Advanced Engineering Informatics 62 (2024) 102643

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Automated data-driven method for creating digital building models from dense point clouds and images through semantic segmentation and parametric model fitting

Mansour Mehranfar *, Miguel A. Vega-Torres, Alexander Braun, André Borrmann

Chair of Computational Modeling and Simulation, School of Engineering and Design, Technical University of Munich, Arcisstr. 21, Munich, 80333, Germany

ARTICLE INFO

Dataset link: https://doi.org/10.14459/2024mp1742891

Keywords:
Digital model
Point cloud
Top down approach
Artificial intelligence
Semantic segmentation
Object detection

ABSTRACT

This paper proposes an automated method for creating semantic digital building models using dense point clouds and images. The method employs a hybrid bottom-up, top-down approach, integrating artificial intelligence capabilities in scene understanding with domain engineering knowledge to overcome challenges in indoor 3D reconstruction. The pre-trained PointTransformer semantic segmentation model extracts thirteen building objects, where the wall and ceiling segments are utilized in a 3D space parsing algorithm. The parameterized floor plan map is then generated using a data-driven approach, enabling the creation of an extruded volumetric digital model. Additionally, the YOLO_{V8} object detection network recognizes doors and windows in images derived from projected points of the wall instances. The validation results for six building datasets with different layouts showcase the effectiveness of the proposed model reconstruction algorithm, with a mean error of about 7 cm between the parameters of elements in digital reference models and reconstructed models. This highlights AI's potential in automating the creation of digital models for the real world.

1. Introduction

Today, Building Information Modeling (BIM) and Digital Twinning have emerged as transformative technologies within Architecture, Engineering, Construction, and Operations (AECO). While BIM offers a comprehensive digital portrayal of a building's physical and functional attributes, Digital Twins (DTs) expand upon this concept by generating dynamic virtual replicas [1]. In contrast to static models, these replicas faithfully emulate not only the structural components but also integrate real-time data to mirror the evolving behaviors and performance nuances of the physical asset. This capability enhances communication and collaboration for facility management and re-design purposes [2]. At the core of these advancements lies the creation and utilization of updated digital models encompassing geometry and semantics. These models incorporate not only the physical shape and layout of structures but also rich semantic data, capturing detailed information about the components and functionalities [3,4]. This enables practitioners to visualize and simulate different scenarios in the built environment and refine decision-making processes for optimal outcomes.

Recently, significant progress has been made in developing sensors and reality-capturing techniques that enable the accurate capturing of built environments on different scales. Laser scanning technology, in particular, offers an accurate and efficient means to capture and analyze building data, aiming to enhance building performance and sustainability. The process of constructing digital building models using the laser scanner point cloud entails the generation of a virtual replica of the physical assets of a building. Despite the potential advantages of high-density point cloud collection at elevated speeds and precision, the scanning process for large-scale buildings has consistently been associated with challenges such as complex space layouts, clutter, and obstructions. The raw data from sensors such as images and point clouds needs extensive processing to derive a high-end geometric–semantic model such as building information models (BIM models) suitable for engineering purposes.

Currently, creating high-quality digital representations from raw remote sensing data demands significant manual effort and time. The developed methodologies for the creation of digital building models are predominantly sensitive to suboptimal data quality and often face challenges in effectively reconstructing meaningful objects and establishing their interrelationships. However, creating an accurate digital building model grounded in the concepts of BIM and DT needs a comprehensive understanding of the intricate relationships among assets within the built environment [5,6].

* Corresponding author.
E-mail address: mansour.mehranfar@tum.de (M. Mehranfar).

https://doi.org/10.1016/j.aei.2024.102643
Received 4 March 2024; Received in revised form 27 May 2024; Accepted 7 June 2024
Available online 20 June 2024

Recent technological advancements have transformed AI techniques into unique solutions in the realms of computer vision and computational modeling, addressing problems and tasks related to the understanding of complex indoor scenes. This research paper aims to propose a novel framework for the automated creation of parametric digital building models with rich semantics and coherent geometry (at LOD 200) using a dense laser-scanner point cloud and images. This involves the representation of a virtual volumetric model of a building system, where all graphical and non-graphical information, including geometric properties and spatial relationships, is presented [7].

Most of the developed methods for creating volumetric building models estimate the parameters of elements individually. In contrast, our proposed method aims to integrate the advantages of the parametric modeling concept and model fitting through optimization to simultaneously estimate the parameters of the entire model consistently. The proposed method integrates domain engineering knowledge in the design and construction of buildings with the capabilities of AI techniques to formulate contextual relations between elements. This enables the creation of a parametric digital model with consistent geometry, offering the flexibility to adjust parameter values (such as length, width, height, and location) during the model reconstruction process. The resulting parametric digital building models allow for frequent geometric updates throughout their operational lifespan and can effectively meet specific operational needs, striking a balance between information richness and practicality.

This paper outlines the principles of the developed research works in [8,9]. In particular, the main contributions of our current research are as follows:

- hybrid bottom-up, top-down approach for automatically creating digital building models.
- Aligning the capabilities of AI methods in scene understanding with domain knowledge.
- Creation of digital building models through parametric model fitting.
- Detection of doors and windows in indoor space using an object detection network.
- Creating digital building models corresponding to the schematic design at LOD 200.

