2025 European Conference on Computing in Construction
CIB W78 Conference on IT in Construction
Porto, Portugal
July 14-17, 2025

A HYBRID BOTTOM-UP TOP-DOWN AI METHOD FOR IMPROVED DIGITAL TWINNING OF THE BUILDINGS USING POINT CLOUD DATA AND RGB IMAGES

Mansour Mehranfar[1], Alexander Braun[1], and André Borrmann[1]
[1]Chair of Computing in Civil and Building Engineering, TUM Georg Nemetschek Institute
Technical University of Munich, Germany

Abstract

Digital twins have become transformative tools in design and operations, providing critical capabilities for real-time monitoring and management of building assets. However, creating high-quality digital building models required for the digital twinning of the built environment remains challenging and requires significant human effort. This paper presents a novel AI-based method for automatically creating digital building models from dense laser scanner point clouds and images. The results highlight the performance of the proposed method across multiple case studies, achieving an average accuracy of approximately 7 cm in estimating building parameters, including the location and dimensions of structural and opening elements.

Introduction

Recent advances in information technology, particularly in Artificial Intelligence (AI) and Machine Learning (ML), have opened new opportunities for the Architecture, Engineering, and Construction (AEC) sectors.

One significant application of these technologies is the digitization of built environments by creating digital building models from remote sensing data, such as point clouds and RGB images. These digital models offer valuable capabilities for monitoring and managing building assets, enhancing the accuracy and flexibility of planning and maintenance processes, and improving transparency in decision-making.

Over the last decade, the development of digital twin (DT) concepts and their application across various fields, including industry, construction, and medicine, has led to their integration into diverse research areas (Sacks et al., 2020). Depending on the specific purpose, a DT model can encompass a wide range of information. Given its broad and versatile nature, the concept of a DT can be refined and scaled to provide a more precise definition tailored to its intended use (Brilakis et al., 2019).

In this paper, building digital twinning and the DT concept are defined as a purpose-driven semantic digital representation of building physical assets (Noichl et al., 2024). These representations can be regularly updated by transferring data between physical assets and their digital replicas and include 3D geometric models of building elements, which are further enriched with semantic information and topological relationships.

Background and Related Work

The creation of digital building models is fundamental to the effective implementation of DT for the assets of the built environment. The manual creation of digital building models is labor-intensive and time-consuming, requiring significant human effort and expertise. However, AI technologies offer promising advancements for developing automated digital building model creation methods, significantly reducing time and costs and making it more efficient and scalable (Pan et al., 2023).

Creating digital building models typically consists of two major steps: data capturing and data processing (Noichl et al., 2024). In the data capturing step, raw visual and spatial data, such as point clouds and RGB images, are generated using laser scanning and photogrammetry technologies, efficiently acquiring geometric information of the built environment. Data processing involves interpreting raw point cloud data and images to extract meaningful, readable information for humans and machines, subsequently creating digital models of target objects. This typically involves segmentation, detection, and classification of different building elements to create a structured model through Bottom-Up, Top-Down, and AI-based techniques.

The **Bottom-Up** approach encompasses a range of techniques for segmenting point clouds, each with specific strengths and limitations. Common methods include Region Growing, Model-Based, and Edge-Based techniques, which start from individual data points and build up to form higher-level structures (Dimitrov and Golparvar-Fard, 2015).

In the **Top-Down** approach, the segmentation and classification are guided by predefined relationships among the elements, allowing for the identification and organization of various components based on their spatial and semantic attributes. The Top-Down approach typically requires a substantial amount of prior knowledge about the structure and characteristics of the data, necessitating the establishment of rules and reasoning frameworks to facilitate the segmentation process (Ochmann et al., 2019; Tran and Khoshelham, 2020).

AI-driven approaches significantly enhance the ability to analyze and interpret point cloud data and images using semantic segmentation, object detection, and classification methods (Mehranfar et al., 2024; Pan et al., 2022). In contrast to the often manual and rule-based segmentation processes used in Bottom-Up methods, AI can automate the identification and classification of features within point clouds and images (Tang et al., 2022). Despite their advantages, AI-based approaches require large annotated datasets for training, which can be challenging and time-consuming to obtain. Furthermore, AI methods may struggle with certain edge cases or rare scenarios that are underrepresented in training data, potentially leading to inaccuracies in segmentation or classification (Pan et al., 2022). Thus, the sole reliance on AI-based methods for data processing can disrupt semantic understanding and interpretation of topological relationships between elements, ultimately hindering the creation of high-quality digital building models with rich semantic detail and coherent geometry.

This research aims to introduce a novel AI-based method for improved digital twinning of buildings. The goal is to leverage the strengths of all mentioned data processing approaches by combining domain-specific knowledge in building design and construction with the capabilities of AI in scene understanding. This integration enables the creation of high-quality digital building models with consistent geometry and rich semantics, using point cloud data and RGB images.

