A Comparative Analysis of Four Semantic Segmentation Models for Classification of Building Structural Elements in Highly Occluded Environments

Mojtaba Akhoundi Khezrabad*[1], Davood Shojaei[1], Martin Tomko[2]

[1] Centre for Spatial Data Infrastructures and Land Administration, Department of Infrastructure Engineering, The University of Melbourne, Melbourne, VIC 3053, Australia – makhoundikhe@student.unimelb.edu.au, shojaeid@unimelb.edu.au
[2] Department of Infrastructure Engineering, The University of Melbourne, Melbourne, VIC 3053, Australia – tomkom@unimelb.edu.au

Keywords: BIM, Scan-to-BIM, Semantic Segmentation, Structural Model, Point-Based Networks, Point Cloud.

Abstract

Semantic segmentation of point clouds plays a critical role in automatic Scan-to-BIM workflows. This study evaluates the performance of two Multi-Layer-Perceptron (MLP)-based deep learning networks (PointNet++, PointNeXt-XL) and two transformer-based networks (Point Transformer V1 (PTv 1), Point Transformer V3 (PTv 3)) for the identification of building structural elements in highly occluded environments. The models are trained and tested on three LiDAR datasets of reinforced concrete structures including office buildings and a multi-storey carpark. Results show that transformer-based networks significantly outperform MLP-based architectures under heavy occlusions. Within the transformer-based models, PTv 1 achieved the highest Overall Accuracy (OA) at 92.62% while PTv 3 provided the best balance across all classes, particularly for beam and clutter classes due to its larger receptive field and enhanced geometric encoding. Because segmentation approaches can compensate for errors in ceiling, floor, and column class identification, we recommend PTv 3 for Scan-to-BIM applications focused on structural modelling.

1. Introduction

Structural Building Information Modelling (BIM) has been developed to facilitate collaboration among stakeholders in the Architecture, Engineering, and Construction (AEC) industry. A structural BIM stores a digital representation of the structural elements of a building and their associated information, supporting a range of engineering and management activities including structural analysis (Hasan et al., 2019), structural design (Chi et al., 2015), and structural health monitoring (Wang et al., 2022).

Despite the wide range of capabilities that a structural BIM can provide, many existing buildings still lack such models. Scan-to-BIM is the process of capturing a point cloud using laser scanning or photogrammetry and converting it into a BIM. This process can be employed either to generate a BIM for buildings when no BIM exists or to update outdated BIMs with current as-built information. An essential step within the scan-to-BIM frameworks is the identification of building elements and the classification of point cloud contents according to their semantics. The classification task can be accomplished using rule-based or learning-based methods (See Section 2). Semantic segmentation is a learning-based method that uses machine or deep learning models to assign a label to each individual point, and it is the most widely used method for completing the classification task within scan-to-BIM frameworks. By operating directly on individual points, semantic segmentation facilitates subsequent processing steps (e.g. geometric segmentation) by filtering the points of the features of unwanted classes. Thus, the overall quality of the generated BIM depends on the accuracy of semantic segmentation.

To evaluate the impacts of semantic segmentation on the results of scan-to-BIM, Patil and Kalantari (2025) compared the models generated from five networks, PointNeXt, PointMetaBase, Point Transformer V1 (PTv 1), Point Transformer V3 (PTv 3), and Swin3D and showed that a higher accuracy in semantic segmentation does not necessarily lead to a better 3D model. Their study used a modified Stanford Large-Scale 3D Indoor Spaces (3DIS) (Armeni et al., 2016) dataset with eight classes, including wall, floor, ceiling, beam, column, door, window, and clutter, but the reconstruction analysis was limited to walls. Aksu and Demirel (2024) compared the results of six machine learning models for semantic segmentation of structural elements and related indoor classes, including beam, column, ceiling, floor, wall, door, window, desk, lighting, and other. They handled class imbalance by applying random under-sampling and over-sampling, but their research was mainly concerned with processing time and identifying the best features for the semantic segmentation task.

This study compares four point-cloud deep learning networks, including PointNet++, PointNeXt-XL, Point Transformer V1 (PTv1), and Point Transformer V3 (PTv3), for semantic segmentation of structural elements in highly occluded Scan-to-BIM environments. The aim is not to introduce a new segmentation architecture, but to provide an application-oriented evaluation of existing point-based networks under structural modelling requirements. The selected models allow comparison between hierarchical Multi-Layer-Perceptron (MLP)-based and transformer-based feature-learning strategies. Beyond overall accuracy, the analysis focuses on class-wise performance, confusion between structurally similar elements, and the impact of segmentation errors on downstream structural modelling. This is particularly important for smaller or partially occluded classes, such as beams and columns, whose misclassification can strongly affect the completeness and quality of the generated structural model.

The remainder of this paper is organised as follows. Section 2 reviews related works, and Section 3 describes the datasets, the architecture of the selected models, and the implementation details. Section 4 analyses the results and provides a discussion. Finally, Section 5 concludes the paper by summarising the main findings and offering recommendations.

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-XII-4-W1-2026-1-2026

2. Related Works

2.1 Classification of Structural Elements in Point Clouds

Existing methods for the classification of structural elements can be categorised into rule-based and learning-based techniques.

