# DEEP-LEARNING GUIDED STRUCTURAL OBJECT DETECTION IN LARGE-SCALE, OCCLUDED INDOOR POINT CLOUD DATASETS

Viktor Drobnyi[1], Shuyan Li[1], and Ioannis Brilakis[1]
[1]University of Cambridge, Cambridge, UK

## Abstract

Automatic geometry digitisation of existing buildings remains challenging due to the large scale and heavy clutter of input point clouds. This paper presents a two-stage hybrid method to detect structural objects. The first stage detects areas of interest that are likely to contain an object, while the second stage finds precise objects. The method benefits from data-driven and model-driven approaches to achieve high accuracy for large-scale, highly cluttered and occluded real-world environments. We evaluate our method on the Stanford3D S3DIS dataset to show that the method detects from 83% to 98% of structural objects, such as columns, doors and windows.

## Introduction

This paper is about digitising existing buildings' geometry from Point Cloud Datasets (PCDs), i.e. constructing their geometric Digital Twins (geometric DTs or gDTs). Constructing DTs is important because they improve various processes during the operation, maintenance, and renovation stages of the building lifecycle (Becerik-Gerber et al. 2012; Bryde et al. 2013; Sacks et al. 2018). However, the costs of constructing a gDT of an existing building counteract the benefits of DTs (Hossain and Yeoh 2018).

Digitisation of existing buildings uses PCDs as a reference for as-is geometry because building design models are outdated or unreliable (Mahdjoubi et al. 2015). It remains a time-consuming and manually intensive process. State of Practice commercial software, e.g. EdgeWise3D or Faro as-built, can fit parametric models into point clusters of objects, detect planar/cylindrical surfaces as a basis for object modelling, and select the most appropriate models from a catalogue. Although it serves as a strong support for semi-automatic modelling, the industry still requires a significant amount of manual work, such as modelling undetected objects and ensuring model consistency, among other.

While buildings have numerous object types, only a few of them frequently appear. Studies show that slabs, walls, columns, beams, doors and windows cover 85% of structural objects (Drobnyi et al. 2022). This work aims to automate the detection of these objects for constructing gDTs of existing buildings. In general, automatic object detection methods can be classified in two types, model-driven approaches and data-driven approaches. The former ones yield robust detection and accurate boundaries but remain impractical in large-scale or cluttered indoors environments due to the exploding computational costs. While the latter ones, such as the work of Qian et al. (2022), are scalable and easily extendable to new object types and environments, they have insufficient performance on smaller objects, such as columns, beams, doors and windows.

In this paper, we propose a hybrid method to detect structural objects in large-scale, heavily cluttered indoor PCDs. We formulate our approach in a two-stage framework: first data-driven and second model-driven stages benefit from both of the above-mentioned approaches. The data-driven stage significantly reduces the search space for objects, while the model-driven stage yields a precise object location with crisp, well-defined boundaries. The proposed method can successfully detect objects, such as columns, doors and windows, in real-world PCDs. We conducted experiments on the public dataset S3DIS (Armeni et al. 2016). Experimental results show that our proposed method can achieve promising performance for gDT construction.

## Background

We give an overview of existing research in the area of digitising buildings and PCD deep learning and highlight gaps in knowledge in this section. Object detection methods in PCD for indoor environments can be split into model-driven and data-driven methods. The model-driven methods detect handcrafted features and compose them to form objects. These methods are based on planar/cylindrical surface detection using RANSAC, Hough Transform, or region-growing algorithms. This allows for the detection of walls, ceilings, floors, columns, beams, and pipes in indoor PCDs (Anagnostopoulos et al. 2016; Dimitrov and Golparvar-Fard 2015; Han et al. 2021; Wang et al. 2021). These methods, however, are designed for environments with low occlusion and no clutter and become computationally infeasible on large-scale inputs.

Model-driven automatic detection of doors can be achieved by analysing wall surfaces. It can use empty regions of wall planes for opened doors (Shi et al. 2019) or colour discontinuity for closed walls (Quintana et al. 2018). These, however, remain sensitive to the quality of wall detection methods and occlusions or assume that local colour distribution will be very distinctive for walls and doors.