Methodology

As shown in Figure 1, the proposed workflow for the automatic creation of digital building models, according to the schematic design at Level of Detail (LOD 200), consists of four major steps: (1) preprocessing, (2) semantic labeling, (3) creation of parametric building model, and (4) door and window detection. The following subsections provide details for each step.

Preprocessing

Effective preprocessing of input point cloud data is essential for optimizing subsequent analyses and computational efficiency. Key steps in preprocessing include subsampling the point cloud to improve processing speed and filtering out noise and clutters.

Subsampling reduces the number of 3D points in the original point cloud data to lower computational complexity while preserving essential geometrical and structural information. In this regard, random sampling and grid-based sampling are common techniques. The Random point cloud subsampling selects points at random, which is efficient, but may lead to uneven distribution and potential loss of critical features. Grid-based subsampling divides the point cloud into equal-sized cells with a dimension of d, selecting one point per cell to maintain uniformity and preserve fine details. Therefore, grid-based subsampling is chosen to enhance processing efficiency and ensure effectiveness.

Noise and outliers are outer points that deviate from the primary data distribution, commonly resulting from sources such as sensor inaccuracies, environmental variability, or transient objects. Noise and outliers points can be effectively removed using statistical or neighborhood-based analysis techniques. Among these, Connected Components Segmentation (CCS) is particularly effective; it applies a predefined distance threshold to identify clusters of connected points, treating each cluster as a distinct segment (Trevor et al., 2013). In indoor environments, noise and outliers typically appear as small, randomly fragmented segments, which facilitates their detection and removal from the dataset.

Semantic Labeling

Semantic labeling refers to assigning meaningful labels to points at various levels of contextual information, making the data more interpretable for processing by humans and machines.

Point Labeling

In this study, the pre-trained PointTransformer semantic segmentation model is employed to assign semantic labels to data at the point level (Zhao et al., 2021) (Figure). The PointTransformer network for semantic segmentation utilizes self-attention in combination with simple linear layers and a multi-layer perceptron (MLP) (Zhao et al., 2021). This model is specifically trained on the Stanford 3D Indoor Dataset (S3DIC), a benchmark dataset for 3D semantic understanding of indoor environment, which includes thirteen object classes such as *Wall, Door, Window, Ceiling, Clutter, and Furniture* (Armeni et al., 2016). The PointTransformer model was trained using data from areas 1 through 4 and area 6, and its performance was subsequently evaluated by conducting semantic segmentation on the unseen data from area 5. The hyperparameter settings used to train the PointTransformer model are detailed in Table 1. These values were selected based on existing literature recommendations and determined by fine-tuning within specified ranges.

Table 1: Hyperparameter settings for the Pre-trained PointTransformer semantic segmentation model.

Parameter	Value
Model Configuration:	
Input feature channels	6
Voxel size (m)	0.04
Maximum number of voxels	50000
Number of points per batch	40960
Maximum training epochs	512
Optimizer Settings:	
Initial learning rate	0.01
Momentum	0.9
Weight decay	0.0001

Walls, floors, and ceilings constitute the primary structural components of buildings, serving crucial functions in shaping both the structural skeleton and the spatial layout. Therefore, the primary objective of this step is to ac-