2.1.1 Rule-based methods: Rule-based methods classify the point clouds or geometric segments into structural elements by considering assumptions about the structural elements. For example, ceilings and floors are horizontal planes while walls, columns, windows, and doors are (roughly) vertical elements. Based on these assumptions, Jung et al. (2018) used the histogram of Z components of the coordinates to identify ceiling and floor points. Fang et al. (2021) classified planar segments into ceiling, floor, and walls, based on the planes normal. Díaz-Vilariño et al. (2015) determined points of vertical columns based on their rectangular or cylindrical shapes. Truong-Hong and Lindenbergh (2022) used vertical continuity of the columns to determine column points. Macher et al. (2017) distinguished walls and clutter elements based on their proximity to the ceilings. Akhoundi Khezrabad et al. (2023) identified doors based on the idea that doors are shorter than ceilings. Jung et al. (2018) distinguished openings from gaps caused by occlusions on walls by analysing the shape of the gaps.
The applicability of the rule-based methods remains limited to situations where their core assumptions are valid, and they cannot address various environments. For example, analysing the Z histogram cannot address ceilings and floors with slopes. Slanted columns and walls may not produce any gaps in the projection of the point clouds on the horizontal plane. Clutter, such as furniture and storage may be high enough to reach the ceiling. Due to occlusions and clutter, the top part of the columns or beams may not be captured within the point clouds, affecting the continuity of the elements. Clutter attached to or close to structural elements reduces the ability of these techniques to classify structural elements properly.

To sum up, the rule-based classification techniques can classify dominant structural classes including floors, ceilings, and walls properly only when their assumptions about the structural elements are valid. However, these methods are not generalizable to every building and shape of structural elements. These methods cannot properly classify smaller structural elements like beams, especially in highly occluded environments.

2.1.2 Learning-based methods: Learning-based methods employ machine and deep learning approaches to classify structural elements. They learn patterns from the data directly, making them adaptable to variations in the shape and geometry of elements. These techniques can be used to classify the elements at the point level, called semantic segmentation or at the object level, which is called object classification. Object classification is mainly conducted using machine learning methods and require a feature engineering step. Bassier et al. (2020) used Random Forest and Support Vector Machine (SVM) for classification of planar segments into structural classes. Perez-Perez et al. (2021) used Markov Random Field (MRF) to enforce coherency between semantic and shape in object classification tasks. Before performing object classification, the point cloud must first be divided into smaller segments, each containing only the points belonging to a specific element. The accuracy of the object classification task, thus, becomes heavily dependent on the quality of the segmentation. Segmentation performance can, however, degrade in the presence of occlusions or clutter—especially when clutter points are attached to the actual elements.

Semantic segmentation, on the other hand, assigns a class label to each individual point. This allows segmentation algorithms to operate only on points from a single class, improving segmentation quality by eliminating irrelevant points. Considering their benefits, many researchers employed semantic segmentation as an important step within the scan-to-BIM process (Mahmoud et al., 2024; Roman et al., 2024; Chen et al., 2023). The networks for 3D semantic segmentation are discussed in section 2.2.

2.2 3D Semantic Segmentation Networks

The most significant challenge for the semantic segmentation of point clouds lies in the scattered and unordered nature of point clouds in 3D space. Based on the strategy employed to manage this challenge, existing models for semantic segmentation of point clouds are divided into projection-based, voxel-based, and point-based methods.

Inspired by the success of 2D Convolutional Neural Networks (CNNs), projection-based methods project 3D point clouds onto 2D image planes and process them using 2D CNNs to extract features. Examples include the works done by Tatarchenko et al. (2018) and Su et al. (2015). However, the projection stage may not consider the sparsity of point clouds upon forming and cause loss of geometric data.

Voxel-based methods convert the irregular nature of the point clouds into a regular structure by 3D voxelization of data. Then, they apply 3D convolutions on the regular voxelised point cloud. Samples of voxel-based methods are provided in VoxNet (Maturana and Scherer, 2015), and OctNet (Riegler et al., 2017). Similar to projection-based models, voxel-based methods may cause loss of geometric details because continuous point coordinates are converted into discrete voxel cells. These models are also computationally expensive and have large memory requirements to process data.

Instead of voxelizing or projecting point clouds that cause loss of geometric data, point-based methods process the point cloud directly. PointNet was the pioneering point-based model and it offered a permutation-invariant model using a symmetric pooling function and point-wise Multi-Layered Perceptrons (MLPs) (Qi et al., 2017a). However, it could not capture local structures. To overcome this, PointNet++ introduced a hierarchical neural network that applied PointNet to overlapping local regions (Qi et al., 2017b). PointNeXt revisited PointNet++ by improving the training strategies and introducing an inverted residual bottleneck design (Qian et al., 2022). In contrast, DGCNN creates a graph from the point clouds using k-Nearest Neighbour (KNN) and applying EdgeConv operation (Wang et al., 2019). Inspired by CNNs, PointCNN and KPConv introduced convolution-like operators for point clouds. While KPConv constructs the kernels based on the points coordinates (Thomas et al., 2019), PointCNN uses X-conv to learn the kernel (Li et al., 2018).