The space decomposition methods allow splitting the large-scale input into multiple smaller PCDs for better computational efficiency of object detection methods. This usually involves 2D detection on a projected PCD onto an XY plane (Macher et al. 2017; Ochmann et al. 2019; Shi et al. 2019), which remains sensitive to clutter and occlusions and is subject to other constraints, such as known positions of sensor location. Alternatively, methods such as Pan et al. (2021), Tran et al. (2018) can partition spaces directly in 3D, but they are limited to Manhatten-world buildings with limited occlusions.

Data-driven methods infer objects based on previously seen labelled samples. This allows for predicting bounding boxes of objects directly from PCDs using deep neural networks or other learnable classifiers (Armeni et al. 2016; Xu et al. 2021a; b) or object clusters of building elements (Perez-Perez et al. 2021a; b). However, these methods are designed for environments with no clutter or require the input to be split into individual spaces beforehand, yielding better performance than for large-scale cluttered environments (Qian et al. 2022).

Besides, these methods operate with clusters of points of original PCD or their bounding boxes, which only cover a part of object surfaces. However, many objects, such as columns, are only partially visible; therefore, these methods detect only a part of the object by design. Figure 1 shows an example of this phenomenon. A bounding box of the ground truth cluster of points of a column covers roughly one-third of the volume of the column.

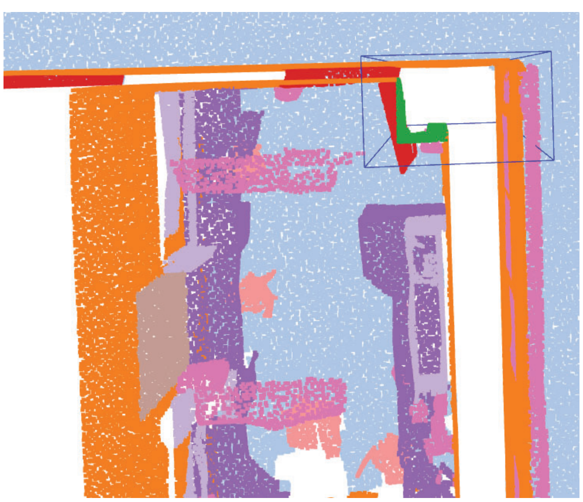

*Figure 1 Ground truth cluster (dark green) and parametric model (blue bounding box) of a rectangular column.*

Many works in the computer vision domain perform class segmentation of points in PCDs (Qian et al. 2022; Thomas et al. 2019; Zhao et al. 2020), which can be helpful for object detection as a first filtering step. While these methods yield high-quality results for large objects, such as floors, ceilings, and walls, their performance on other structural objects is notably smaller. The predicted segmentation fades columns, windows and doors (mainly closed doors) into walls or other classes as they have similar geometry. Also, the boundaries of clusters are usually poorly shaped. Figure 2 shows the predicted semantic class labels for points generated with the PointNext-B model and highlights both of these issues. It limits pure data-driven methods' applicability in detecting building objects.

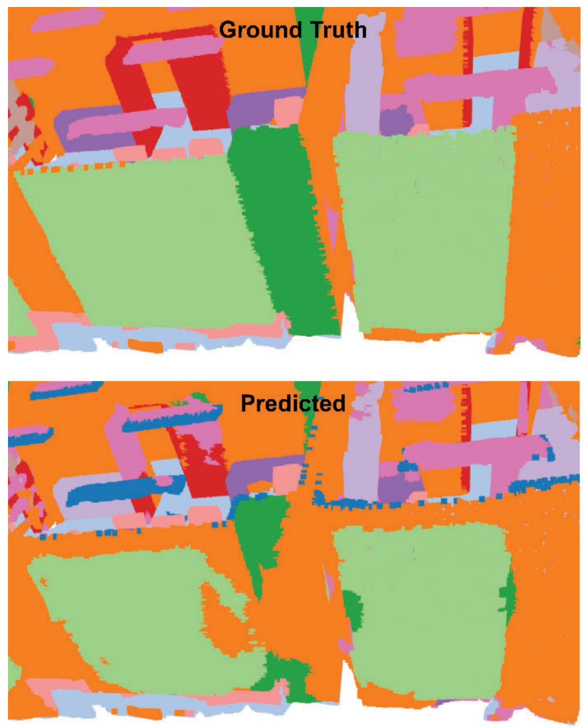


*Figure 2 Example of semantic segmentation. Dark green – columns, light green – windows, red – doors, orange - walls.*