The paper is structured as follows: Section 2 presents an extensive literature review on the developed methods for creating digital building models using point cloud data. Section 3 offers a comprehensive theoretical exposition of the developed methodology. Section 4 presents several case studies to substantiate the feasibility of the proposed approach. Lastly, Section 5 discusses the primary findings of the research and outlines the potential avenues for future research directions within the field.

2. Background

Over the last decade, a significant number of studies have explored various aspects of creating digital building models under the term 'Scan to BIM', where authors process remote sensing data to generate digital models. According to the literature, the automatic process of creating digital models for building structures is typically divided into two major steps: (1) point cloud processing for semantic data labeling and (2) geometry provision for representing digital models. Among the various developed methods, this paper specifically focuses on the reconstruction of indoor digital models using point clouds.

2.1. Reconstruction of indoor digital models

Point clouds are crucial tools for creating accurate 3D models of indoor environments. However, raw point cloud data is typically unstructured and challenging to interpret without proper labeling. Semantic labeling is a critical process that involves assigning semantic labels to individual points or groups of points based on their geometrical-spectral attributes and features. This process is essential for distinguishing between structural elements within the environment. Various algorithms and techniques, such as clustering, semantic segmentation, and classification, can be employed through data-driven, model-driven approaches or advanced AI techniques to achieve accurate labeling.

2.1.1. Data-driven or bottom-up approaches

Bottom-up approaches, here denoted also as data-driven methods, involve direct interpretation from a point cloud, beginning by labeling several random seed points and gradually extending to all points until a higher-level surface, volume, or model is achieved. These higher levels are commonly represented by meshes [10], voxels [11], and planes [12]. In this regard, normal vectors, curvatures, and RGB values are typical features used in common data-driven methods such as Region Growing (RG), Model-based, and Edge-based to differentiate between the geometrical and spectral details of surfaces (Table 1).

In [13], the authors developed an automatic 3D reconstruction framework that used the voxelized point cloud to recognize patches such as walls, ceilings, or floors based on boundary limits. In [14], the authors proposed a supervised region-growing method for segmenting unstructured point clouds using geometric features like surface roughness and curvature. In [15], the authors proposed a Knowledge-driven method that first segments the point cloud into five classes (including ceilings, walls, floors, beams, and clutter) using a surface-growing algorithm. The wall-beam center lines are then extracted to partition the building layout space into individual rooms. Next, a series of topological rules derived from domain knowledge are applied to maintain the consistency of walls and beams during model reconstruction under the Manhattan assumption. In [16], the authors proposed a pipeline for quickly extracting the vertical elements in dense building point clouds using image processing and computer vision techniques. In [17], the authors proposed an adaptive down-sampling method for segmenting planar and non-planar surfaces. The proposed method calculates changes in the normal vector direction within a specified neighborhood to detect the edge points. A similar strategy was later proposed in [18,19], which used a combination of geometrical features of planar surfaces and their topological relations (e.g., distances and parallelism) to reconstruct the indoor volumetric models.

In other studies, researchers have devised techniques employing mathematical formulas to iteratively fit basic geometric shapes like spheres, cylinders, and planes to points and cluster them based on the most correlation with the predefined shapes. In [20], the authors used the Random Sample Consensus (RANSAC) algorithm to detect planar surfaces of main structural elements (e.g., walls, floors, ceilings, etc.) within the point cloud. In [21], the authors developed an automated algorithm for architectural 3D interior reconstruction from 3D point clouds. The proposed method employs the Hough Transform to detect 3D planes and then calculate the intersection of merged planes to create the planar building model.

2.1.2. Model-driven or top-down approaches

The top-down approaches, here denoted as model-driven methods, ensure the geometrical coherency of building models using predefined geometry, relations, and constraints (Table 2). In [22], the authors proposed an automatic method for reconstructing volumetric indoor models through multi-label optimization. The proposed approach maximizes visibility overlaps from different viewpoints in indoor spaces and the point coverages of vertical surfaces. In [23], the authors proposed a global optimization method for creating parametric 3D models. The method effectively distinguishes between exterior and interior structural elements by maximizing the coverage of orthogonality points projected onto the floor plan. To enhance their previous work, the authors proposed a novel method for reconstructing volumetric models for multi-story buildings [24]. They defined the creation of volumetric digital models as an integer linear optimization problem, maximizing

Table 1
Overview of the data-driven or bottom-up approaches for the creation of digital building models.