Inspired by the success of transformers in Natural Language Processing (NLP), where self-attention allows each element to selectively focus on other relevant elements in the input Zhao et al. (2021), introduced the point transformer layer to apply this concept to point clouds. In point clouds, self-attention enables each point to assign different weights to neighbouring points based on their geometric and feature relationships. Point transformer V1 includes a point-wise attention layer using relative positional encoding to aggregate features across nearby points. Point Transformer V2 introduced group vector attention, a grouped weight encoding layer, and a partition-based pooling, enabling the model to keep accuracy in large datasets (Wu et al.,

2022). Point Transformer V3 prioritised simplicity and efficiency and removed the requirement for the kNN neighbourhood computation with a serialised neighbour mapping of point clouds that enabled scaling while preserving the functionality and efficiency (Wu et al., 2024).

3. Materials and Methods

This section introduces the datasets, selected models, implementation details, and evaluation metrics used in this study.

3.1 Datasets

Three datasets were used: the public S3DIS benchmark and two real-world datasets captured for this study, namely the Electrical and Electronic Engineering (EEE) building and a multi-storey carpark. S3DIS provides a standard indoor benchmark, while the EEE dataset represents a cluttered multi-storey office/laboratory building and the carpark dataset represents a highly occluded structural environment. Together, these datasets allow the selected models to be evaluated under different levels of clutter, occlusion, and structural-element visibility.

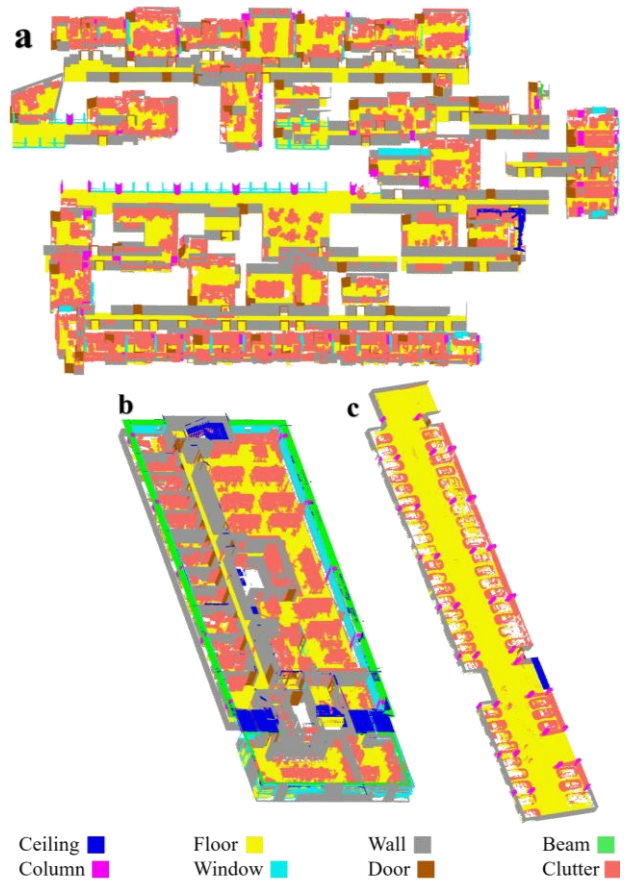

Figure 1: Ground truth for the datasets, (a) S3DIS Area 5, (b) EEE Level 5, (c) Carpark Basement 1 – (Top parts of the point clouds are removed for visualisation)

3.1.1 Stanford Large-Scale 3D Indoor Space Dataset (S3DIS): S3DIS (Armeni et al., 2016) dataset includes six large-scale indoor areas from different buildings that mainly include office areas and educational and exhibition spaces. These are captured using a Matterport Scanner. The dataset includes 13 classes, including ceiling, floor, walls, beam, column, window, door, table, chair, sofa, bookcase, board, and clutter. However, as this research focuses on the structure of buildings, chair, sofa, bookcase and board are merged into the clutter class. In this study, areas 1 to 4, 5, and 6 are used for training, testing, and validation, respectively.

3.1.2 Electrical and Electronic Engineering: This dataset includes five storeys of the EEE building of the University of Melbourne. Both the interior and exterior of the building are captured using a Leica BLK 360 scanner, and the point cloud is manually registered. The dataset mainly includes office spaces, with personal offices, and several laboratories. The exterior part is highly blocked by the trees and vegetation in the vicinity of the building. Although the point clouds for trees are manually removed, this causes gaps and high variations in the density of the points captured from the exterior. The interior of the building includes a high level of clutter, with furniture, boards, ventilation systems and Mechanical, Electrical, Plumbing (MEP) assets attached to beams and ceilings. Level 5 of the building is manually labelled and is divided into 59 tiles, among which 44, 10, and 5 tiles are used for training, testing, and validation, respectively. Other stories of the building are used for visual inspection of the prediction results.

3.1.3 Carpark: This dataset is captured from three storeys of a multi-storey carpark using a BLK2Go laser scanner. The dataset was captured on a busy day (full capacity) of the car park. The parked cars, in addition to MEP, traffic signs, and lighting equipment, caused a high level of occlusions and gaps within the point clouds, particularly on the floor and wall classes. The upper storey is manually labelled and tiled into 57 tiles, among which 39, 12, and 6 tiles are used for training, testing, and validation, respectively. The other two storeys are used for visual inspection of the results. Figure 1 shows the datasets. Table 1 and Table 2 show the distribution of points per class and per patch tile in the mentioned three datasets.