Overall, robust detection of structural objects, excluding walls, ceilings and floors, remains challenging for real-world PCDs. The gap in knowledge is that existing methods for structural object detection rely on space partioning or require limited occlusions instead of directly detecting objects in large-scale inputs with clutter and occlusions. The paper aims to fill this gap. The research questions this work answers:

1. How to effectively find areas that are likely to contain structural objects in large-scale, cluttered PCDs?
2. How to robustly detect objects when a notable part of objects' surface is not visible?

## Proposed Solution

We propose a two-stage hybrid method for detecting rectangular columns, doors and windows in large-scale,

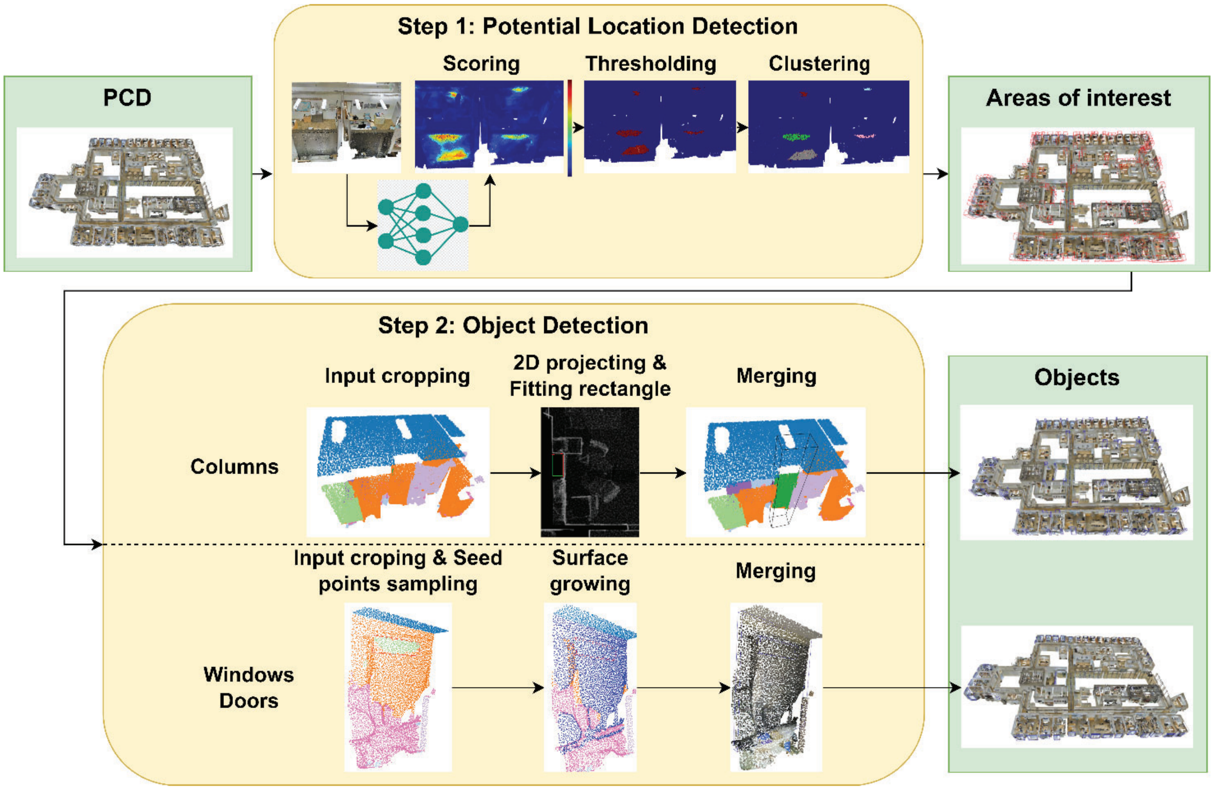

*Figure 3 Method overview.*

occluded PCDs of existing buildings. In contrast to existing studies, this paper uses data-driven methods to detect a rough potential position of structural objects and applies model-driven methods to refine detected objects. The first stage uses deep neural networks to generate areas of interest, inside which there is high likelihood that the object is fully contained. The second stage uses model-driven methods to detect and refine particular object types for each area of interest from the first stage. Figure 3 provides an overview of the proposed solution.