Reference	Highlight and details	Limitations
[13] Xiong et al. (2013)	Modeling the main structural components Detection and modeling the doors and windows Robust to the clutter and occlusion	Reconstruction the surface based models Restrict to the Manhattan-world layout
[14] Dimitrov and Golparvar-Fard (2015)	Use the Region Growing for points segmentation Employing the surface roughness and local curvature features	Require parameter adjustment Over segmentation problems
[15] Xiong et al. (2023)	knowledge based BIM models reconstruction Automatic room segmentation Combining the optimization of room segmentation and geometric regularization	Restrict to the Manhattan-world layout Use the distance and angle thresholds
[16] Vega et al. (2021)	Detection of vertical objects in large point clouds Use the image processing over vertical projections Use deep learning to checks the cross-sections	Require inferring geometric constraints on the target objects
[17] Qiu et al. (2022)	Develop an adaptive down-sampling method Keep the critical geometric/semantic information Geometry-based segmentation to identify edge points and non-planar points	Require thresholds adjustment
[18] Macher et al. (2015)	Multi-scale segmentation of building point cloud Segment floors, rooms and planes 2D Region Growing for room segmentation Use RANSAC to separate the 3D planes	Semi-automatic Require thresholds adjustment Need for classification of extracted plans into groups such as walls, columns, etc.
[19] Nikoohemat et al. (2020)	Reconstruction of volumetric 3D model Use geometric features and topological relations to classify planar surfaces Reconstruction of spaces based on space-enclosure Heuristic control rules to verify the consistency of the final 3D model	Challenges in the reconstruction of columns in walls Sensitive to the noise points caused by glass reflection Incapable of detection doors in the closed state
[20] Arikan et al. (2013)	Create polygonal models from point clouds Use the RANSAC to separate the planes and extract the boundary of polygons Utilize local adjacency relations among parts to enforce the connection of elements	Required thresholds adjustment
[21] Dumitru et al. (2013)	Use 3D Hough Transform plane detection algorithm Use the Support Vector Machine (SVM) to detect opening candidates from extracted lines in the depth images	Require parameter adjustment Restrict to the Manhattan-world layout

the number of supporting points belonging to the volumetric bounding surfaces and the probability of the surfaces' visibility from locations inside each space. In [25], the authors presented a procedural-based hybrid method integrating shape grammar and a data-driven approach that utilizes a reversible jump Markov Chain Monte Carlo (rjMCMC) algorithm to guide the automated application of grammar rules in deriving indoor digital models for both Manhattan and non-Manhattan environments. In [26], the authors introduced a progressive model-driven approach for the 3D modeling of indoor spaces employing watertight predefined models. This approach initially segmented spaces into rectangular and non-rectangular regions with an even number of sides. Subsequently, a point density occupancy map is used to enhance the level of detail in the intrusion and protrusion parts of models.

Generally, the major challenge of the model-driven approaches is accurately defining the basic geometric relationships and constraints. In contrast, data-driven methods can be utilized in more complex building designs and allow the creation of digital building models that are closer to the real world. However, these approaches are particularly sensitive to data quality, especially regarding occlusion, and their performance may decline in the presence of challenges such as clutter or noise. Moreover, they require high-quality data to achieve optimal performance.

2.1.3. AI techniques for reconstruction of indoor digital models

In recent years, the substantial growth of AI and machine learning (ML) concepts has yielded promising results in the field of computer vision, particularly in the semantic understanding of large-scale point cloud data (Table 3). Unlike traditional bottom-up and top-down data-driven approaches, AI techniques and ML models can learn various characteristics of different datasets without the need for manual selection and fine-tuning of decisive features.

In this regard, in [27], the authors proposed an automated information modeling framework to recognize construction objects and their properties from point cloud data using an AI approach. The proposed workflow utilized the PointNet++ architecture to accurately classify points into twelve pre-defined classes (including building elements, temporary structures, and equipment) based on the values of XYZRGB and intensity characteristics. In [28], the authors developed a deep learning method called Scan2BIM-NET to classify six building components, including walls, ceilings, floors, beams, columns, and pipes. The method utilizes RGB values and geometric features such as normal vectors and curvature. It employs one CNN for assigning semantic labels and another for assigning geometric labels, with an additional RNN used to enforce coherence between the semantic and geometric labels. In [29], the authors proposed a multi-step algorithm integrating laser scanner point cloud and RGB images to enrich the geometric digital building models. The method employs AI-based image segmentation to extract object classes, including electrical elements, safety elements, plumbing system elements, and other objects (door signs, board) from images. These extracted classes are then mapped to the 3D point cloud, segmenting it into point clusters. Subsequently, to create the digital building model, geometric primitives' shapes are fitted to the point clusters.

In [30], the authors proposed an automated reconstruction method for creating digital building models. The proposed method initially clusters planar segments per wall element using a parallel region-growing method combined with a Conditional Random Field (CRF). Next, the parametric volumes are fitted to each cluster using least squares and

Table 2
Overview of the model-driven or top-down approaches for the creation of digital building models.

Reference	Highlight and details	Limitations
[22] Mura et al. (2014)	Model-based method for 3D building reconstruction Robust to the noise and clutter Automatic room segmentation using construction of cell complex in the 2D floor plane Model buildings with Manhattan and non-Manhattan layout	Challenges in modeling slanted wall and ceiling with different height Focus on the robust extraction of the basic room shapes without architectural details Rely on the assumption that each room contains at least one scan position
[23] Ochmann et al. (2016)	Reconstruction of parametric/volumetric building models Use a global optimization to determine configuration of walls and partitioning the spaces Model buildings with Manhattan and non-Manhattan layout	Challenges in segmentation of spaces which are not completely enclosed by walls Require the availability of separate scans Failure in modeling wall elements which are not connected to other walls Modeling only piecewise linear wall structures
[24] Ochmann et al. (2019)	Reconstruction of volumetric 3D building model Employ an optimization method to arrangements of volumetric wall entities Incorporate hard constraints to fit a consistent volumetric model to the observed data Model buildings with Manhattan and non-Manhattan layout	Problem in computing 3D cell complex Challenges in modeling the slanted walls and ceilings Challenges in handling optimization process for very large datasets
[25] Tran and Khoshelham (2020)	Procedural modeling of indoor environments using point cloud Employ reversible jump Markov Chain Monte Carlo (rjMCMC) to automated application of grammar rules in the derivation of a 3D indoor model Model buildings with Manhattan and non-Manhattan layout	Limited to buildings with planar surfaces Challenges in modeling the slanted walls and ceilings
[26] Abdollahi et al. (2023)	A progressive model-driven approach for the 3D modeling of indoor spaces Robust to the noise, local gaps and clutter	Restrict to the Manhattan-world layout Over/under segmentation problems Sensitive to the density of point cloud data Reconstruction the surface-based models

RANSAC algorithms to extract the wall geometry parameters. Finally, various connection types, including intersecting, orthogonal, blended, and direct connections, are employed to reconstruct complete wall structures with consistent topology.