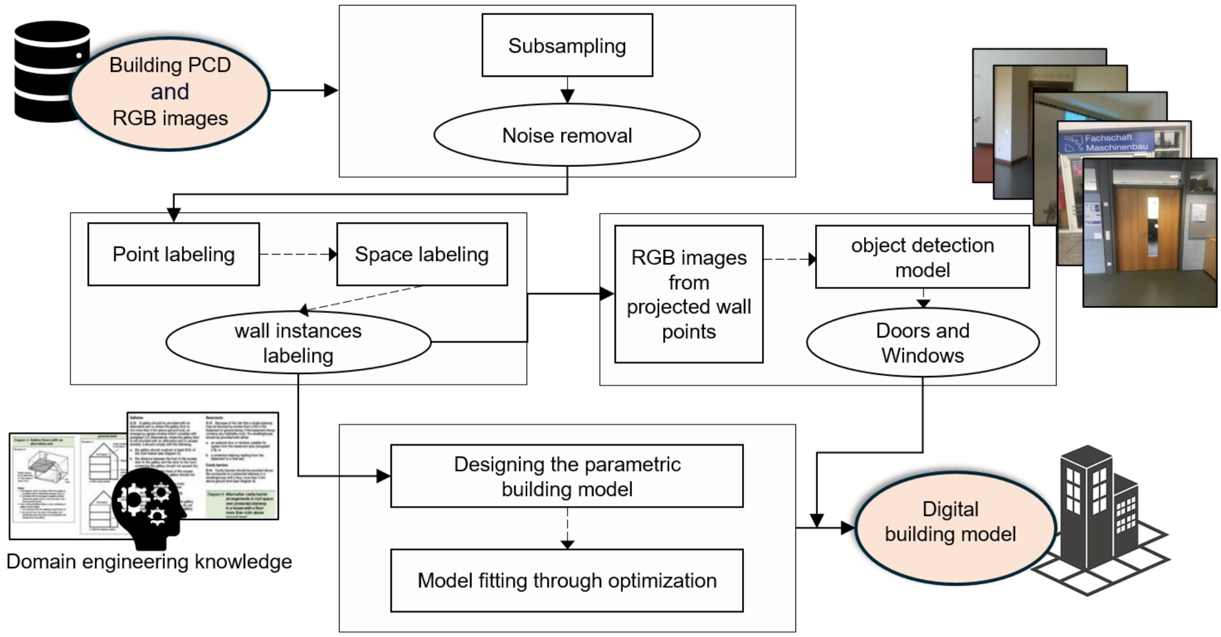
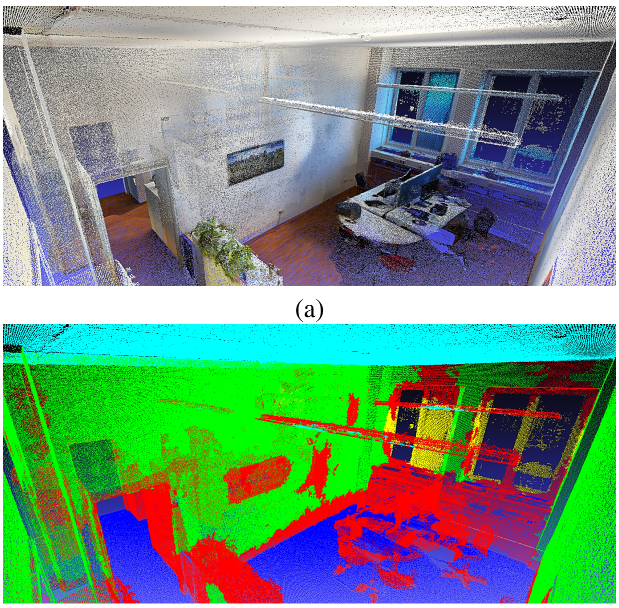


Figure 1: The proposed pipeline for the automatic creation of digital building models using laser scanner point cloud and RGB images.

curately separate points for Ceiling, Floor, and Wall elements. In this regard, the PointTransformer model demonstrates an average accuracy of approximately 93% in the semantic segmentation of the main structural elements, highlighting its effectiveness in identifying them within a complex and cluttered building point cloud.

Table 2: Semantic segmentation results on the S3DIS dataset, evaluated on Area 5 (Zhao et al., 2021).

PointTrasnformer:	
OA	90.8
mAcc	76.5
mIoU	70.4
Accuracy:	
ceiling	94.0
floor	98.5
wall	86.3
door	74.3
window	63.4
others	62.4



(a)

(b)

floor ceiling wall door window others

Figure 2: The result of point labeling using PointTransformer semantic segmentation model; (a) original point cloud of TUM Building 1 dataset, (b) the result of the point semantic labeling.

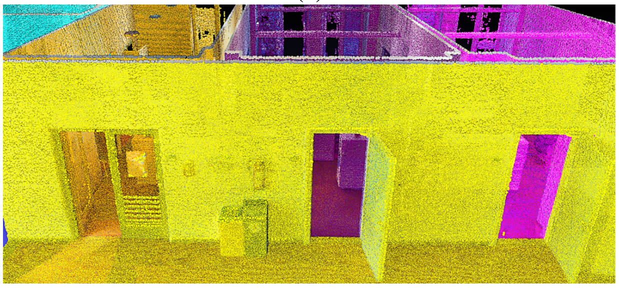
Space Labeling

Segmentation and geometric representation of individual rooms and spaces are essential for developing an accurate and detailed virtual model of a building layout and serve as a key reference to support effective planning and to optimize the functional use of architectural spaces.

A room or space is an enclosed area within a building, separated from other areas by walls and openings. Each space is characterized by attributes such as size, shape, and other distinctive features unique to it. This research employs a knowledge-based bottom-up approach to automatically label the data at the space level and segment individual wall instances within the point cloud.

For 3D space labeling, ceiling points within a *th* distance from the wall segment are excluded from the ceiling segment. This results in the remaining ceiling points being isolated into scattered segments farther from the walls. Next, the Density-Based Spatial Clustering algorithm (DBSCAN) is used to segment the scattered ceiling points into distinct groups (Czerniawski et al., 2018). Finally, the nearest neighbor method is employed to assign the potential segment labels to each 3D point in the point cloud (Figure 3). The value of *th* is selected according to the average wall thickness typically observed in the particular type of buildings being studied.

