3DIS dataset.

2.2.1 PointNeXt

PointNeXt is an evolution of the PointNet++ architecture, incorporating modern training techniques and efficient building blocks. It processes point clouds in a hierarchical manner, progressively learning features from local neighbourhoods before combining them into broader spatial understanding. The model uses Farthest Point Sampling (FPS) to select representative points and groups nearby points around these centres to capture local geometric patterns. The network employs inverted residual MLP blocks, inspired by MobileNetV2, and depth-wise separable MLP to reduce computational load. PointNeXt was trained on the axis-aligned version of S3DIS for 100 epochs with a batch size of 8.

2.2.2 PointMetaBase

PointMetaBase abstracts common operations in point cloud processing into four meta-functions: neighbour update, neighbour aggregation, point update and positional embedding. It uses lightweight MLP, max pooling, and explicit spatial encodings. The model is simple but effective, offering good generalisation. It was also trained on the axis-aligned S3DIS dataset for 100 epochs with a batch size of 8.

2.2.3 PointTransformer V1

Point Transformer V1 (PTV1) introduces vector self-attention mechanisms adapted for point clouds. For each point, attention weights are computed based on feature and positional differences with neighbours. These allow flexible, context-aware feature aggregation. Relative positional encodings are incorporated using a learnable transformation. PTV1 was trained for 3000 epochs with a batch size of 4 on the raw S3DIS dataset.

2.2.4 Point Transformer V3

Point Transformer V3 (PTV3) addresses the scalability issues of V1 by replacing repeated KNN with patch-based processing. Points are arranged using space-filling curves such as Z-order, allowing patch-level attention. Local and global context is preserved with reduced memory cost. PTV3 was trained on raw S3DIS for 3000 epochs with a batch size of 2.

2.2.5 Swin3D

Swin3D extends Swin Transformer concepts to sparse voxelised point clouds. The model uses multiscale voxel grids, sparse 3D convolutions, and shifted window attention blocks. Contextual Relative Signal Encoding (cRSE) further enhances spatial and appearance feature integration. It was trained for 3000 epochs with a batch size of 4 on raw S3DIS input.

These five architectures were chosen for their strong performance on semantic segmentation benchmarks and for covering diverse architectural design spaces: point-based (PointNeXt, PointMetaBase), attention-based (PTV1, PTV3) and voxel-transformer hybrid (Swin3D).

2.3 Opening Detection Workflow

To transition from point-level semantics to instance-level representations suitable for BIM workflows, a four-stage detection pipeline was implemented: semantic filtering, clustering, bounding box fitting and geometry extraction.

2.3.1 Semantic Filtering

Each model outputs pointwise semantic predictions across the eight relabelled classes. For the detection pipeline, only points predicted as "door" or "window" were retained. All other points were excluded from the detection process. This class-level filtering step ensures that downstream clustering focuses on relevant spatial subsets.

2.3.2 Clustering via DBSCAN

Clusters of predicted door / window points were identified using DBSCAN (Deng, 2020), a robust, unsupervised method that does not require a predefined number of clusters. The algorithm was configured with a neighbourhood radius (eps) of 0.15 meters and a minimum sample threshold of 50 points. To reduce false positives, a minimum cluster size was enforced: 4,000 points for doors and 8,000 points for windows. Clusters that did not meet the minimum size criterion were discarded, and the remaining clusters were treated as candidate door or window detections for further geometric processing.

2.3.3 Bounding Box Fitting

Each retained cluster was enclosed in an Axis-Aligned Bounding Box (AABB). The AABB is computed by identifying the minimum and maximum coordinates of all points within the cluster along the X, Y, and Z axes. From each AABB, the following geometric attributes were extracted:

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-XI-1-2026-437-2026 | © Author(s) 2026. CC BY 4.0 License.

Centroid: Midpoint between min and max coordinates
Width, Height, Depth: Extent along X, Y, Z axes

2.3.4 Geometry Extraction

These geometric descriptors serve as foundational parameters for BIM reconstruction. In IFC, openings such as doors and windows are represented as parametric objects defined by their position, dimensions, and orientation. The extracted centroid provides the placement coordinates (IfcLocalPlacement), while width, height, and depth define the opening's geometric representation (IfcShapeRepresentation). Together, these parameters enable the automatic instantiation of standardized IFC entities (IfcDoor, IfcWindow).