Class	***Dataset***		
	S3DIS	EEE	Carpark
ceiling	51.2 (19.12%)	3.0 (13.31%)	1.6 (26.27%)
floor	43.5 (16.27%)	2.8 (12.58%)	1.7 (28.30%)
wall	75.4 (28.18%)	6.9 (30.59%)	0.5 (8.76%)
beam	3.8 (1.42%)	3.1 (13.85%)	1.0 (15.99%)
column	5.3 (1.97%)	0.2 (0.84%)	0.3 (4.48%)
window	6.9 (2.58%)	1.3 (5.72%)	0.0 (0.00%)
door	13.1 (4.88%)	0.3 (1.21%)	0.0 (0.14%)
clutter	68.5 (25.60%)	5.0 (21.91%)	1.0 (16.06%)

Table 1: Distribution of classes in each dataset. (Number of points in million and their percentage)

Class	***Dataset (Total tiles/patches)***		
	S3DIS (272)	EEE (59)	Carpark (57)
ceiling	272	59	57
floor	272	59	57
wall	272	58	28
beam	106	56	57
column	124	24	23
window	115	43	1
door	256	23	3
clutter	270	59	57

Table 2: Patch/tile-level occurrence of classes in the three datasets

3.2 Model Selection

The selected models are not intended to provide an exhaustive benchmark of all available 3D semantic segmentation networks. Instead, they were chosen to support a controlled comparison between two common point-based architectural families used in

indoor point cloud segmentation: hierarchical MLP-based networks and transformer-based attention networks. PointNet++ and PointNeXt-XL were selected to represent an earlier and a more recent MLP-based architecture, while Point Transformer V1 and Point Transformer V3 were selected to represent an earlier and a more recent transformer-based architecture. This selection allows us to analyse how the transition from local MLP-based feature aggregation to attention-based feature aggregation affects the classification of structural elements under occlusion. The architecture of the selected models is introduced in Sections 0 to 0.

3.2.1 PointNet++: PointNet++ (Qi et al., 2017b) is an MLP point-based model developed to address the main limitation of PointNet in capturing local structures induced by the metric space. The network adopts a hierarchical architecture that recursively applies PointNet as a local feature learner on nested partitions of the point cloud. Its core components are Set Abstraction (SA) layers, which include sampling, grouping, and point-wise feature extraction. These layers allow the network to extract local geometric features and aggregate them into higher-level representations. In addition, PointNet++ uses density-adaptive grouping strategies, such as Multi-Scale Grouping (MSG) and Multi-Resolution Grouping (MRG), to handle non-uniform sampling density in real-world point clouds.

3.2.2 PointNeXt-XL: PointNeXt (Qian et al., 2022) was developed by revisiting the PointNet++ architecture. The authors observed that recent state-of-the-art networks surpassed PointNet++ largely due to improved training strategies, such as data augmentation and optimisation techniques, as well as increased model sizes. This suggested that the potential of the classical PointNet++ architecture had not been fully explored. Architecturally, PointNeXt is built upon the hierarchical SA blocks of PointNet++, but introduces Inverted Residual MLP (InvResMLP) blocks to enable more effective model scaling. These blocks incorporate residual connections, separable MLPs, and an inverted bottleneck design to improve feature extraction. PointNeXt-XL is the largest variant of this family and achieved higher mIoU than PTv1 (74.9% compared to 73.5%) on S3DIS 6-fold cross-validation when released.

3.2.3 Point Transformer V1: PTv1 (Zhao et al., 2021) was developed to investigate the application of self-attention networks to 3D point cloud processing. The architecture is built upon a residual Point Transformer block centred around a vector self-attention layer, which applies attention locally within a kNN-defined neighbourhood, typically using k = 16. The model also incorporates trainable relative position encoding, defined by an MLP, to better use the 3D positional information of neighbouring points. The complete network follows a U-Net design, using Transition Down modules with Farthest Point Sampling (FPS) and Transition Up layers with interpolation and skip connections to process and decode features hierarchically.

3.2.4 Point Transformer V3: PTv3 (Wu et al., 2024) was developed to address the trade-off between accuracy and efficiency, based on the hypothesis that model performance is strongly influenced by scale. Therefore, PTv 3 replaces computationally and memory-expensive operations in previous versions with more efficient alternatives, expanding the effective receptive field from 16 to 1024 points. Its core innovation is Point Cloud Serialisation, which converts unstructured point clouds into a structured order using space-filling curves, such as Z-order and Hilbert curves. This removes the need for inefficient kNN neighbour search. PTv 3 also replaces Relative Positional Encoding (RPE) with Enhanced Conditional Positional Encoding (xCPE), implemented using a sparse convolution layer with a skip connection before the attention layer. Its feature abstraction is then performed through Patch Attention, which groups points into non-overlapping patches along the serialised order.