(a)



(b)

space (1) space (2) space (3) space (4) space (5)

Figure 3: 3D space parsing of the indoor point cloud; (a) the raw point cloud of the TUM Building 1 dataset, (b) the result of the point labeling at the space level.

Wall Instances Labeling

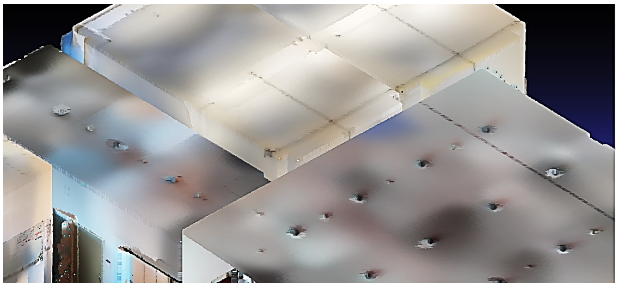
The complexity of indoor environments, combined with noise, clutter, and geometric similarities between walls and furniture, may lead to inaccuracies in labeling wall points and subsequently separating individual wall instances.

The proposed pipeline employs a bottom-up approach to identify wall footprints within each enclosed space and segment 3D wall instances. Since ceiling and wall elements share boundaries within an enclosed space, the boundary points of the ceiling can be utilized to extract wall footprints on the 2D X-Y plane. In cases involving intersecting walls, variations in the principal component analysis (PCA) parameters of wall segments can detect breakpoints or sudden changes, which correspond to endpoints of each wall instance and indicate changes in their orientations.

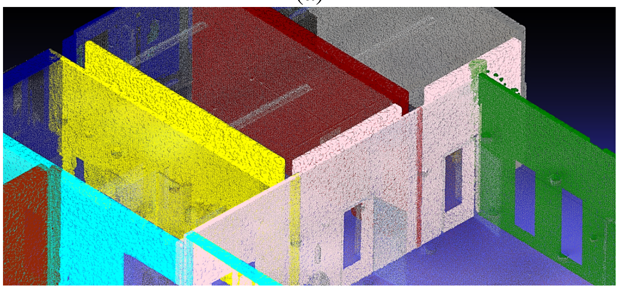
This approach begins by applying the Mean Shift and Alpha-shape algorithms to extract the boundary points of the ceiling for each enclosed space. Next, the PCA coefficient values are computed for each boundary point p by analyzing its k neighboring points and calculating the covariance matrix c as defined in Equation (1):

$$c = \frac{1}{k}\sum_{i=1}^{n}(p_i - \bar{p}).(p_i - \bar{p})^T \qquad (1)$$

where k denotes the number of neighboring points, p_i and p represent the coordinates of the boundary points under consideration.



(a)



(b)

wall (1) wall (2) wall (3) wall (4) wall (5) wall (6)

Figure 4: 3D wall instance segmentation; (a) the original point cloud of the TUM Building 1 dataset, (b) the result of the point labeling at the wall instance level.

The boundary points are then classified into distinct groups according to their orientation, as determined by their PCA coefficient values. These orientation-based groups are categorized as vertical, horizontal, or inclined. To further cluster boundary points belonging to the same wall instance, the DBSCAN clustering algorithm is applied. Finally, the 3D points corresponding to each wall instance are extracted from the original point cloud by applying a buffer b around each identified wall segment (Figure 4). This approach enables effective separation and grouping of wall instances based on their orientation.

Creation of parametric building model

To create the digital building model, data obtained from the 3D space parsing and wall instance extraction steps are integrated using a bottom-up approach. In this context, the initial floor plan mask is generated through a plane-plane intersection method, which is then extruded to form a volumetric representation (Figure 5). However, due to errors in the plane-fitting process and the presence of noise and clutter, the resulting model may exhibit inaccuracies in wall locations as well as width and height estimations.

To address this challenge, a top-down approach is used. Initially, domain-specific knowledge from BIM and architectural design is employed to formulate a set of geometric and mathematical rules and constraints, which are implemented as internal relationships among system components. These rules formulate the interactions between elements (such as walls and slabs) and define both their degrees of freedom and the allowable parameter variations.

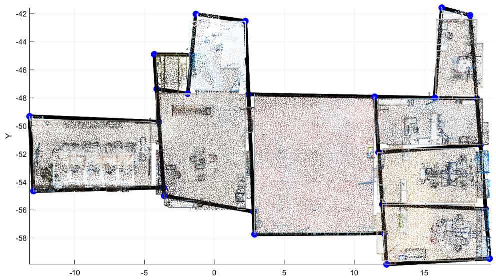


Figure 5: The generated floor plan mask using the bottom-up plane-plane intersection method.