2.3.5 IFC Object Instantiation

Each detected opening is instantiated as a parametric IFC entity using IfcOpenShell, an open-source library compliant with the IFC4 schema standard. The spatial structure follows the standard IFC hierarchy of project, site, building and storey. Each cluster is instantiated as either an IfcDoor or IfcWindow entity and assigned to the building storey. The geometry is defined using an IfcBlock primitive, with width, depth and height taken directly from the axis-aligned bounding box. A transformation matrix derived from the bounding box centroid is applied to encode the position and translation of each opening within the world coordinate system.

2.3.6 Validation using Matterport3D Dataset

To assess the cross-dataset generalisation of the proposed framework, the five trained models were evaluated on a subset of the Matterport3D dataset without any fine-tuning or domain adaptation. Matterport3D comprises 90 building-scale scenes captured predominantly in residential environments, representing a substantially different architectural domain from the office and educational spaces of S3DIS. The dataset is organised into train, validation and test splits containing 1,554, 234 and 406 individual rooms respectively, totalling 2,194 rooms across 90 buildings. Each room is stored as a separate folder containing the point cloud and associated annotation files. Eleven rooms were selected from across the three splits based on the presence of doors and windows, with an emphasis on diversity in opening types including curtained, open, closed and partially occluded instances, to assess model robustness across varying real-world conditions.

The Matterport3D annotations were relabelled to match the eight-class S3DIS schema used for training, with the beam and column classes assigned no ground truth instances as these structural elements are not present in the residential environments captured by Matterport3D. The selected rooms were processed according to the input requirements of each trained model and added as Area 7 to the S3DIS dataset structure, allowing the existing testing scripts to be run directly on the Matterport3D data without modification to the evaluation pipeline.

Since the Matterport3D point clouds are sparser than S3DIS scans, the minimum points per cluster threshold used in DBSCAN was tuned individually for each room to account for the reduced point density, ensuring that valid opening instances were not discarded due to data sparsity. This validation assesses whether the segmentation and reconstruction patterns observed on S3DIS are reproducible in unseen environments and provides insight into the robustness of each model architecture under domain shift conditions.

2.4 Evaluation Metrics

For semantic segmentation performance the standard segmentation metrics are reported, including accuracy, mean Intersection over Union (IoU) / mean dice and class specific IoU / class-specific dice for walls, doors and windows. Opening detection performance was evaluated using two metrics: detection completeness and geometric reliability.

2.4.1 Semantic Segmentation Performance

The following standard segmentation metrics are reported for all five models, based on the 8-class relabelled S3DIS schema:

Overall Accuracy (oAcc): The percentage of correctly classified points across the entire test set.

Mean Accuracy (mAcc): The average of per-class pointwise accuracy, giving equal weight to each class.

Mean Intersection-over-Union (mIoU): The average IoU across all eight classes.

Class-specific IoU: Reported specifically for the wall, door and window classes due to their architectural relevance.

Dice (F1): Overlap metric equivalent to the F1 score for a class (1).

$$Dice = \frac{2\ IoU}{1 + IoU} \tag{1}$$

These metrics quantify how well each model segments architectural elements in the point cloud and provide a baseline for understanding the class-wise segmentation challenge.

2.4.2 Detection Performance

Predicted bounding boxes were matched to the ground truth using 3D Intersection-over-Union (IoU). A match is valid if IoU > 0.5. From these matches, standard detection metrics were computed:

True Positives (TP): Matched predictions

False Positives (FP): Unmatched predictions

False Negatives (FN): Unmatched ground truths

From these:

$$Detection\ Rate\ (Recall) = \frac{TP}{(TP + FN)} \tag{2}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{3}$$

$$F1 - Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{4}$$

These metrics were calculated separately for doors and windows, enabling class-specific performance comparison (Eq. (2),(3),(4)).

2.4.3 Geometric Performance

For each true positive detection, geometric error metrics were computed:

Centroid Error: Euclidean distance between predicted and ground truth centroids

Width Error & Height Error: Absolute differences in size along respective axes

3D IoU: Overlap between predicted and ground truth bounding boxes

Mean and standard deviation are reported for each metric across all true positive detections. This provides a measure of both central accuracy and geometric stability.

This comprehensive pipeline supports fair and reproducible comparison of deep learning models in the context of architectural opening detection. It also allows direct quantification of reconstruction suitability, offering insights into how well different model architectures serve downstream BIM automation tasks.

3. Results and Analysis

This section presents the evaluation results for the five semantic segmentation models across semantic performance, instance-level detection completeness, and geometric reliability of opening reconstructions. All results were obtained on Area 5 of the S3DIS dataset. Evaluation was conducted separately for doors and windows to isolate class-specific challenges and trends.

3.1 Semantic Segmentation Performance

Semantic segmentation performance is summarised in Table 1.