### Areas of interest detection

The first stage receives the entire PCD as input and yields bounding boxes that may contain objects of interest. We use a deep neural network to predict the scores for each point and class of an entire floor of a building. We design a PointNeXt-based model (Qian et al. 2022) and train with semantic class segmentation supervision. We sample multiple overlapping spheres from the input, predict class scores for each point, and average them among samples.

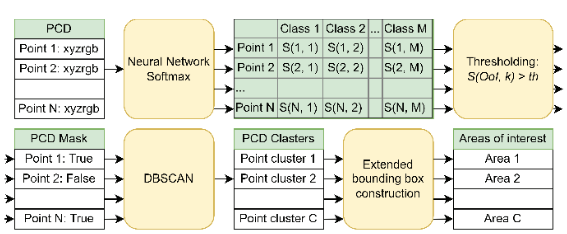

*Figure 4 Overview of the stage one. OoI - Object of Interest (e.g. column or door), th - threshold.*

The neural network predicts probability-like (softmax) values for each point for each object type. Unlike the semantic class segmentation task, which assigns a class label to a point taking a class with the maximum score, we threshold scores of classes of interest (Figure 4). We keep a point if the score of the object type, e.g. column, for the point exceeds the threshold and discard it otherwise. This approach is chosen because points of object types of interest are usually misclassified as other classes which are in the majority of the dataset, e.g. walls. It allows detecting objects with lower likelihood, which frequently happens with columns, doors and windows.

We then cluster the remaining points using the DBSCAN algorithm in 3D space to retrieve seed clusters. After that, we compute extended bounding boxes of these clusters to include potential false-negative detections and capture context. These extended bounding boxes serve as areas of interest for objects for later stages.

### Object detection

The second stage takes every bounding box and fits a model of a particular object type. This stage aims to check if the input bounding box contains an object of interest. We detect an object if the fitted model aligns well with the PCD. Otherwise, we discard the bounding box. We represent the geometry of objects as oriented bounding boxes with one axis parallel to the Z axis.

The implementation of the method varies depending on the object type. Rectangular columns are oriented cuboids inside empty regions of the PCD. We reduce the problem of fitting a column into a 2D problem. We project the point cloud inside a bounding box to the XY plane and find linear segments there. We then find all pairs of close perpendicular segments and check the number of points falling inside the bounding box spanned by these line segments. If there are no points inside the bounding box and the dimensions of this box are within column size (e.g. each side is from 0.2 to 1 metre), we consider it to be a column. Otherwise, we discard it.

After that, we iterate over each pair of bounding boxes to check if they belong to the same column and merge them. We merge the pair if the union (i.e. a minimal bounding box that includes both bounding boxes of a pair) is empty inside and of a reasonable size. This is necessary because each corner spans an object, meaning that multiple objects per single ground truth column may be generated.

We assume that doors and windows have cuboidal shapes with frames around them. Practically, windows and open and closed doors are visible only from one side. Therefore, only a planar surface of an object remains visible. We use point clusters from the first stage as seed points for plane segmentation. We estimate plane parameters for each cluster of areas of interest using RANSAC. We then reclassify points lying on the plane and compute a bounding box of it.

Later, we check the dimension of each proposed object and filter out too-short, too-tall, too-narrow and too-wide objects because these bounding boxes belong to other object types. Lastly, we merge overlapping bounding boxes because a few bounding boxes from the first stage may cover the same object.

The benefits of the proposed method are based on the following assumptions:

- Generated areas of interest cover only a part of the input PCD.
- Objects of interest have a distinctive geometry.

## Experiments

We use the S3DIS dataset (Armeni et al. 2016) for the experiments. We use Areas-1, 2, 3, 4, and 6 for the training of the neural network for the first stage and Area-5 for testing as a standard split. Area-5 covers a floor of a university building, it has 53 rooms and multiple corridors and contains 53 rectangular columns, 41 windows and 76 doors. Area-5 is the dataset's largest, most occluded, and most challenging scene. We also exclude round columns and windows with no points on their surface because the training set has none of these objects. We report the performance only for rectangular columns (R Columns).