In the designed parametric digital building model, each wall instance is assigned one of three possible orientations: horizontal, vertical, or inclined and defined rules and constraints further regulate the relationships between each wall instance and its connecting walls.

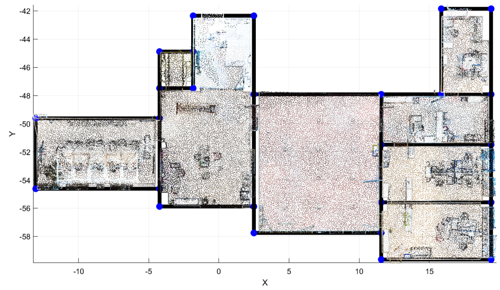
Although the parametric model preserves a consistent topology, it may lack geometric accuracy in terms of element dimensions and their spatial positions within the environment. To enhance the model precision, the entire digital parametric model is fitted to the point cloud data using the Particle Swarm Optimization (PSO) algorithm (Kennedy and Eberhart, 1995). This optimization process extracts the optimal values for the model parameters (Figure 6b). During model fitting process, each wall is modeled as a box with adjustable dimensions (eg., width, length, and height) to reflect the observed geometry.

In the global optimization problem, the average Points-to-Model distance is employed as the overall objective function to achieve geometrically accurate models by optimizing both the dimensions and positions of building elements. This process entails computing the distances between point cloud data and the surfaces of the digital model. Although the use of global optimization in parametric model fitting may introduce increased computational complexity, it ensures topological and geometrical consistency across the entire model reconstruction workflow. Figure 6c shows the optimal objective function values across successive iterations during the model fitting process.

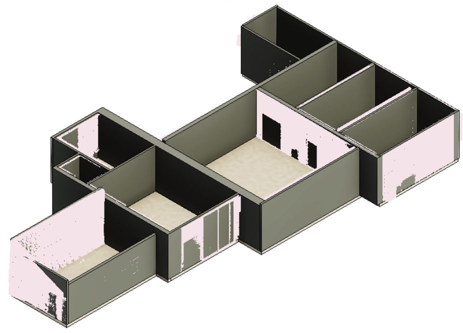
Door and window detection

Although the PointTransformer model demonstrates strong performance in the semantic segmentation of main structural elements, the segmentation results for door and window elements remain suboptimal, often affected by noise and issues of over-segmentation. This issue arises from factors such as sparse points caused by glass reflections and the similarity of these elements geometrical and spectral features to other structural components and results in an inaccurate estimation of the location and dimensional parameters of door and window instances within the built environment.

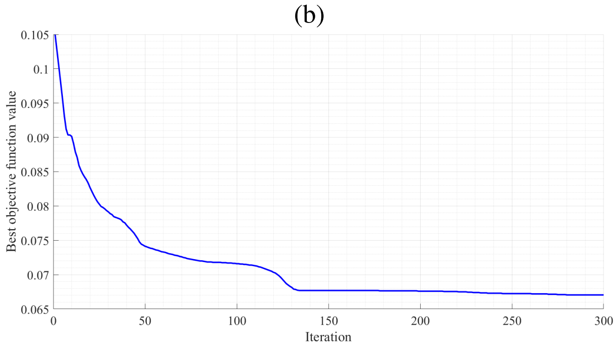
This research proposes a novel approach for improving the detection and modeling of door and window elements in indoor environments using YOLOv8 object detection network (Jocher et al., 2023).



(a)



(b)



(c)

Figure 6: The result of fitting parameterized digital building model to point cloud data using PSO; (a) the floor plan mask after optimization, (b) digital building model with optimum parameter values, (c) the trend of best objective function values over iterations during the optimization.

In this process, the points belonging to one of the two surfaces of each wall, located within a 1 meter distance from the wall instance, are first projected onto the X-Z and Y-Z planes. A gridding and sampling method with a defined resolution d is then applied to convert the point cloud into an RGB image. In this regard, the pixel values are calculated by averaging the RGB values of the points within each corresponding grid cell.

A comprehensive image dataset comprising two distinct sub-categories is compiled to train the YOLOv8 object detection network for the detection of doors and windows (Figure 7). The first subset comprises 214 RGB images captured from various buildings at the Technical University of Munich (TUM), depicting doors and windows in diverse conditions—including open, semi-open, and closed states—and exhibiting different materials and appearances. To enhance the diversity of the training data, an additional 89 images are included, generated from the projection of wall points in the TUM point cloud datasets. Bounding boxes for door and window elements were meticulously annotated within the collected images. The annotated dataset was subsequently divided into training and validation subsets using an 80:20 ratio. The object detection model was then trained using the hyperparameters outlined in Table 3.