Table 1: Semantic Segmentation Performance of five deep-learning models

Model	oAcc	mIoU/ mDice	Wall IoU / Wall Dice	Door IoU / Door Dice	Window IoU / Window Dice
Swin3D	94.83	72.24 / 83.88	88.95 / 94.15	81.73 / 89.95	65.45 / 79.12
PTV3	93.43	67.13 / 80.33	86.41 / 92.71	69.08 / 81.71	61.04 / 75.81
PTV1	93.03	66.65 / 79.99	84.97 / 91.87	70.03 / 82.37	61.43 / 76.11
Point NeXt	92.39	65.78 / 79.36	82.25 / 90.26	75.22 / 85.86	59.37 / 74.51
Point Meta Base	93.56	68.88 / 81.57	85.74 / 92.32	79.44 / 88.54	63.21 / 77.46

Among the models, Swin3D demonstrated the strongest semantic segmentation performance, ranking first across all key metrics. It achieved the highest overall accuracy (94.83%), mean accuracy (77.89%) and mean IoU (72.24%), along with the best per-class IoUs for walls (88.95%), doors (81.37%) and windows (65.45%). This consistent lead indicates its robust capacity to handle both dominant and minor architectural classes. PointMetaBase ranked second, with a notably high door IoU of 79.44% and stable performance across other classes. While it did not surpass Swin3D (Figure 5, Figure 10) in any single metric, its segmentation results were consistently competitive, especially on door and window classes. PTV3 (Figure 4, Figure 9) and PTV1 (Figure 3, Figure 8) showed similar overall accuracies but slightly lower mean IoUs Between the two, PTV3 achieved better performance on wall segmentation (86.41%) but lagged behind in door IoU (69.08%), while PTV1 recorded a marginally higher door IoU (70.03%) despite a lower window score. PointNeXt (Figure 2, Figure 7) ranked lowest overall. However, its door IoU (75.22%) was unexpectedly higher than both PTV1 and PTV3, suggesting that it captured that specific class more reliably despite lower general accuracy. Across all models, walls were the easiest to segment, consistently yielding the highest IoUs (82–89%), followed by doors (69–81%), while windows remained the most challenging, with IoUs ranging between 59% and 65%. This hierarchy of walls > doors > windows in segmentation performance is observed across all five models, regardless of their architectural differences (e.g., voxel-based, attention-based, or MLP-based) (Figure 1 to Figure 10).

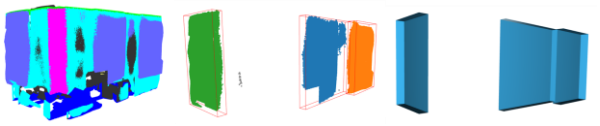

Figure 1: From left to right: the result of semantic segmentation using PointMetaBase, the bounding box results after clustering using DBSCAN and the IFC output for window detection for Office 1 Area 5 of S3DIS Dataset

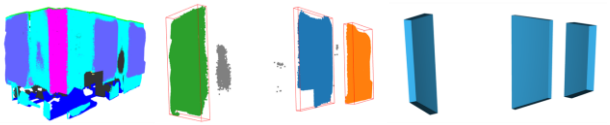

Figure 2: From left to right: the result of semantic segmentation using PointNeXt, the bounding box results after clustering using DBSCAN and the IFC output for window detection for Office 1 Area 5 of S3DIS Dataset

Figure 3: From left to right: the result of semantic segmentation using PTV1, the bounding box results after clustering using DBSCAN and the IFC output for window detection for Office 1 Area 5 of S3DIS Dataset

Figure 4: From left to right: the result of semantic segmentation using PTV3, the bounding box results after clustering using DBSCAN and the IFC output for window detection for Office 1 Area 5 of S3DIS Dataset

Figure 5: From left to right: the result of semantic segmentation using Swin3D, the bounding box results after clustering using DBSCAN and the IFC output for window detection for Office 1 Area 5 of S3DIS Dataset

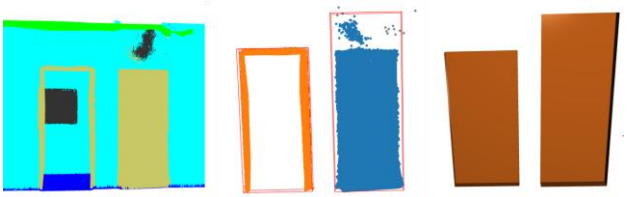

Figure 6: From left to right: the result of semantic segmentation using PointMetaBase, the bounding box results after clustering using DBSCAN and the IFC output for door detection for Office 1 Area 5 of S3DIS Dataset

https://doi.org/10.5194/isprs-annals-XI-1-2026-437-2026

Figure 7: From left to right: the result of semantic segmentation using PointNeXt, the bounding box results after clustering using DBSCAN and the IFC output for door detection for Office 1 Area 5 of S3DIS Dataset