We evaluate the performance of both stages. We evaluate the first stage in terms of area coverage and object coverage. The proposed stage splits the input into multiple areas of small area (~5sq.m.) that cover the majority of objects of the original PCD. Figure 5 shows the detected areas of interest for columns.

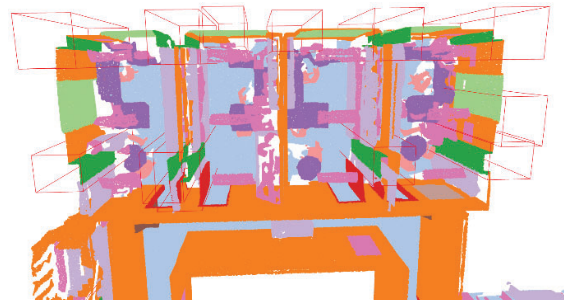

*Figure 5 Detected areas of interest for columns (red bounding boxes). Colours are ground truth class labels (dark green - columns)*

*Table 1 Performance of the first stage. It shows what part of the area is covered by areas of interest (in per cent) and how many objects of interest are inside (in per cent).*

| Object type | R Columns | Windows | Doors |
|---|---|---|---|
| Area covered | 27% | 5% | 11% |
| Objects covered | 92% | 100% | 91% |

Table 1 shows the portion of the original PCD inside of all of the bounding boxes and what part of ground truth objects are inside it. It shows that the first stage can significantly reduce the search space for the second stage, improving the overall method's componential complexity.

False-negative column area of interest detection in the first stage falls into two categories: columns with large dimensions (e.g. around 1 metre) with only one side fully captured and columns with only one side visible and one side occluded by a bookshelf. In both cases, these objects are hardly distinguishable from walls. Besides, the ground truth labels are inconsistent from object to object (e.g. a column surface is marked as a column on one side and as a wall on another), which hardens the training of the neural network. It is worth noting that the method still detects the majority of such columns.

False-negative door detections in the first stage correspond to closed doors aligned with the wall's surface, i.e. lie on the same plane as the wall. This is also because of the small number of such objects in the training dataset.

In the second stage, we model the geometry of an object as a minimal enclosing bounding box of the object. We then compare it to the bounding boxes of ground truth object clusters and register a correct detection if the overlap is at least 50%.

Table 2 shows object-level precisions and recalls for object types in consideration. While the precision for rectangular columns is low, the others precisions and recalls are high. This highlights that the method accurately detects doors and windows while generating