3.3 Implementation and Results

In addition to the coordinates, (Red – Green - Blue) RGB colours of the points are introduced as features for the model. The implementation of MLP-based models and Transformer-based models were based on OpenPoints (Qian et al., 2022) and Pointcept (Pointcept Contributors, 2023) repositories developed for S3DIS dataset. All the models are trained for 100 epochs on the datasets and for each model the best epoch in terms of Overall Accuracy (OA) is used for prediction. Standard metrics including OA, Intersection over Union (IoU), and mean Intersection over Union (mIoU) as introduced in **Error! Reference source not found.** to **Error! Reference source not found.** are employed to evaluate the performance of the models.

$$OA = \left(\frac{TP+TN}{TP+TN+FP+FN}\right) \times 100 \quad (1)$$

$$IoU_i = \left(\frac{TP_i}{TP_i+FP_i+FN_i}\right) \times 100 \quad (2)$$

$$mIoU_i = \left(\sum_{i=1}^{n} IoU_i\right)/n \quad (3)$$

Where TP, TN, FP and FN are the total numbers of true positive, true negative, false positive, and false negative points respectively; i represents a specific class, and n shows the number of classes. Table 3 reports per dataset OA metric for each model, while Table 4 shows the evaluation metrics reported in the point cloud on the combination of the datasets.

Model	***OA (%)***		
	S3DIS	EEE	Carpark
PointNet++	88.14	93.76	90.16
PointNext-XL	88.94	90.98	96.25
PTv 1	92.48	93.90	98.79
PTv 3	92.19	95.15	98.82

Table 3: Performance of the selected models for semantic segmentation of structural elements on each dataset

Metric (%)	***Model***			
	PointNet++	PointNeXt-XL	PTv1	PTv3
OA	88.29	89.13	**92.62**	92.40
mIoU	62.56	63.96	73.69	**75.19**
Ceiling IoU	88.38	93.01	**93.26**	91.76
Floor IoU	98.15	98.29	**98.57**	98.09
Wall IoU	77.28	77.49	**85.85**	82.88
Beam IoU	44.12	36.45	71.96	**79.76**
Column IoU	28.08	27.89	**39.76**	35.09
Window IoU	25.21	34.2	**63.35**	59.66
Door IoU	61.51	64.59	51.72	**67.58**
Clutter IoU	77.76	79.77	85.08	**86.69**

Table 4: Performance of the selected models for semantic segmentation of structural elements on the combination of datasets

OA and mIoU indicate that Transformer-based networks significantly offer better performance than MLP-based methods. To be exact, MLP-based networks have recorded comparative IoU for ceiling and floor classes only. This highlights that all the networks can reach a high accuracy for dominant structural classes, but for more complex classes, transformer-based networks are required. While PTv 3 reached the highest IoU for beam, door, and clutter classes, the other five classes recorded their highest IoU for the PTv 1. While PTv 1 has reached the

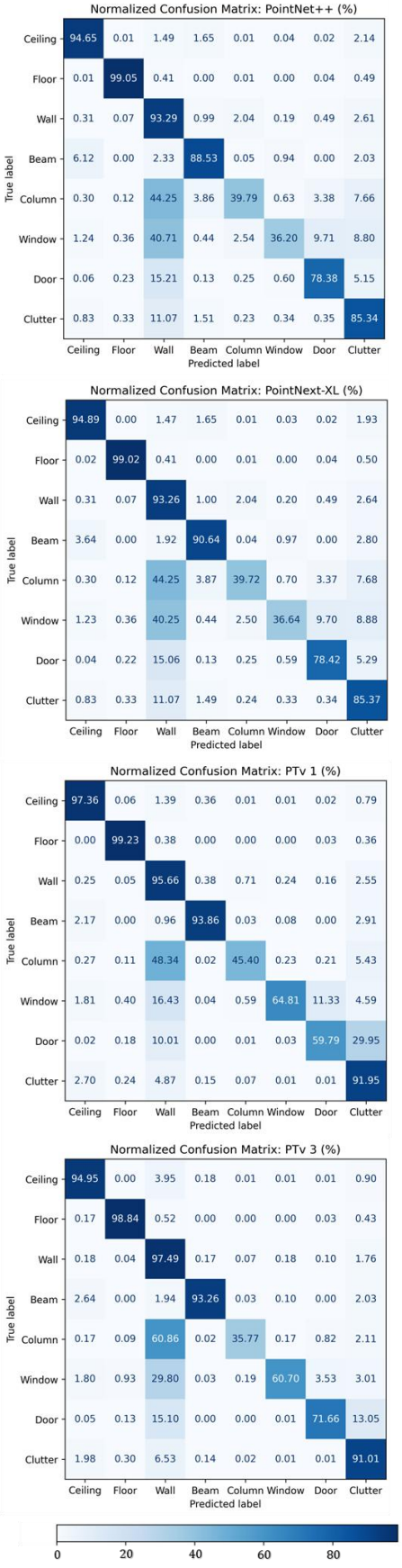

Figure 2: Normalized Confusion matrices for PTv 1 and PTv 3 on the combination of datasets

highest OA, the best mIoU is recorded by PTv 3. This implies that due to an imbalance of data, where dominant structural elements including ceiling, floor, and walls include more than 63% of the whole dataset, OA gets biased towards PTv 1. PTv 3 on the other hand, obtains the highest mIoU, which shows it has a more balanced performance across all classes.