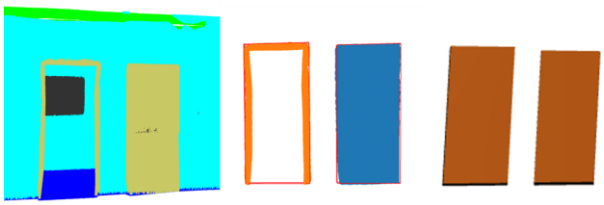

Figure 8: From left to right: the result of semantic segmentation using PTV1, the bounding box results after clustering using DBSCAN and the IFC output for door detection for Office 1 Area 5 of S3DIS Dataset

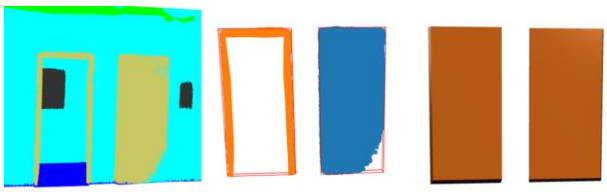

Figure 9: From left to right: the result of semantic segmentation using PTV3, the bounding box results after clustering using DBSCAN and the IFC output for door detection for Office 1 Area 5 of S3DIS Dataset

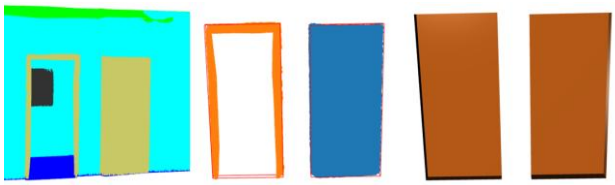

Figure 10: From left to right: the result of semantic segmentation using Swin3D, the bounding box results after clustering using DBSCAN and the IFC output for door detection for Office 1 Area 5 of S3DIS Dataset

3.2 Detection Performance

Instance-level detection performance was evaluated using detection rate (recall), precision, and F1-score for both door and window openings. The results are presented in Table 2 and Table 3.

Table 2: Detection Performance for Doors

Metric Doors	Detection Rate (%)	Precision	F1 Score
Swin3D	96.85	0.8849	0.9248
PTV3	85.04	0.7500	0.7970
PTV1	81.89	0.7939	0.8062
PointNeXt	87.40	0.8538	0.8638
PointMetaBase	92.91	0.8872	0.9077

Table 3: Detection Performance for Windows

Metric Windows	Detection Rate (%)	Precision	F1 Score
Swin3D	75.00	0.9512	0.8387
PTV3	75.00	0.9512	0.8387
PTV1	71.15	0.9024	0.7957
PointNeXt	67.31	0.7447	0.7071
PointMetaBase	67.31	0.8140	0.7368

For door detection, Swin3D achieved the highest detection rate (0.9685), precision (0.8849) and F1-score (0.9248), ranking first across all three metrics. PointMetaBase also performed strongly, ranking second on all door metrics. The others showed lower recall and precision, and consequently F1-score. Overall, Swin3D and PointMetaBase were the most effective at detecting door instances, PointNeXt was middling and PTV1 and PTV3 lagged in terms of completeness and reliability (Figure 6 to Figure 10).

For window detection, both Swin3D and PTV3 achieved the highest F1-score of 0.8387, combining a moderate detection rate (0.75) with a high precision (0.9512). These models correctly identified the majority of window instances while producing very few false positives. PTV1 also performed well while PointMetaBase and PointNeXt had similar detection rates (0.6731) but diverged on precision (Figure 1 to Figure 5).

Across all models, window detection rates were consistently lower than door detection rates, but precision values were generally higher for windows. This suggests that while fewer window instances were detected, the predictions that were made tended to be more accurate and less noisy. Doors, on the other hand, were more accurately detected (higher recall) but with slightly lower precision in some models.

In terms of overall model ranking across both classes, Swin3D consistently performed best, ranking first on all door metrics and tied first for windows. PointMetaBase followed, especially for door detection, while PTV3 ranked higher for windows than for doors. PointNeXt showed strong door recall but lower precision for windows, and PTV1 remained near the middle across most metrics.

3.3 Geometric Performance Results

Geometric performance was evaluated using centroid error, bounding box size error (width and height) and 3D IoU for true positive detections. The results for each model are shown in Table 4 and Table 5.