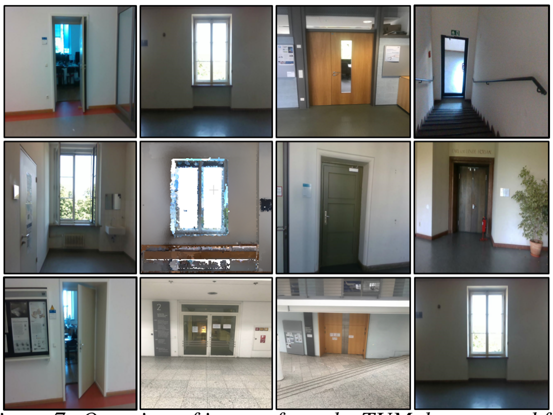


Figure 7: Overview of images from the TUM dataset used for training the door and window detection network.

Table 3: Hyperparameter values used for training the YOLOv8 object detection network.

Parameter	Value
Image size	640
Batch size	8
Training iterations	150
Initial learning rate	0.001
Optimization algorithm	Adam

To evaluate the network's performance in the training phase, the annotated element instances in the images are compared with the detected instances using standard metrics Precision, Recall, and mean average precision (mAP), through Equations 2-4:

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$mAP = \frac{1}{classes} \Sigma_{c \in classes} \frac{|TP_c|}{|FP_c|+|TP_c|} \quad (4)$$

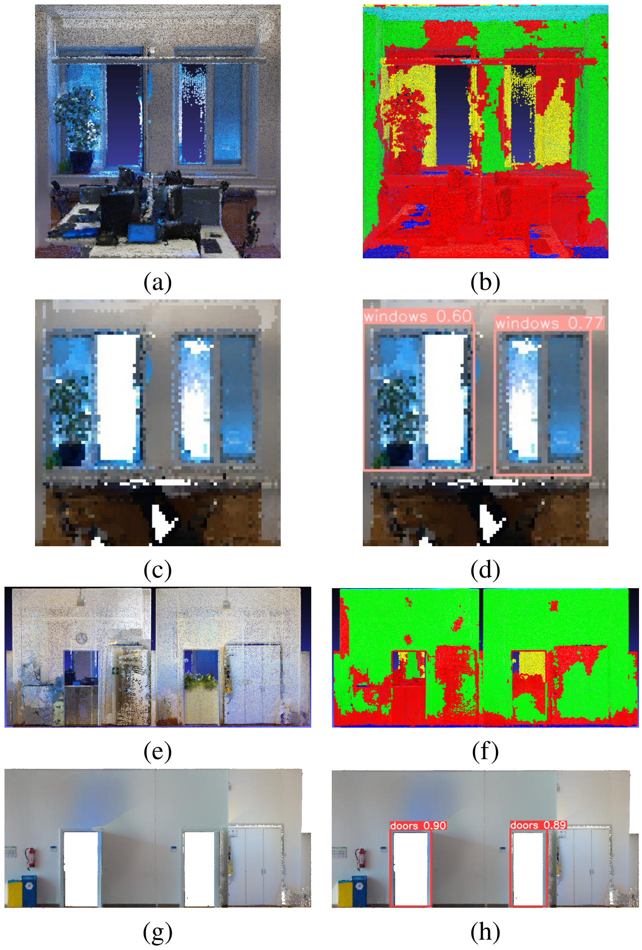
Where TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively. In this regard, the mAP of about 95% in the learning process highlights the effectiveness of the utilized object detection network in the detection of door and window elements (Tables 4-5).

Creation of digital door and window model

Following the detection of door and window instances in the images, their bounding box coordinates are mapped back from the 2D image space to the 3D coordinates of

Table 4: Accuracy evaluation of trained network for door and window detection, mAP (Mean average precision at IoU thresholds of 50 and 50-95).

Class	Precision	Recall	mAP(50)	mAP(50-95)
Doors	0.94	0.86	0.95	0.77
Windows	0.93	1.00	0.95	0.69
All	0.94	0.93	0.95	0.73



floor ceiling wall door window others

Table 5: The result of door and window detection on the projected wall points images; (a) and (e) original point cloud, (b) and (f) the result of the point cloud semantic segmentation using PointTransformer model, (c) and (g) the projected wall instance points, (d) and (h) the result of door and window detection using YOLOv8 object detection model.

the corresponding wall point cloud through reverse projection. However, inaccuracies in projecting wall points onto images and in precisely detecting door and window pixels with the object detection network may result in errors when estimating the parameters of these elements.

A comprehensive library of door and window families, specifically designed for BIM applications, is compiled to improve accuracy and ensure the consistency of element representation. For each detected door and window instance, the closest matching model from the library is selected based on dimensional parameters. This selected model is then used to refine the detected elements by replacing their initial primitive dimensions with precise parameters (Figure 8).