In [31], the authors developed a deep learning-based automatic pipeline that initially employs the PointNet semantic segmentation network to identify building objects within the point clouds. Subsequently, the DBSCAN clustering method is utilized to separate 3D object instances and extract their corresponding bounding boxes. Finally, information from the extracted bounding boxes is employed to create parametric object models. In [32], the authors proposed an automated pipeline combining 3D deep learning and an improved morphological approach for creating volumetric BIM models. The proposed method uses the RandLANet semantic segmentation network to separate building components within the point cloud. Next, a morphological approach is used to separate individual 3D spaces. Subsequently, the extracted space boundaries are modified through an energy minimization method using the Markov Random Field energy function. Finally, a grammar-enhanced point-line polygon and parametric description are used to generate the BIM models.

In [33], the authors developed an automated algorithm for creating BIM models using the Photogrammetric point clouds. The proposed method utilizes the DeepLab semantic segmentation network to group elements (e.g., walls, slabs, and columns) in images collected from the environment. Next, the inverse Photogrammetric pipeline is used to recognize element categories in the point cloud by projecting isolated 3D planes into 2D images. Finally, the extracted information from segmented point clouds is used to create the parametric BIM model.

In [34], authors proposed an automatic algorithm for creating geometric digital models of indoor building environments. The method utilized a pre-trained KPConv model for the semantic segmentation of indoor point clouds and extraction of main structural elements, such as walls, ceilings, and floors. The extracted information is subsequently employed to detect void spaces and rooms within the indoor environment, serving as initial seed points for the proposed void-growing approach in the creation of geometric digital building models. In [35], the authors developed a multi-step data-driven algorithm enriched with AI techniques for the 3D reconstruction of building models at LOD 400. The proposed approach employs an AI semantic segmentation network to categorize classes such as doors, door leaves, windows, walls, ceilings, floors, and clutter. Subsequently, a 2D projection combined with a neighborhood graph structure is used to partition 3D space within indoor environments. The RANSAC plane fitting algorithm is then applied to reconstruct 3D models of walls. Finally, bounding boxes are fitted to the points corresponding to each individual instance of door and window elements.

Despite significant progress in the field of AI, the developed networks for scene-understanding tasks still face several challenges that can impede their performance. Generally, the developed networks require numerous distinct labeled datasets to achieve optimal results in the learning process. Acquisition of such data for indoor environments with complex layouts and cluttered scenes is costly and time-consuming. In addition, most AI architectures rely solely on local and global points features for training and decision-making processes and neglect domain engineering knowledge in building design and the interaction of elements. Incorporating such knowledge can serve as a key tool for overcoming common challenges in indoor digital model reconstruction and converting problems into small, manageable tasks, ultimately leading to improved overall accuracy in the reconstruction of digital building models.

2.2. Door and window detection in indoor environment

One of the main tasks in reconstructing a highly detailed digital building model is the detection and modeling of doors and windows, which has significant applications in indoor navigation, path planning, space management, etc. Despite the considerable demand, the automatic detection of door and window elements in indoor environments from point cloud data faces significant challenges.

During the capturing of indoor environments with a laser scanner, doors are typically open. Also, the laser beam does not accurately reflect off the surfaces of glass doors and windows. These result in doors and windows appearing as void areas on walls. Consequently, most of the proposed approaches for detecting doors and windows using point cloud data rely on specific assumptions and are applicable only to doors and windows in the open state. In [36], the authors proposed

Table 3
Overview of the AI-based approaches for the creation of digital building models.