Table 4: Geometric Accuracy for Doors

Metric Doors	Centroid Error (m)	Width Error (m)	Height Error (m)	3D IoU
Swin3D	0.0256	0.0079	0.0279	0.9473
PTV3	0.0360	0.0134	0.0243	0.9214
PTV1	0.0317	0.0157	0.0322	0.9338
PointNeXt	0.0990	0.0237	0.8638	0.8220
Point MetaBase	0.0433	0.0178	0.9077	0.9148

Table 5: Geometric Accuracy for Windows

Metric Windows	Centroid Error (m)	Width Error (m)	Height Error (m)	3D IoU
Swin3D	0.0764	0.0690	0.1315	0.6957
PTV3	0.0871	0.0499	0.1461	0.7597
PTV1	0.0926	0.0942	0.1479	0.7290
PointNeXt	0.1162	0.0832	0.1477	0.6970
Point MetaBase	0.0813	0.0595	0.1075	0.7505

For doors, Swin3D showed the highest geometric reliability, with the lowest centroid error (0.0256 m), lowest width error (0.0079 m) and highest 3D IoU (0.9473). PTV3 and PTV1 followed closely, with slightly higher centroid and width errors but similarly high IoUs above 0.92. PointMetaBase also performed well, though its height error (0.0588 m) was higher than that of Swin3D and PTV3. PointNeXt recorded the highest door centroid and height errors, and the lowest IoU (0.8220), indicating less precise spatial alignment and bounding accuracy.
For windows, PTV3 achieved the highest 3D IoU (0.7597), indicating the best volume overlap with ground truth. PointMetaBase followed closely while Swin3D had the lowest centroid error for windows (0.0764 m), though its IoU (0.6957) was slightly lower. PointNeXt and PTV1 had higher centroid and dimension errors across all window metrics, with PointNeXt exhibiting the highest width accuracy gap and PTV1 showing the largest overall height error (0.1479 m).
Across all models, centroid and height errors were consistently larger for windows than for doors, suggesting greater variation and uncertainty in vertical placement and spatial extent. Width errors remained smaller on average, with most values falling under 0.1 m.
In terms of geometric ranking, Swin3D performed best for doors, while PTV3 achieved the most accurate window volume fits. PointMetaBase offered balanced accuracy for both classes. PTV1 delivered moderately accurate geometry for both elements, while PointNeXt produced the highest deviations, especially for doors.

3.4 Performance on Matterport3D

Matterport3D is a large-scale dataset comprising predominantly residential indoor environments, whereas S3DIS consists entirely of office and educational spaces. To assess cross-dataset generalizability and validate whether the performance patterns observed on S3DIS hold across different scanning qualities and environmental contexts, the five trained models were applied to 11 rooms selected from Matterport3D without any fine-tuning.
As expected, segmentation accuracy was lower than on the S3DIS Area 5 test set across all models, reflecting the substantial change in scene, point distribution, and architectural style. The magnitude of this drop varied considerably by model. PointMetaBase-XXL achieved the highest overall accuracy (82.07%), while Swin3D and PTV1 recorded comparable mIoU values of 45.81% and 45.86% respectively, making them the strongest performers on this metric. PTV3 and PointNeXt recorded the lowest overall accuracies at 73.22% and 70.68% respectively. Notably, the cross-dataset ranking differs from the in-domain ranking observed on S3DIS, where Swin3D led overall. On Matterport3D, PointMetaBase-XXL achieves the highest overall accuracy, suggesting it generalises more robustly to unseen residential environments. The mDice scores follow the same trend, with PTV1 achieving the highest mDice (60.91%), closely followed by Swin3D (57.71%). Training on a larger and more diverse set of indoor environments, or fine-tuning on a representative subset of Matterport3D rooms, may improve overall accuracy and alter the relative ranking of the models.
The results of one of the rooms, X7HyMhZNoso-26, show that even if the geometry of the doors is preserved during semantic segmentation, we can get a high geometric accuracy (Figure 11). For the provided room, PointNext achieved the best overall performance with a centroid error of 0.040 m, zero width error, a height error of 0.032 m, and a 3D IoU of 0.964, followed by PTV3 with a centroid error of 0.071 m and an IoU of 0.908, and PTV1 with the lowest centroid error of 0.028 m and an IoU of 0.818. Swin3D and PoitMetaBase recorded IoU values of 0.809 and 0.775 with centroid errors of 0.102 m and 0.132 m respectively. Although PointMetaBase, PointNext, PTV3, and PTV1 produced fragmented segmentations, since the door structure is segmented the geometric error is low. These geometric measurements are the ones required for IFC reconstruction rather than complete accurate segmentation, as further evidenced by the Swin3D door result.