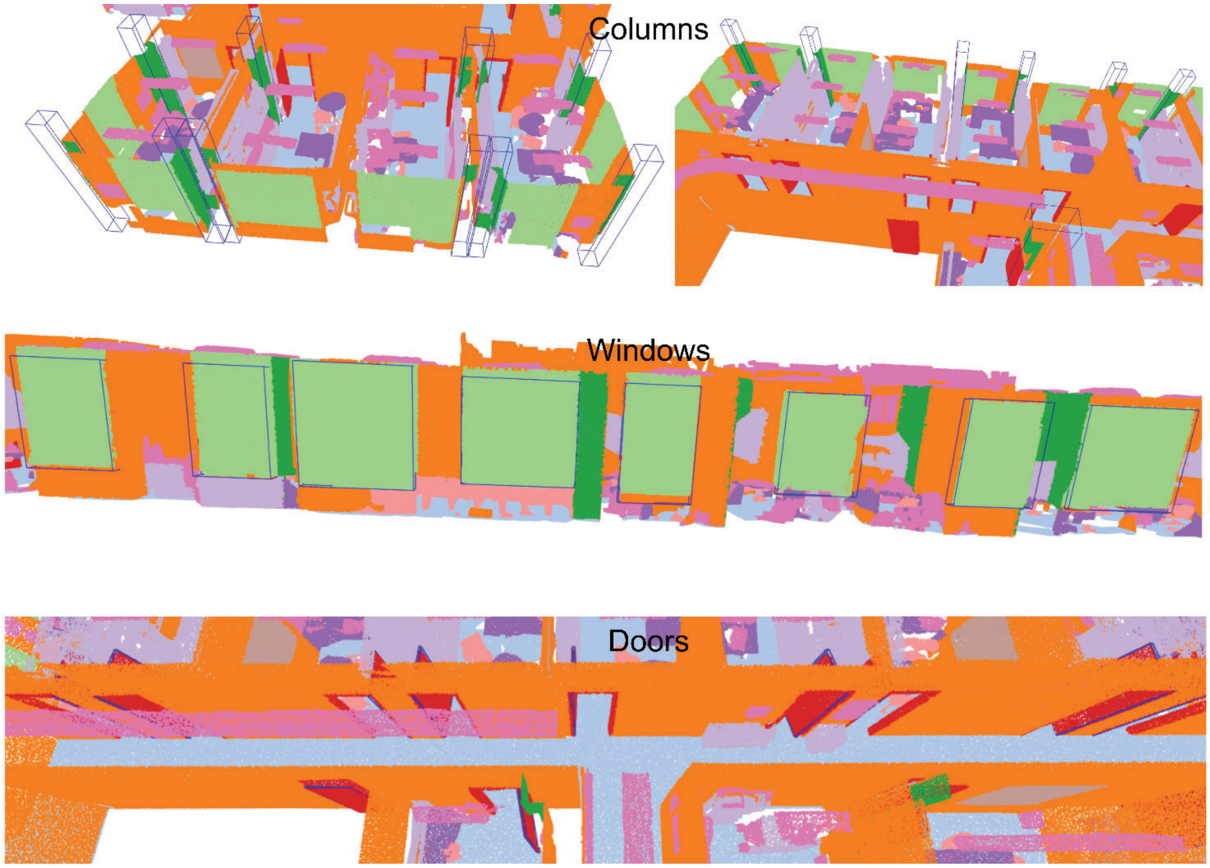


*Figure 6 Examples of detected rectangular columns, windows and doors. PCD colour are ground truth (dark green - columns, light green - windows, red - doors). Blue bounding boxes - detected objects.*

many false positive columns. It indicates that column detection should be improved. Figure 7 shows the results of the second stage of the method for columns. Figure 6 shows the final results of the method.

*Table 2 Object-level precision and recall at overlap threshold 0.5*

| Object type | R Column | Window | Door |
|---|---|---|---|
| Precision | 53% | 95% | 97% |
| Recall | 90% | 98% | 91% |

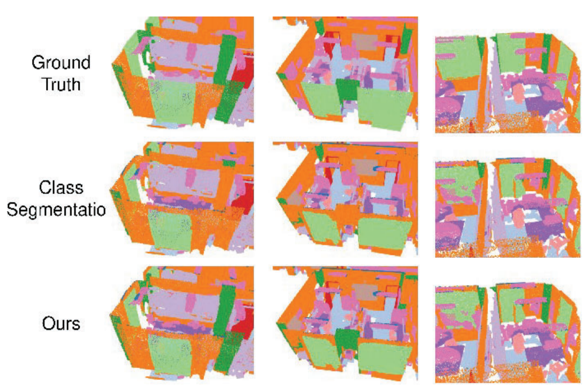


*Figure 7 Example of results for columns. Dark gree - columns.*

The detected columns tend to be larger in volume than bounding boxes of ground truth clusters (Figure 1). This is usually a better result because it aligns better with construction patterns.

The undetected column parts are usually occluded with bookshelves aligned with the column surface, similar to the false negatives in the first stage. The method struggles to detect them because the column-bookshelf border is hardly visible and poorly recognisable in terms of geometry. This leads to columns being unrecognised or partially recognised. Figure 8 highlights this situation; note that only a small part of a column side is visible while the visible side is aligned with a bookshelf.

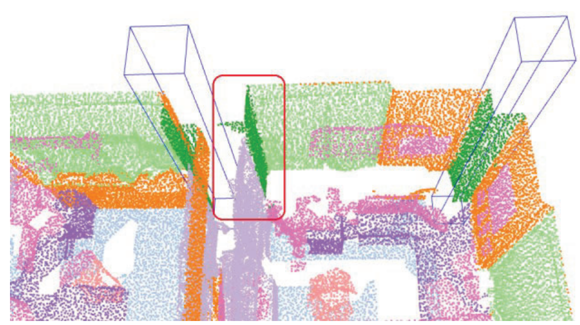

*Figure 8 False negative detection of part of a column (in red).*

The vast majority of false-negative door detection corresponds to closed doors that are on the same plane as the wall they are on. This indicates that closed doors may not have enough descriptive geometry to be detected. Their detection would require colour information in the second stage.