Reference	Highlight and details	Limitations
[27] Park et al. (2021)	Automatic recognizing construction objects and their properties with deep learning approaches Utilize the PointNet++ architecture to classify points into twelve pre-defined classes including building elements, temporary structures, and equipment	Challenges in presenting the relationship between objects and the shape characterization of objects
[28] Perez et al. (2021)	AI based segmentation of building point cloud including planar, nonplanar and mechanical components Use the point s neighborhood features such as the surface roughness, the curvature, and the normal vector	Challenges in group the clutter points into the pre-defined main classes (walls, ceilings, floors, beams, columns, and pipes)
[34] Pan et al. (2022)	Fuse the laser scanning and photogrammetric technologies Detection of small objects of different classes and texts to improve the digital models	The manual registration of the photogrammetric and laser-scanned point cloud
[30] Bassier et al. (2022)	Use a parallel region-growing method combined with a Conditional Random Field to cluster planar segments per wall Model buildings with Manhattan and non-Manhattan layout	Only reconstruct the wall components not the spaces Challenge in extracting the planes of small indoor elements such as columns and shaft panels
[31] Park et al. (2022)	A deep learning-based automatic pipeline for generating digital building models Use the PointNet semantic segmentation network to identify thirteen building objects Creation the parametric object models	Require parameter adjustment Restrict to the Manhattan-world layout Challenges to define the parametric models for all segmented objects
[32] Tang et al. (2022)	Use 3D deep learning method and an improved morphological approach for creating the parametric BIM models Separate volumetric spaces Model buildings with Manhattan and non-Manhattan	Low accuracy in the reconstruction of curved wall structure
[33] Xiang et al. (2023)	Integrating inverse photogrammetry and the deep learning based point cloud segmentation Creation BIM model using parametric IFC elements	Generalization of the 2D semantic segmentation (require huge training data to cover all objects with different shapes) The difference between models and actual scanned objects
[29] Pan et al. (2023)	Automatic 3D reconstruction algorithm based on the void-growing method Use the KPConv model for semantic segmentation of point cloud	Restrict to the Manhattan-world layout Require parameter adjustment
[35] Maximilian et al. (2023)	3D reconstruction of building models at LOD 400 Use AI semantic segmentation networks to categorize classes such as doors, door leaves, windows, walls, ceilings, floors, and clutter Use the neighborhood graph structure to partition 3D space	Require parameter adjustment Error in partitioning 3D space Low accuracy in modeling doors and windows

a knowledge-based methodology for extracting window elements in building point clouds. The method utilized information from main structural elements, such as walls, openings, and roofs, along with features including sizes, positions, orientations, and topology to recognize window elements. In [37], the authors developed a projection-based algorithm combining semantic features and material characteristics to detect open doors and windows in laser scanner point clouds. The method utilizes the RANSAC plane fitting algorithm to separate wall points and subsequently projects them onto the X–Z and Y–Z planes. Then, an improved Bounding Box algorithm is employed to identify the empty regions among the projected wall points, representing potential open doors and windows.

Recently, researchers have integrated geometric data from point clouds with spectral information from RGB images to enhance the accuracy of detection of door and window elements in closed and semi-open states. In [38], the authors developed a robust algorithm for detecting open, semi-open, and closed doors in indoor environments. The proposed method utilized assumptions such as the angle between detected door points and the corresponding adjacent wall and spectral characteristics and shadows in RGB images to detect doors in semi-open and closed states.

In the realm of the use capabilities of AI methods for the detection of door and window elements, in [39], the authors proposed a hybrid bottom-up, top-down network for 3D instance segmentation of main elements, such as walls, ceilings, doors, and windows. The proposed workflow comprises a soft grouping method and a refinement algorithm that assigns multiple classes to each point to alleviate issues arising from semantic prediction errors. In [40], the authors developed a novel instance segmentation network for separating elements, such as doors and windows. The proposed network utilizes a hierarchical point grouping algorithm to progressively merge semantically segmented points into multi-scale groups, enhancing the clustering of points into instance proposals. In [35], the authors proposed an automated workflow for the detection and 3D modeling of key structural elements, such as doors, door leaves, windows, walls, and ceilings in indoor point clouds. The workflow begins by computing geometric features (e.g., planarity, linearity, surface variation, etc.) for individual points. Subsequently, it utilizes various semantic segmentation networks, including PointTransformer, RandLA-Net [41], and KpConv [42], to extract main elements segments. Finally, a bounding box fitting algorithm is employed to create the 3D model for the detected door and window elements.

Most of the developed AI techniques for point cloud scene understanding utilize both geometric and spectral features for training purposes. Due to the complexity of indoor environments and the similarity between the geometric features of closed doors and windows with other structural elements such as walls, distinguishing between these points can become challenging.

2.3. Methods for representation of geometric digital models

3D visualization is a crucial requirement for the development of digital building models, playing a key role in providing a quick and highly detailed representation of the real world. A 3D digital model offers crucial capabilities to experts and planners, enabling them to navigate and simulate within the 3D environment. Meanwhile, one of the essential features required for presenting a comprehensive and dynamic digital representation of built environments is the ability to update the model by adding metadata, semantics, or updating the shapes. In this regard, various methods have been developed in the literature to represent a 3D digital model, including implicit representation, boundary representation (B-rep), procedural modeling, and parametric modeling. Specifically:

Implicit Representation: The implicit shape representation is a method that involves defining symmetric surfaces and primitive objects using mathematical equations in Euclidean space. Implicit methods utilize functions, such as F(x, y, z), to represent curves and surfaces of objects with arbitrary constructive topology [43].

Boundary Representation (**B−rep**): The boundary representation method is a technique for representing 2D or 3D objects based on their vertices, edges, and loops and their topological relationships to form the object. In this method, vertices are defined by their coordinates (x, y, z), and lines (straight or curved) or faces are described by parametric equations [44].

Procedural modeling: The procedural modeling is a method used to create diverse base models by applying a set of rules, algorithms, or operations such as sweeping, extrusion, and chamfering. Due to its consideration of the workflow in creating geometry and topological integration, has the ability to update base models without creating an entirely new model. In this context, Constructive Solid Geometry (CSG) is known as a subset of procedural modeling that creates models of objects by combining 3D primitive shapes (e.g., cuboids, cylinders, spheres, cones, etc.) or a combination of them through operations such as union, intersection, and difference [45].

Parametric Modeling: The parametric modeling is a method used to create a dynamic geometric model that can be manipulated and adopted by changes to its steering parameters. In the parametric models, rules create relationships between different parts of the elements, and any change in the model is handled automatically by internal logic arguments [46,47].