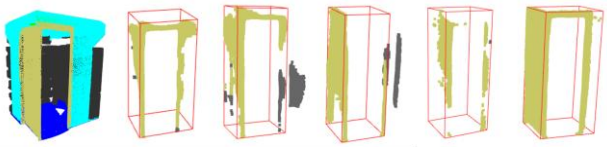

Figure 11: From left to right - Ground Truth followed by outputs using PointMetaBase, PointNext, PointTransformer V1, PointTransformer V3, Swin3D of room X7HyMhZNoso-26

On the other hand, PointNext did not preserve the geometry of the window, with a centroid error of 0.617 m, a height error of 1.557 m, and a near-zero IoU of 0.006, meaning the measurements differed drastically from the ground truth, directly compromising the reliability of the resulting IFC reconstruction (Figure 12). The rest of the models, however, have bounding boxes closer to the expected ground truth even if they were not as accurately segmented as Swin3D, which recorded a centroid error of 0.042 m and an IoU of 0.487, with PTV3 achieving the best IoU of 0.531 and the lowest height error of 0.337 m.

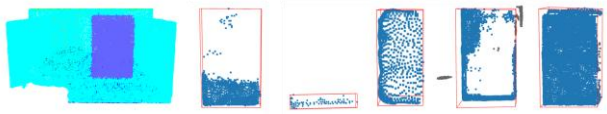

Figure 12: From left to right - Ground Truth followed by outputs using PointMetaBase, PointNext, PointTransformer V1, PointTransformer V3, Swin3D of room 7y3sRwLe3Va-02

Overall, the results from Matterport3D also support our analysis. While higher segmentation accuracy typically improves IFC quality, results show that objects can be reconstructed accurately even with lower mIoU if the overall geometry is preserved, and the focus should remain on reconstruction performance, as models with lower semantic scores can still achieve successful geometric outcomes.

4. Discussion

4.1 Patterns and Trends in Opening Detection

A consistent pattern emerges across all five models in terms of class-wise segmentation and detection performance. Walls are the easiest structural element to segment and detect, with all models achieving high IoU values in the range of 82% to 89%. Their large, flat surfaces and dominant spatial footprint in indoor environments ensure high point density, strong edge continuity, and minimal occlusion. These characteristics facilitate reliable segmentation and bounding box extraction, leading to precise geometric reconstruction.
Doors, while smaller and more varied than walls, are still segmented and detected with relatively high accuracy. IoU values range between 69% and 81%, and detection rates remain consistently above 85% for most models. Their upright orientation, rectangular shape, and position at room entrances make them more distinguishable than other object classes. However, door detection may be affected by partial occlusion by furniture, clutter or incomplete scans of door edges and frames.
In contrast, windows are the most challenging architectural element to segment and detect. Across all models, window IoUs remain in the 59% to 65% range, and window detection rates are

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-XI-1-2026-437-2026 | © Author(s) 2026. CC BY 4.0 License.

significantly lower than those for doors. Several factors contribute to this performance gap: windows often appear co-planar with adjacent walls, their point density is reduced due to reflective or transparent glass surfaces, and they are frequently obscured by curtains, blinds or other interior elements. These attributes reduce the model’s ability to distinguish windows from the surrounding wall geometry.

Importantly, this class-wise performance hierarchy, namely walls > doors > windows, remains stable across all five models, despite substantial differences in architecture. Whether the model employs voxel-based encoding (Swin3D), hierarchical MLP (PointNeXt), meta-architecture framework (PointMetaBase), or point-wise attention (PTV1, PTV3), the relative challenge of each object class is unchanged. This suggests that inherent geometric and contextual properties of each opening class play a more dominant role in segmentation performance than the model architecture itself.

4.2 Influence of Model Architecture

While class-specific trends remain consistent, architectural differences still influence overall performance and per-class ranking of the models.

Swin3D outperformed all other models across segmentation, and the downstream tasks of detection and geometry recognition. Its design combines sparse voxelisation, hierarchical feature aggregation and shifted window attention, enabling it to capture both local structure and global spatial context. The hierarchical attention mechanism in Swin3D efficiently aggregates features across multiple resolutions while maintaining computational efficiency. This appears to contribute directly to its high recall and low centroid error in detecting both doors and windows.

PointMetaBase, the second-best performing model, employs a modular meta-architecture composed of repeatable point-based operations: grouping, neighbour aggregation, point update and positional encoding. Its relatively strong performance, particularly in door detection and window geometry, demonstrates that a lightweight point-based design can achieve competitive results, even without full attention mechanisms or voxel-based encoding.