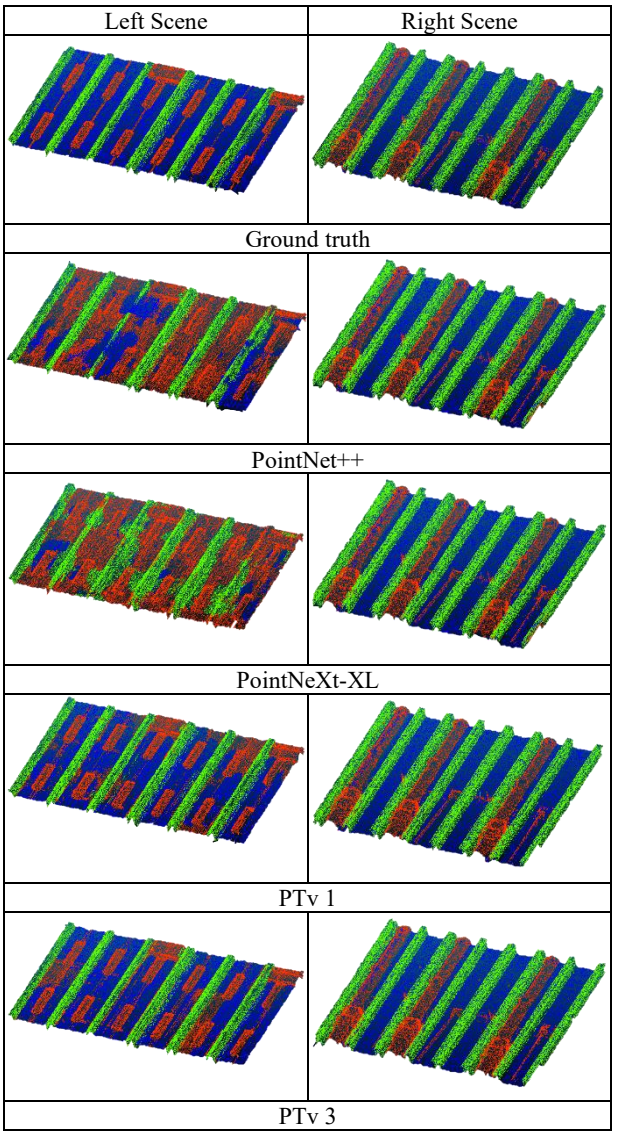

Figure 3: Visual Comparison of the models on EEE dataset (Red: Clutter; Green: Beam; Blue: Ceiling)

The lowest IoU is recorded for the column class in all models, with the best result reaching only 39.76% for PTv 1. To analyse this error in more detail, normalised confusion matrices were used, as shown in Figure 2. The matrices show a strong confusion between columns and walls, where column points are more likely to be classified as walls than assigned to their true class in some models (approximately 48% vs. 45% for PTv 1 and 61% vs. 36% for PTv 3). This suggests that the weak column performance is not only due to the number of column points, but also to the geometric ambiguity between columns and walls. This issue is also reported in scan-to-structural BIM literature, where columns are identified as difficult to classify because they contain fewer points than dominant classes, share similar vertical planar features with walls, and are more affected by occlusion and incomplete scans due to their smaller size (Akhoundi Khezrabad et al., 2026). This ambiguity is especially critical when columns

are attached to walls, have small extrusions from the walls, or have only one visible side. Therefore, improving column classification may require not only increasing the number of column samples through data augmentation, but also using larger contextual information, column-specific geometric constraints, or post-processing strategies to better separate wall and column classes.

In terms of determining clutter, PTv 1 has shown a higher recall by 1%, however, it misclassifies more non-clutter into the clutter class, causing a better IoU for PTv 3 by 86.69% compared to 85.08%. This implies that PTv 3 is a better option for classifying structural elements in highly occluded environments.

Figure 3 shows two scenes within the EEE dataset. In the right scene, all four models have shown a good performance in distinguishing beam, ceiling and clutter classes. However, in the left scene, PointNet++ and PointNeXt-XL could not properly distinguish beams, ceiling, and clutter elements. The main difference between these two scenes is that, in the left scene, the extrusion of the beam elements from the ceiling is lower than in the right scene (~15 cm compared to ~50 cm). As the same trend is seen over the whole dataset, this implies that transformer-based methods can manage smaller details, but MLP-based models struggle with it.

4. Discussion

After semantic segmentation, the next step in the scan-to-BIM framework is the geometric segmentation of point clouds. We argue that available segmentation techniques that are commonly used in structural elements modelling can partly compensate for the errors in the semantic segmentation of the dominant structural elements including wall, ceiling, and floor classes, which can be identified as planar segments with specific orientations. For example, Patil and Kalantari (2025) compared the accuracy of the wall models produced from the results of five semantic segmentation networks. After semantic segmentation, they extracted wall segments using RANSAC, known to be robust against noise and showed that lower accuracy in semantic segmentation may not necessarily cause a reduction in the modelling accuracy.