While solid modeling methods of Implicit Representation, B-rep, and Procedural modeling offer precise and accurate representation of models by explicitly defining boundaries and surfaces, they encounter challenges in handling complex relationships between elements of the model. These methods often demand more effort for extensive changes and modifications. In contrast, the parametric modeling approach maintains a parametric format using a set of functions, facilitating effortless modification of geometry through adjustments to parameter values, dependencies, constraints, and the incorporation of metadata and semantic information. This adaptability proves particularly advantageous in the realms of BIM and digital twinning, where bidirectional links facilitate the updating of existing volumetric digital models based on input values [48].

In the realm of digital building model reconstruction, existing literature defines the parametric modeling process as a combination of model reconstruction followed by setting the geometric rules [23]. Additionally, in certain research studies, building parametric modeling has been conceptualized as floor plan generation through a set of inputs and rules, capable of systematically generalizing arbitrary constraints [49,50]. Nevertheless, an effective approach for creating a digital building model with coherent geometry and rich semantics should integrate the principles of parametric modeling into the model reconstruction process. This integration ensures accurate estimation of model parameters and guarantees consistency among the different components of digital models.

2.4. Research gap

Today, the concept of DTs has been extended to indoor environments to support building management and planning during the operation phase. Despite all the progress made, the automatic creation of digital building models using point cloud data has always been associated with challenges in handling large-scale data processing, understanding the topological relationships between elements, etc. These aspects represent obstacles to the automatic creation of volumetric digital models with rich semantics and consistent geometry.

Most developed methods for creating digital building models from indoor point cloud data are based on data-driven bottom-up approaches. In these methods, the data is initially segmented into sub-individual segments, such as object or surface instances, and 3D parametric models are then fitted to these objects. These methods often require future post-processing steps to address the geometric inconsistency. The developed model-driven approaches also utilize domain knowledge to create 3D models with consistent geometry. However, these approaches mainly rely on significant assumptions (e.g., Manhattan layout, equality of height and thickness of walls) and thresholds (e.g., angle and distance). These hinder the development of a generalized algorithm, restricting its applicability to only a limited range of buildings with specific layouts and designs. Nonetheless, to tackle these issues, an effective parametric modeling approach could additionally incorporate parametric design principles into the model reconstruction process by defining rules, dependencies, and relationships between different parts of the digital system. This results in accurately estimating the object's parameters and ensures geometric consistency between different components of digital models.

Most algorithms that detect doors and windows using point cloud data suffer from low accuracy and often require the adjustment of threshold values. While AI semantic segmentation models can achieve high accuracy in separating main building elements (e.g., walls, ceilings, and floors), the geometric and spectral features similarity between door and window elements and other building elements and the presence of clutters lead to significant errors. Consequently, point cloud semantic segmentation outcomes are not reliable enough for accurately creating digital building models, especially for door and window elements. Additionally, methods that utilize geometric features from point clouds and spectral information from RGB images are only feasible when the environment is captured synchronously using the laser scanner and camera sensors or the photogrammetry technique is used to generate a point cloud from images. In our proposed method, following the reconstruction of the digital models of the building structure, an AI object detection network is employed to detect and model the doors and windows elements in images of projected wall points. This approach eliminates the need to capture images using camera sensors and can detect the door and window elements in different states. Further insights into the proposed methodology will be elaborated upon in the subsequent section.

3. Proposed method

This section outlines the proposed workflow for creating parameterized digital building models using the dense RGB point cloud (Fig. 1). The proposed hybrid bottom-up, top-down method aligns AI capabilities in scene understanding with domain knowledge in the design and construction of buildings to increase accuracy and efficiency in large-scale raw building point cloud processing. First, an AI semantic segmentation network separates the main building structural elements, clutter, and furniture components within the indoor point cloud. Subsequently, the derived information of wall and ceiling segments is utilized in a knowledge-based 3D space parser algorithm that converts complex indoor environments into individual spaces. The top-down approach involves designing parameterized building models that represent the current spatial layout of the environment and mirror the typical topology of buildings. The main idea is to utilize a data-driven method to design a parametric digital building model with sufficient freedom and a high consistency level, enabling the update of geometry by manipulating and changing parameter values. The designed parametric model is then fitted to the observed point cloud data using an optimization process to estimate the best values for the entire system's parameters, allowing it to accurately reflect the real environment. This involves estimating dimensional parameters and simultaneously finding the optimal location of the elements using internal logic arguments. To enhance the LOD of the reconstructed digital model, an AI object detection network is employed to detect doors and windows with any state (e.g., open, semi-open, and closed) in the indoor point cloud and integrate them into the digital building model. Further details of each step are provided in the following subsections.