PTV1 and PTV3 are both transformer-based architectures, however they differ in how they handle local neighbourhoods. PTV1 relies on vector attention and KNN-based neighbour selection, which can be computationally expensive and sensitive to point density variation. PTV3 introduces patch grouping and neighbourhood serialisation using space-filling curves, improving efficiency but possibly limiting context beyond patch boundaries. These models generally perform well but trail Swin3D and PointMetaBase, particularly in window detection and geometry, where fine-grained contextual information may be lacking.

PointNeXt, although the lowest ranked model in terms of segmentation accuracy and 3D IoU, shows unexpectedly competitive performance on door detection. It follows a classic hierarchical MLP-based architecture similar to PointNet++, but with modernised residual MLP blocks and training techniques. Its ability to detect doors despite lower per-point accuracy suggests that even models with simpler feature aggregation can preserve structural grouping at the instance level, enabling successful detection.

In terms of geometric reconstruction, models that incorporate multiscale attention (Swin3D) or explicit positional encoding (PointMetaBase and PTV1) tend to exhibit lower centroid errors and higher 3D IoUs. This may indicate that geometry-aware representations help localise and fit openings more precisely, especially when dealing with occluded or partially scanned surfaces.

Overall, while the relative difficulty of class segmentation is architecture-agnostic, the quality of instance detection and geometric characterisation is affected by how each model encodes spatial relationships. Attention-based and voxel-based models appear to capture complex geometry more robustly, while lightweight MLP-based models benefit from simplicity and generalisation. These differences should inform future decisions about which architectures are best suited for Scan-to-BIM applications, particularly when prioritising semantic accuracy, instance completeness or geometric precision.

5. Conclusion

This study evaluated the capability of five semantic segmentation architectures to support the detection and geometric characterisation of doors and windows in indoor point clouds. A unified pipeline was implemented, comprising semantic filtering, DBSCAN clustering and axis-aligned bounding box fitting, and applied to Area 5 of the S3DIS dataset. Performance was measured across semantic, instance-level and geometric metrics. The results confirm that while doors can be reliably detected and reconstructed across all models, windows remain significantly more challenging due to occlusion, transparency and co-planarity.

Importantly, the findings highlight that semantic segmentation IoU is not always predictive of reconstruction accuracy. Some models with moderate IoU produced accurate bounding boxes, while high-IoU models occasionally failed to preserve geometric consistency. This suggests that the evaluation of segmentation architectures for Scan-to-BIM requires downstream validation beyond point-level metrics.

The bounding boxes generated in this study provide a valid basis for IFC parameterisation, although IFC object creation was not implemented. Future work may extend this pipeline to incorporate wall-aligned filters, orientation estimation and automatic instantiation of IFCDoor and IFCWindow elements.

Overall, the study demonstrates that while semantic segmentation architectures have matured significantly, class-aware post-processing and element-specific evaluation remain essential for robust Scan-to-BIM automation.

References

Adekunle, S.A., Aigbavboa, C., Ejohwomu, O.A., 2022. SCAN TO BIM: a systematic literature review network analysis. IOP Conf. Ser. Mater. Sci. Eng. 1218, 012057. https://doi.org/10.1088/1757-899x/1218/1/012057

Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S., 2016. 3D Semantic Parsing of Large-Scale Indoor Spaces.

Borruso, G., Huang, W., Balletto, G., Kainz, W., Abreu, N., Pinto, A., Matos, A., Pires, M., 2023. Procedural Point Cloud Modelling in Scan-to-BIM and Scan-vs-BIM Applications: A Review. ISPRS International Journal of Geo-Information 2023, Vol. 12, Page 260 12, 260. https://doi.org/10.3390/IJGI12070260

Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M., 2017. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017 2017-January, 2432–2443. https://doi.org/10.1109/CVPR.2017.261

Deng, D., 2020. DBSCAN Clustering Algorithm Based on Density. Proceedings - 2020 7th International Forum on

Electrical Engineering and Automation, IFEEA 2020 949–953. https://doi.org/10.1109/IFEEA51475.2020.00199

Engel, N., Belagiannis, V., Dietmayer, K., 2021. Point transformer. IEEE Access 9, 134826–134840. https://doi.org/10.1109/ACCESS.2021.3116304

Hojjatoleslami, S.A., Kittler, J., 1998. Region growing: A new approach. IEEE Transactions on Image Processing 7, 1079–1084. https://doi.org/10.1109/83.701170

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. Randla-Net: Efficient semantic segmentation of large-scale point clouds. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 11105–11114. https://doi.org/10.1109/CVPR42600.2020.01112

Li, Y., Rui Bu, †⇤, Sun, M., Wu, W., Di, X., Chen, B., 2018. PointCNN: Convolution On X-Transformed Points. Adv. Neural Inf. Process. Syst. 31.