The proposed method can successfully detect rough locations of the majority of rectangular columns in the first stage. This stage is also easily extendable to other object types and environments by providing corresponding labelled PCDs for training. The quality of the outcome for the objects considered in this study can be improved in the same way.

## Discussion

The implemented object detection method finds most objects from the input. Although it produces a substantial number of false positives for rectangular columns due to the similar nature of their geometry to other objects, it can benefit from a better-designed method for detecting/fitting object models (2$^{nd}$ stage). Alternatively, false-positive detections can be easily removed manually in practical applications.

The overall method can robustly detect objects directly in large-scale PCDs, even if objects are heavily occluded. Unlike other methods, our method does not require the input to be split into individual spaces or other objects to be detected, e.g. walls. Additionally, two stages of the method are decoupled and can be used in combination with other segmentation neural networks for better performance of the first stage or model fitting methods for better performance of the second stage. The first stage can also be used as a preprocessing step for other model fitting methods to improve their componential efficiency on large-scale inputs.

## Conclusions and Future Work

The paper presents a two-stage approach to generating objects in large-scale, highly cluttered PCDs of existing buildings. It benefits from the flexibility and extensibility of data-driven approaches in the first stage and the preciseness, robustness and formality of model-driven approaches in the second stage.

The proposed method can highlight the most likely areas for objects and significantly reduce computational costs for further geometry modelling. This can improve practical semi-automatic pipelines and research methods by reducing search space for precise object geometry fitting. It also proposes a set of simple model-fitting methods to refine object proposals and generate precise geometry of objects.

It can contribute towards reducing gDT construction costs for buildings from PCD to the point of their profitability. It will unlock the benefits of DTs for existing buildings and lead to a more cost-effective, sustainable and resilient built environment. This will benefit the overall economy.

We plan to improve both stages to increase the precision and recall for both stages and evaluate the presented approach on other datasets. The future work for the first stage includes investigating other neural network models, a better hyperparameter selection, and other thresholding strategies. For the second stage, we plan to better method for rectangular columns, add other object types, and consider colour values to detect objects with poor-defined geometry.

## Acknowledgement

This work is funded by the EU Horizon 2020 CBIM project under agreement No. 860555 and EU Horizon 2020 BIM2TWIN project under agreement No. 958398.

## References

Anagnostopoulos, I., V. Patraucean, I. Brilakis, and P. Vela. 2016. "Detection of Walls, Floors, and Ceilings in Point Cloud Data." 2302–2311. American Society of Civil Engineers. https://doi.org/10.1061/9780784479827.229.

Armeni, I., O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese. 2016. "3D Semantic Parsing of Large-Scale Indoor Spaces." 1534–1543.

Becerik-Gerber, B., F. Jazizadeh, N. Li, and G. Calis. 2012. "Application Areas and Data Requirements for BIM-Enabled Facilities Management." *J. Constr. Eng. Manag.*, 138 (3): 431–442. American Society of Civil Engineers. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000433.

Bryde, D., M. Broquetas, and J. M. Volm. 2013. "The project benefits of Building Information Modelling (BIM)." *Int. J. Proj. Manag.*, 31 (7): 971–980. https://doi.org/10.1016/j.ijproman.2012.12.001.

Dimitrov, A., and M. Golparvar-Fard. 2015. "Segmentation of building point cloud models including detailed architectural/structural features and MEP systems." *Autom. Constr.*, 51: 32–45. https://doi.org/10.1016/j.autcon.2014.12.015.

Drobnyi, V., Y. Fathy, and I. Brilakis. 2022. "Generating geometric digital twins of buildings: a review." Computing in Construction, 0–0. University of Turin.

Han, J., M. Rong, H. Jiang, H. Liu, and S. Shen. 2021. "Vectorized indoor surface reconstruction from 3D point cloud with multistep 2D optimization." *ISPRS J. Photogramm. Remote Sens.*, 177: 57–74. https://doi.org/10.1016/j.isprsjprs.2021.04.019.