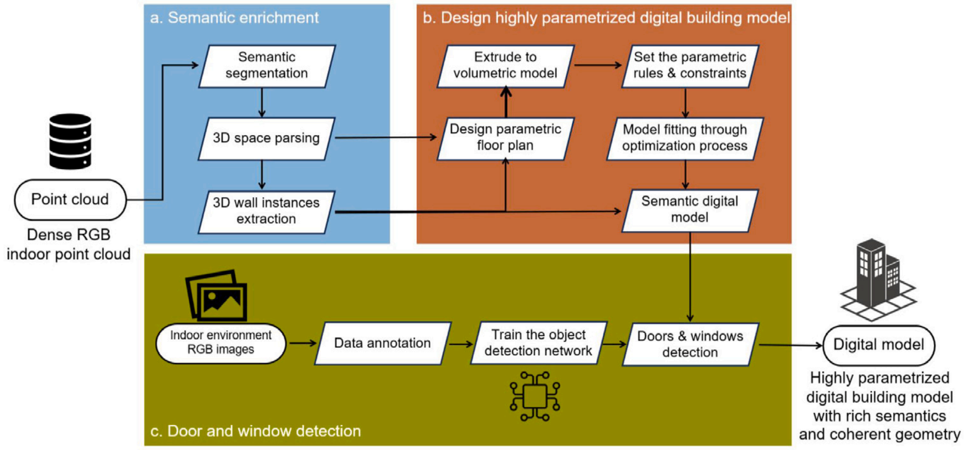


Fig. 1. The proposed workflow for automatic creation of highly parameterized digital building model using laser scanner point cloud.

Table 4
Parameters of pre-trained semantic segmentation model using PointTransformer network.

Parameter	Value
Model:	
Input channels	6
voxel size	0.04
Max voxels	50 000
Number of points per voxel	40 960
Max epoch	512
Optimizer:	
Learning rate	0.01
Momentum	0.9
Weight decay	0.0001

3.1. Semantic enrichment

3.1.1. Semantic segmentation of indoor point cloud

Indoor scenes inherently involve the frequency of objects, complex space layouts, clutter, and obstructions. Ceilings, floors, and walls are the main structural elements of a building that play crucial roles in determining the entire structural property and space layout. Consequently, the first step in the proposed method involves performing semantic segmentation of the indoor point cloud to separate these main elements that comprise the building's structure.

In this study, we utilize a pre-trained semantic segmentation model based on the PointTransformer network [51]. The PointTransformer is a novel and efficient semantic segmentation network that uses the self-attention layer with a combination of simple linear layers and a multi-layer perceptron (MLP). The PointTransformer layer is invariant to permutation and cardinality and is thus inherently suited to point cloud processing. Table 4 shows the configuration and values of the hyper-parameters used in training the network. The values were selected based on the recommendation in [51], which were determined through fine-tuning within certain ranges.

To train the network and label the main structural elements within the point cloud the Stanford 3D dataset (S3DIC) is used. The S3DIC dataset is a well-recognized benchmark for the 3D indoor point cloud processing tasks, comprising thirteen object classes including *Ceiling, Floor, Wall, clutter*, and *Furniture*, and encompasses data from six distinct building areas, as illustrated in Fig. 2 [52]. Specifically, the PointTransformer network is trained on areas 1–4 and 6 and subsequently evaluated its performance on semantic segmentation of area 5. This distribution is designed to fulfill the requirements of adequate training data, effective generalization assessment, computational efficiency, and the mitigation of data imbalance. As mentioned earlier, the primary objective of this step is to separate *ceiling*, *floor*, and *wall* points. In this context, the PointTransformer network has achieved an average accuracy of approximately 93% for segmenting the main structural elements, demonstrating its superiority over other AI semantic segmentation networks [51]. Table 12 presents a comparison of the overall accuracy of the indoor point cloud semantic segmentation using the Point Transformer network with other developed network architectures.

3.1.2. 3D space parsing

Individual rooms and spaces constitute the main building blocks, significantly influencing the structural foundation of a building. Partitioning and separating individual spaces in the environment, along with the inference of prevailing topological relationships between them, are important in creating an accurate and precise virtual representation of a building.

The proposed method utilizes the developed algorithm in [8] to successfully partition 3D spaces, including rooms, hallways, etc. (Fig. 3). In this regard, first, the ceiling points in *th* distance from the wall's points are removed from the ceiling segment. This results in the remaining ceiling point clouds being scattered segments that are distant from the exterior and common shared walls. Then, the density-based clustering method (DBSCAN) is applied to group sparsely distributed remaining points of the ceiling segment into unique clusters [53]. This separates points with high density within a specific neighborhood radius from regions of lower density. Finally, a hierarchical nearest neighbor method is employed to assign the closest cluster label to each 3D point in the building point cloud space including structural elements (e.g., wall, ceiling and floor, column, etc.) and furniture or clutter (e.g., board, sofa, chair, etc.)(Fig. 3b).

The algorithm does not require prior knowledge, such as the layout of indoor environments or the location of sensors, to separate the spaces. The *th* distance for removing ceiling-to-wall points and the distance tolerance for the DBSCAN clustering are the key parameters of the employed method for 3D space parsing. These parameter values are set based on the average width of interior walls in the type of buildings being considered. The pseudo-code for the developed method for 3D space parsing is presented in Algorithm 1.

In building floor plans, individual spaces are separated by common walls and connected through openings. The adjacency graph represents the space allocation in the scene, offering diverse processing possibilities for BIM, re-purposing, and redesign applications. The adjacency graph, denoted as G, can be represented by a symmetric matrix of order $n \times n$. It is defined as $G(V, E)$, where vertices represent individual spaces, and edges indicate adjacency between two spaces. In the realm of indoor digital model reconstruction, the adjacency graph G plays a crucial role in creating a parametric indoor digital model with rich semantics and coherent geometry, especially in representing the linkage and interaction between individual spaces. In this research, the method of calculating the distance between the point clouds of space instances with a neighborhood tolerance threshold is used to determine the adjacency relationships (Fig. 3c) [19].