Lin, H., Zheng, X., Li, L., Chao, F., Wang, S., Wang, Y., Tian, Y., Ji, R., 2023. Meta Architecture for Point Cloud Analysis. Institute of Electrical and Electronics Engineers (IEEE), pp. 17682–17691. https://doi.org/10.1109/cvpr52729.2023.01696

Macher, H., Landes, T., Grussenmeyer, P., 2017. From Point Clouds to Building Information Models: 3D Semi-Automatic Reconstruction of Indoors of Existing Buildings. Applied Sciences 2017, Vol. 7, Page 1030 7, 1030. https://doi.org/10.3390/APP7101030

Mahmoud, M., Chen, W., Yang, Y., Li, Y., 2024. Automated BIM generation for large-scale indoor complex environments based on deep learning. Autom. Constr. 162, 105376. https://doi.org/10.1016/J.AUTCON.2024.105376

Maturana, D., Scherer, S., 2015. VoxNet: A 3D Convolutional Neural Network for real-time object recognition. IEEE International Conference on Intelligent Robots and Systems 2015-December, 922–928. https://doi.org/10.1109/IROS.2015.7353481

Patil, J., Kalantari, M., 2025. Automatic Scan-to-BIM—The Impact of Semantic Segmentation Accuracy. Buildings 15, 1126. https://doi.org/10.3390/BUILDINGS15071126

Perez-Perez, Y., Mani Golparvar-Fard, ;, Asce, A.M., El-Rayes, K., Asce, M., 2021. Scan2BIM-NET: Deep Learning Method for Segmentation of Point Clouds for Scan-to-BIM. J. Constr. Eng. Manag. 147, 04021107. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002132

Pexman, K., Lichti, D.D., Dawson, P., 2021. Automated Storey Separation and Door and Window Extraction for Building Models from Complete Laser Scans. Remote Sensing 2021, Vol. 13, Page 3384 13, 3384. https://doi.org/10.3390/RS13173384

Previtali, M., Díaz-Vilariño, L., Scaioni, M., 2018. Indoor Building Reconstruction from Occluded Point Clouds Using Graph-Cut and Ray-Tracing. Applied Sciences 2018, Vol. 8, Page 1529 8, 1529. https://doi.org/10.3390/APP8091529

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. https://doi.org/10.1109/CVPR.2017.16

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: Deep hierarchical feature learning on point sets in a metric space. Adv. Neural Inf. Process. Syst. 2017-December, 5100–5109.

Qian, G., Li, Y., Peng, H., Mai, J., Abed Al Kader Hammoud, H., Elhoseiny, M., Ghanem, B., 2022. PointNeXt: Revisiting PointNet++ with Improved Training and Scaling Strategies. Adv. Neural Inf. Process. Syst. 35, 23192–23204.

Riegler, G., Ulusoy, A.O., Geiger, A., 2017. OctNet: Learning deep 3D representations at high resolutions. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017 2017-January, 6620–6629. https://doi.org/10.1109/CVPR.2017.701

Torr, P.H.S., Zisserman, A., 2000. MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. Computer Vision and Image Understanding 78, 138–156. https://doi.org/10.1006/CVIU.1999.0832

Wang, Y., Sun, Y., Bronstein, M.M., Solomon, J.M., Liu, Z., Sarma, S.E., 2019. Dynamic graph cnn for learning on point clouds. dl.acm.orgY Wang, Y Sun, Z Liu, SE Sarma, MM Bronstein, JM SolomonACM Transactions on Graphics (tog), 2019•dl.acm.org 38, 146. https://doi.org/10.1145/3326362

Wu, X., Jiang, L., Wang, P.-S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H., 2023. Point Transformer V3: Simpler, Faster, Stronger.

Yang, Y.-Q., Guo, Y.-X., Xiong, J.-Y., Liu, Y., Pan, H., Wang, P.-S., Tong, X., Guo, B., 2023. Swin3D: A Pretrained Transformer Backbone for 3D Indoor Scene Understanding.

Zbirovský, S., Nežerka, V., 2025. Open-source automatic pipeline for efficient conversion of large-scale point clouds to IFC format. Autom. Constr. 177, 106303. https://doi.org/10.1016/J.AUTCON.2025.106303