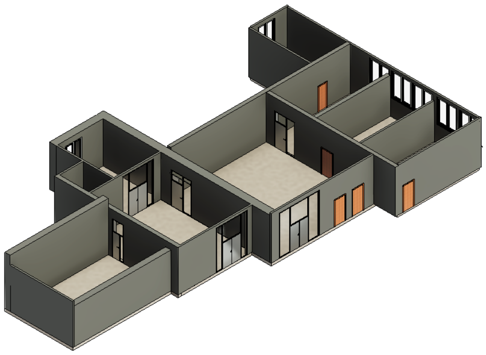


Figure 8: The result parametric digital building model incorporating detected door and window representation.

Result

Case study

To evaluate the performance of the proposed method, four distinct indoor point cloud datasets from different buildings on the TUM city campus (https://doi.org/10.14459/2024mp1742891) and the NavVis company office are used (https://www.navvis.com). These datasets encompass both Manhattan and non-Manhattan building designs and primarily serve educational and research purposes. They include a variety of spaces such as offices, libraries, meeting rooms, and hallways.

Implementation

The proposed pipeline is implemented in Python and tested individually on each point cloud dataset. Table 6 summarizes the key parameters employed in creating the digital building models. These parameter values were selected through fine-tuning within a defined experimental range and are expected to be applicable to other building datasets in real-world scenarios.

Table 6: Overview of key parameter values used in the creation of digital building models.

Parameter	Value
Preprocessing	
1. Grid-based subsampling distance	0.05 m
2. Distance threshold for noise removal	0.25 m
Semantic labeling	
3. Ceiling-wall distance threshold (space parsing)	0.30 m
4. Maximum distance for DBSCAN clustering	0.30 m
5. Number of nearest neighbors for PCA calculation	50
Door and window detection	
6. Grid size for point-to-image conversion	0.05 m

Experimental results of digital model representation

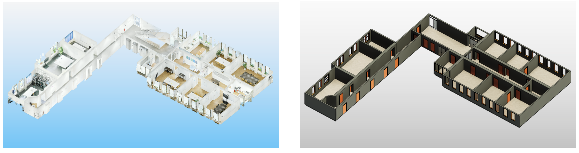
To evaluate the performance of the proposed method for the automatic creation of digital building models, the parameter values of walls, doors, and windows in the reconstructed models are compared against those in the reference models. Corresponding elements in both models are identified based on their spatial coordinates using a buffer zone with a fixed dimension of 10 cm (Table 7). Reconstruction accuracy is then assessed by computing Precision and Recall values according to Equations 2–3. The results indicate a mean accuracy of about 7 cm in estimating model parameters, demonstrating the effectiveness and reliability of the proposed method for the automated creation of digital building models corresponding to the schematic design at LOD 200.

Table 7: Accuracy evaluation of the reconstructed digital building models.

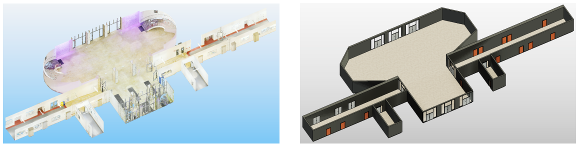
Dataset	Data (1)	Data (2)	Data (3)	Data (4)
Wall:				
precision	0.96	0.68	0.95	0.96
recall	0.85	0.81	0.86	0.96
δ location (cm)	4.9	6.2	5.3	6.8
δ dimension (cm)	4.8	5.7	5.7	6.7
Door and Window:				
precision	0.97	1.00	0.95	1.00
recall	0.98	0.84	0.56	0.93
δ location (cm)	8.7	7.1	6.4	6.6
δ dimension (cm)	9.1	8.7	9.3	9.8

Conclusion

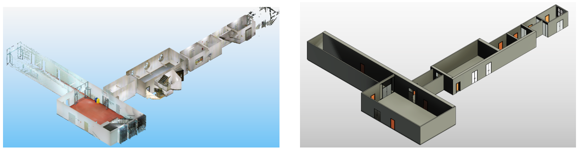
This research introduces a hybrid bottom-up, top-down AI method for the automatic creation of digital building models using laser scanner point cloud data and images. Unlike prevalent bottom-up methods, the proposed method combines existing knowledge from building design and construction with AI capabilities in scene understanding to improve the geometric consistency and the semantic relationships between elements in the resulting digital building models. The proposed parametric model fitting method through global optimization, overcomes the challenges posed by noise and data gaps, minimizing their impact on the accuracy and completeness of the overall model reconstruction. The proposed method for the detection of door and window elements using AI object detection model has shown superiority over the bottom-up semantic segmentation method, allowing for the detection of doors and windows with various types and appearances within the indoor environment. Despite careful consideration, the proposed method encounters limitations in accurately segmenting and modeling curved wall instances. Furthermore, the current door and window detection approach is restricted to identifying elements with rectangular geometries, lacking the capability to detect instances with curved or circular shapes. Future research can focus on addressing these challenges and enhancing the geometric and semantic richness of digital building models by integrating and representing additional structural and architectural elements, such as staircases, columns, and other relevant elements.