Hossain, M. A., and J. K. W. Yeoh. 2018. "BIM for Existing Buildings: Potential Opportunities and Barriers." *IOP Conf. Ser. Mater. Sci. Eng.*, 371: 012051. IOP Publishing. https://doi.org/10.1088/1757-899X/371/1/012051.

Macher, H., T. Landes, and P. Grussenmeyer. 2017. "From Point Clouds to Building Information Models:

3D Semi-Automatic Reconstruction of Indoors of Existing Buildings.” *Appl. Sci.*, 7 (10): 1030. Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/app7101030.

Mahdjoubi, L., C. A. Brebbia, and R. Laing. 2015. *Building Information Modelling (BIM) in Design, Construction and Operations*. WIT Press.

Ochmann, S., R. Vock, and R. Klein. 2019. “Automatic reconstruction of fully volumetric 3D building models from oriented point clouds.” *ISPRS J. Photogramm. Remote Sens.*, 151: 251–262. https://doi.org/10.1016/j.isprsjprs.2019.03.017.

Pan, Y., A. Braun, A. Borrmann, and I. Brilakis. 2021. “Void-growing: a novel Scan-to-BIM method for manhattan world buildings from point cloud.” Computing in Construction, 312–321. ETH.

Perez-Perez, Y., M. Golparvar-Fard, and K. El-Rayes. 2021a. “Segmentation of point clouds via joint semantic and geometric features for 3D modeling of the built environment.” *Autom. Constr.*, 125: 103584. https://doi.org/10.1016/j.autcon.2021.103584.

Perez-Perez, Y., M. Golparvar-Fard, and K. El-Rayes. 2021b. “Scan2BIM-NET: Deep Learning Method for Segmentation of Point Clouds for Scan-to-BIM.” *J. Constr. Eng. Manag.*, 147 (9): 04021107. American Society of Civil Engineers. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002132.

Qian, G., Y. Li, H. Peng, J. Mai, H. A. A. K. Hammoud, M. Elhoseiny, and B. Ghanem. 2022. “PointNeXt: Revisiting PointNet++ with Improved Training and Scaling Strategies.” arXiv.

Quintana, B., S. A. Prieto, A. Adán, and F. Bosché. 2018. “Door detection in 3D coloured point clouds of indoor environments.” *Autom. Constr.*, 85: 146–166. https://doi.org/10.1016/j.autcon.2017.10.016.

Sacks, R., C. Eastman, G. Lee, and P. Teicholz. 2018. *BIM Handbook: A Guide to Building Information Modeling for Owners, Designers, Engineers, Contractors, and Facility Managers*. John Wiley & Sons.

Shi, W., W. Ahmed, N. Li, W. Fan, H. Xiang, and M. Wang. 2019. “Semantic Geometric Modelling of Unstructured Indoor Point Cloud.” *ISPRS Int. J. Geo-Inf.*, 8 (1): 9. Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/ijgi8010009.

Thomas, H., C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas. 2019. “KPConv: Flexible and Deformable Convolution for Point Clouds.” 6411–6420.

Tran, H., K. Khoshelham, A. Kealy, and L. Díaz Vilariño. 2018. “Shape Grammar Approach to 3D Modeling of Indoor Environments Using Point Clouds.” *J. Comput. Civ. Eng.*, 33. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000800.

Wang, B., C. Yin, H. Luo, J. C. P. Cheng, and Q. Wang. 2021. “Fully automated generation of parametric BIM for MEP scenes based on terrestrial laser scanning data.” *Autom. Constr.*, 125: 103615. https://doi.org/10.1016/j.autcon.2021.103615.

Xu, Y., X. Shen, and S. Lim. 2021a. “CorDet: Corner-Aware 3D Object Detection Networks for Automated Scan-to-BIM.” *J. Comput. Civ. Eng.*, 35 (3): 04021002. American Society of Civil Engineers. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000962.

Xu, Y., X. Shen, S. Lim, and X. Li. 2021b. “Three-Dimensional Object Detection with Deep Neural Networks for Automatic As-Built Reconstruction.” *J. Constr. Eng. Manag.*, 147 (9): 04021098. American Society of Civil Engineers. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002003.

Zhao, H., L. Jiang, J. Jia, P. Torr, and V. Koltun. 2020. “Point Transformer.” *ArXiv201209164 Cs*.