Model	***Strength***	***Weakness***	***Scan-to-BIM interpretation***
PointNet++	Good ceiling/floor performance	Weak beam and column IoU	Suitable mainly for less occluded scenes
PointNext-XL	Slightly better mIoU than PointNet++	Weak beam/column performance	Limited for highly occluded structural modelling
PTv 1	Highest OA and best column IoU	Lower beam/clutter IoU than PTv3	Strong overall point-wise classifier
PTv 3	Best mIoU, beam and clutter IoU	Column-wall confusion remains	Most suitable for structural Scan-to-BIM

Table 5: Comparative interpretation of model performance for structural Scan-to-BIM applications

In addition, the fact that vertical columns can cause a gap in the projection of the point clouds in 2D that follows a simple shape (circular or rectangular) supports classification of column points in point clouds. Therefore, in the evaluation of the results of semantic segmentation, the performance on beam and clutter should be given priority. Due to its dynamic feature learning capability, PTv 3 discriminates between clutter and structural elements better. Thus, PTv 3 is the most suitable model.

On the other hand, MLP-based models can reach satisfying results in less occluded environments and when there is a sharp difference between the elements. However, they struggle to delineate fine structural boundaries in cluttered areas, as observed in the mixed projections of the left scene in Figure 3. Table 5 synthesises these findings from a structural Scan-to-BIM perspective.

Hence, the findings of our study can be summarised as follows:

- PointNet++ performs reliably in scenes with limited occlusion and where structural elements exhibit clear geometric extrusions. As set abstraction layers aggregate features over fixed-radius and local regions, mixed neighbourhoods caused by clutter or partial occlusions often lead to feature mixing. Limiting the applicability of PointNet++ when analysing small structural components or highly cluttered indoor environments is needed.

- Although PointNeXt-XL improves optimisation, training strategies, and feature extraction compared to PointNet++, it inherits the same fundamental constraint of MLP-based feature aggregation. As a result, its ability to distinguish small structural details remains limited in highly occluded regions.

- PTv1's vector self-attention and relative positional encoding improve the adaptive aggregation of neighbouring features. It achieves the highest OA and column IoU in this study. However, each attention layer operates within a relatively small local kNN neighbourhood, which may provide less direct access to broader contextual information than PTv3 in highly occluded scenes.

- PTv 3 demonstrates balanced performance across structural classes, largely due to its large effective receptive field and serialisation-based neighbourhoods. These characteristics enable it to integrate both local and global geometric information and reduce the impact of partial occlusions. PTv 3 is the most suitable candidate for structural modelling tasks where beams, clutter, and other small-scale elements are partially visible or hard to distinguish.

5. Conclusion

This study provided a comparative analysis of four semantic segmentation networks for classifying structural elements in buildings. The results showed that transformer-based methods offer a noticeably better performance than MLP-based architectures. Since post-processing techniques can effectively handle ceilings, floors, walls, and even columns, we argue that beams and clutter are the classes with the highest reliance on the quality of semantic segmentation. Among the tested models, PTv 3 achieved the best performance for these classes due to its scalability. Thus, we recommend using PTv 3 for Scan-to-BIM applications required for structural modelling. The accuracy for columns remained low across all networks, largely because of their geometric similarity to walls and the imbalance of data between these two classes. Considering this, innovative semantic

segmentation models or post-processing strategies are needed to address the ambiguity between wall and column classes.

6. References

Akhoundi Khezrabad, M., Shojaei, D., and Tomko, M., 2026. Scan to Structural BIM: A Review of Automatic Structural Modelling of Buildings, Journal of Computing in Civil Engineering, "In Press".

Akhoundi Khezrabad, M., Valadan Zoej, M., and Safdarinezhad, A., 2023. A method for detection of doors in building indoor point cloud through multi-layer thresholding and histogram analysis, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 10, 43-48, https://doi.org/10.5194/isprs-annals-X-4-W1-2022-43-2023.

Aksu, K. and Demirel, H., 2024. Extracting Structural Elements From 3D Point Clouds in Indoor Environments via Machine Learning Techniques, IEEE Access, 12, 94461-94476, https://doi.org/10.1109/ACCESS.2024.3423425.

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., and Savarese, S., Year. 3d semantic parsing of large-scale indoor spaces, Proceedings of the IEEE conference on computer vision and pattern recognition, 1534-1543, Published.

Bassier, M., Yousefzadeh, M., and Vergauwen, M., 2020. Comparison of 2D and 3D wall reconstruction algorithms from point cloud data for as-built BIM, Journal of Information Technology in Construction, 25, https://doi.org/10.36680/j.itcon.2020.011.

Chen, J., Liang, Y., Xie, Z., Wang, S., and Xu, Z., 2023. Automated Reconstruction of Existing Building Interior Scene BIMs Using a Feature-Enhanced Point Transformer and an Octree, Applied Sciences, 13, 13239, https://doi.org/10.3390/app132413239.

Chi, H.-L., Wang, X., and Jiao, Y., 2015. BIM-enabled structural design: impacts and future developments in structural modelling, analysis and optimisation processes, Archives of computational methods in engineering, 22, 135-151.

Díaz-Vilariño, L., Conde, B., Lagüela, S., and Lorenzo, H., 2015. Automatic detection and segmentation of columns in as-built buildings from point clouds, Remote Sensing, 7, 15651-15667, https://doi.org/10.3390/rs71115651.

Fang, H., Lafarge, F., Pan, C., and Huang, H., 2021. Floorplan generation from 3D point clouds: A space partitioning approach, ISPRS Journal of Photogrammetry and Remote Sensing, 175, 44-55, https://doi.org/10.1016/j.isprsjprs.2021.02.012.