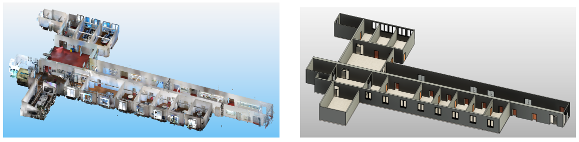
Data (1) - NavVis office building



Data (2) - TUM main entrance



Data (3) - TUM building (1) - Floor (2)



Data (4) - TUM building (1) - CMS chair

Table 8: Overview of the point cloud data and their corresponding reconstructed digital building models.

Acknowledgments

The research presented has been performed in the frame of the project AI4TWINNING (Artificial Intelligence for the automated creation of multi-scale digital twins of the built world) funded by the TUM Georg Nemetschek Institute Artificial Intelligence for the Built World (GNI).

References

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., and Savarese, S. (2016). 3D Semantic Parsing of Large-Scale Indoor Spaces. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 1534–1543, Las Vegas, NV, USA. IEEE.

Brilakis, I., Pan, Y., Borrmann, A., Mayer, H.-G., Rhein, F., Vos, C., Pettinato, E., and Wagner, S. (2019). Built environment digital twining. International Workshop on Built Environment Digital Twinning presented by Technical University of Munich.

Czerniawski, T., Sankaran, B., Nahangi, M., Haas, C., and Leite, F. (2018). 6D DBSCAN-based segmentation of building point clouds for planar object classification. Automation in Construction, 88:pp. 44–58.

Dimitrov, A. and Golparvar-Fard, M. (2015). Segmentation of building point cloud models including detailed architectural/structural features and MEP systems. Automation in Construction, 51:pp. 32–45.

Jocher, G., Chaurasia, A., and Qiu, J. (2023). Ultralytics YOLO.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In Proceedings of ICNN'95 - International Conference on Neural Networks, volume 4, pages 1942–1948 vol.4.

Mehranfar, M., Vega-Torres, M., Braun, A., and Borrmann, A. (2024). Automated data-driven method for creating digital building models from dense point clouds and images through semantic segmentation and parametric model fitting . Advanced Engineering Informatics, 62:102643.

Noichl, F., Collins, F. C., Pan, Y., Mafipour, M., Mehranfar, M., and Borrmann, A. (2024). From purpose to technology: A requirements-driven approach to designing digital twin implementations. In Jebelli, H., Asadi, Somayeh andMutis, I., Liu, R., and Cheng, J., editors, Digital Twins in Construction and the Built Environment, pages 91–130. American Society of Civil Engineers (ASCE).

Ochmann, S., Vock, R., and Klein, R. (2019). Automatic reconstruction of fully volumetric 3D building models from oriented point clouds. ISPRS Journal of Photogrammetry and Remote Sensing, 151:pp. 251–262.

Pan, Y., Braun, A., Borrmann, A., and Brilakis, I. (2022). Enriching geometric digital twins of buildings with small objects by fusing laser scanning and AI-based image recognition. Automation in Construction, 140:104375.

Pan, Y., Braun, A., Borrmann, A., and Brilakis, I. (2023). 3D deep-learning-enhanced void-growing approach in creating geometric digital twins of buildings. Smart Infrastructure and Construction, 176(1):pp. 24–40.

Sacks, R., Brilakis, I., Pikas, E., Xie, H. S., and Girolami, M. (2020). Construction with digital twin information systems. Data-Centric Engineering, 1:14.

Tang, S., Li, X., Zheng, X., Wu, B., Wang, W., and Zhang, Y. (2022). BIM generation from 3D point clouds by combining 3D deep learning and improved morphological approach. Automation in Construction, 141:104422.

Tran, H. and Khoshelham, K. (2020). Procedural Reconstruction of 3D Indoor Models from Lidar Data Using Reversible Jump Markov Chain Monte Carlo. Remote Sensing, 12(5):838.

Trevor, A. J., Gedikli, S., Rusu, R. B., and Christensen, H. I. (2013). Efficient Organized Point Cloud Segmentation with Connected Components. In Proceedings of the semantic perception and mapping exploration, pages 1–6, Karlsruhe, Germany. last access: March 09, 2024.

Zhao, H., Jiang, L., Jia, J., Torr, P. H., and Koltun, V. (2021). Point Transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 16259–16268, Montreal, QC, Canada. IEEE.