Hasan, A. M., Torky, A. A., and Rashed, Y. F., 2019. Geometrically accurate structural analysis models in BIM-centered software, Automation in construction, 104, 299-321, https://doi.org/10.1016/j.autcon.2019.04.022.

Jung, J., Stachniss, C., Ju, S., and Heo, J., 2018. Automated 3D volumetric reconstruction of multiple-room building interiors for as-built BIM, Advanced Engineering Informatics, 38, 811-825, https://doi.org/10.1016/j.aei.2018.10.007.

Li, Y., Bu, R., Sun, M., Wu, W., Di, X., and Chen, B., 2018. Pointcnn: Convolution on x-transformed points, Advances in neural information processing systems, 31, https://doi.org/10.48550/arXiv.1801.07791.

Macher, H., Landes, T., and Grussenmeyer, P., 2017. From point clouds to building information models: 3D semi-automatic reconstruction of indoors of existing buildings, Applied Sciences, 7, 1030, https://doi.org/10.3390/app7101030.

Mahmoud, M., Chen, W., Yang, Y., and Li, Y., 2024. Automated BIM generation for large-scale indoor complex environments based on deep learning, Automation in Construction, 162, 105376, https://doi.org/10.1016/j.autcon.2024.105376.

Maturana, D. and Scherer, S., Year. Voxnet: A 3d convolutional neural network for real-time object recognition, 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS), 922-928, https://doi.org/10.1109/IROS.2015.7353481, Published.

Patil, J. and 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., Golparvar-Fard, M., and El-Rayes, K., 2021. Segmentation of point clouds via joint semantic and geometric features for 3D modeling of the built environment, Automation in Construction, 125, 103584, https://doi.org/10.1016/j.autcon.2021.103584.

Pointcept: A Codebase for Point Cloud Perception Research: https://github.com/Pointcept/Pointcept, last

Qi, C. R., Su, H., Mo, K., and Guibas, L. J., Year. Pointnet: Deep learning on point sets for 3d classification and segmentation, Proceedings of the IEEE conference on computer vision and pattern recognition, 652-660, https://doi.org/10.48550/arXiv.1612.00593, Published.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space, Advances in neural information processing systems, 30, https://doi.org/10.48550/arXiv.1706.02413.

Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H., Elhoseiny, M., and Ghanem, B., 2022. Pointnext: Revisiting pointnet++ with improved training and scaling strategies, Advances in neural information processing systems, 35, 23192-23204.

Riegler, G., Osman Ulusoy, A., and Geiger, A., Year. Octnet: Learning deep 3d representations at high resolutions, Proceedings of the IEEE conference on computer vision and pattern recognition, 3577-3586, https://doi.org/10.48550/arXiv.1611.05009, Published.

Roman, O., Bassier, M., De Geyter, S., De Winter, H., Farella, E. M., and Remondino, F., 2024. BIM Module for Deep Learning-driven parametric IFC reconstruction, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLVIII-2/W8-2024, 403-410, https://doi.org/10.5194/isprs-archives-XLVIII-2-W8-2024-403-2024.

Su, H., Maji, S., Kalogerakis, E., and Learned-Miller, E., Year. Multi-view convolutional neural networks for 3d shape recognition, Proceedings of the IEEE international conference on computer vision, 945-953, https://doi.org/10.48550/arXiv.1505.00880, Published.

Tatarchenko, M., Park, J., Koltun, V., and Zhou, Q.-Y., Year. Tangent convolutions for dense prediction in 3d, Proceedings of the IEEE conference on computer vision and pattern recognition, 3887-3896, https://doi.org/10.48550/arXiv.1807.02443, Published.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., and Guibas, L. J., Year. Kpconv: Flexible and deformable convolution for point clouds, Proceedings of the IEEE/CVF international conference on computer vision, 6411-6420, https://doi.org/10.48550/arXiv.1904.08889, Published.

Truong-Hong, L. and Lindenbergh, R., 2022. Extracting structural components of concrete buildings from laser scanning point clouds from construction sites, Advanced Engineering Informatics, 51, 101490.

Wang, J., You, H., Qi, X., and Yang, N., 2022. BIM-based structural health monitoring and early warning for heritage timber structures, Automation in Construction, 144, 104618.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M., 2019. Dynamic graph cnn for learning on point clouds, ACM Transactions on Graphics (tog), 38, 1-12, https://doi.org/10.48550/arXiv.1801.07829.

Wu, X., Lao, Y., Jiang, L., Liu, X., and Zhao, H., 2022. Point transformer v2: Grouped vector attention and partition-based pooling, Advances in Neural Information Processing Systems, 35, 33330-33342, https://doi.org/10.48550/arXiv.2210.05666.

Wu, X., Jiang, L., Wang, P.-S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., and Zhao, H., Year. Point transformer v3: Simpler faster stronger, Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 4840-4851, https://doi.org/10.48550/arXiv.2312.10035, Published.

Zhao, H., Jiang, L., Jia, J., Torr, P. H., and Koltun, V., Year. Point transformer, Proceedings of the IEEE/CVF international conference on computer vision, 16259-16268, https://doi.org/10.48550/arXiv.2012.09164, Published.