Algorithm 1 3D space parsing of indoor point cloud

Input:
One point $\boldsymbol{p_S} \in \boldsymbol{S}$, laser-scanned point cloud set $\boldsymbol{S}$;
One point $\boldsymbol{p_i} \in \boldsymbol{C}$, Ceiling point cloud set $\boldsymbol{C}$;
Wall point cloud set $\boldsymbol{W}$;
The Ceiling to Wall points distance threshold $\boldsymbol{D}$;
The minimum number of points per space's cluster $\boldsymbol{M}$;

Function to calculate the distance of a query point $\boldsymbol{p_i}$ in $\boldsymbol{C}$ from all points of $\boldsymbol{W}$ in 3D space *PointToPointcloudDist*();
Function for density-based clustering of remaining Ceiling point cloud
set **C'** *DBSCAN*();
Function to count the number of points in clusters *PointsInCluster*();
Function to assign the cluster id of the remaining Ceiling point cloud
set **C'** to the Laser-scanner point cloud set $\boldsymbol{S}$ using nearest neighbor method *AssignLabelToPoints*();

Initialize:
List used to save remaining Ceiling point cloud set **C'**;
List used to save all labels of *DBSCAN*() result $\boldsymbol{l}$;
List used to save labels for all points in $\boldsymbol{S}$ point cloud set $\boldsymbol{L}$;

Algorithm:
```
for p_i ∈ C do
    if PointToPointcloudDist (p_i, W) < D
      remove p_i from C
   end if
end for
l = DBSCAN(C')
for all l' ∈ l do
   if PointsInCluster (l) <= M
      remove l' from l
   end if
end for
for p_S ∈ S do
   L(p_S)=AssignLabelToPoints(p_S,l)
end for
```

3.1.3. Wall instances separation

Despite the semantic segmentation network's high average accuracy in separating main structural elements, wall segment extraction exhibits the lowest accuracy when compared to ceiling and floor segments. The resemblance in geometric and spectral features between walls and other environmental elements can lead to inaccuracies in segmenting the wall points. The proposed method employs a bottom-up knowledge-based approach to detect the wall footprints in each enclosed space and subsequently separate individual 3D wall instances.

The ceiling and wall elements share common outer and inner boundaries within an enclosed space. In this context, the boundary points of the ceiling can be utilized to extract the footprint of the walls (i.e., the result of projecting the wall points into the 2D X–Y plane) for each individual 3D space. Within a closed space featuring intersecting walls, alterations in the principal component analysis (PCA) parameters of wall instances can indicate breakpoints or abrupt changes. These abrupt changes act as endpoints for each wall instance, signifying alterations in curvature. In this method, the Mean Shift [54] algorithm is initially employed to extract the ceiling's boundary points per spaces (Fig. 4a). Subsequently, the PCA coefficient values are calculated for each boundary point p by considering its k neighbor points and determining the covariance matrix, denoted as c, through Eq. (1):

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

where k is the number of neighboring points, p_i and p also refer to the coordinates of the boundary points being considered. The boundary points are subsequently grouped into three different groups depending on their orientation based on their PCA coefficient values. The points parallel to the X–Z plane, the ones parallel to the Y–Z plane, and the rest that are perpendicular to the X–Y plane, but do not belong to the previous groups. From the top view perspective (X–Y), as illustrated in Figs. 4b and 6, these groups can be denoted as vertical, horizontal, and inclined classes. In this regard, the corresponding ceiling boundary points are classified into these three classes (as in Fig. 4b). In order to group the boundary points that correspond to the same wall instances (such as the pink wall in Fig. 4c), the DBSCAN clustering algorithm is employed, as shown in Fig. 4c. Finally, the points corresponding to each wall within the 3D space are extracted from the original point cloud by considering the buffer b around the wall instances. The method can effectively separate and group wall instances based on their orientation, particularly in datasets that may include clutter and gaps resulting from the presence of glass or mirror materials.

3.2. Digital model representation

3.2.1. Design the parameterized building model

Common methods for creating solid geometric models often employ data-driven approaches, utilizing techniques such as the RANSAC algorithm or least-squares optimization to fit lines and planes to each wall instance individually. Despite their reliable geometric accuracy, these methods frequently result in digital models with inconsistent topology, specifically in the spatial arrangement and connectivity of points or vertices representing the surfaces and structures within the building. These methods often necessitate subsequent post-processing steps to address the inconsistencies, considering certain assumptions and thresholds. This limits the method's applicability to specific building designs and may also displace the wall instance from its previously determined position in the line-plane fitting step, thus reducing geometric accuracy. To address the problems and limitations, we propose a top-down approach to creating digital building models with rich semantics and coherent geometry by designing a prototypical, parameterized digital model.

The first step to designing a parameterized digital building model involves creating a reference model. This model can be a rough representation of the building floor plan, generally depicting the current layout of spaces and the locations of walls in the environment. In this regard, information extracted from the 3D space parsing and wall instance extraction steps is utilized to generate an initial floor plan mask through a plane–plane intersection method (Fig. 5b).

After creating the initial floor plan mask, it is extended into a 3D volumetric representation, this model is created with the Revit API. Next, a set of geometrical–mathematical rules and constraints are considered and applied as internal relations between system elements (Fig. 5c). These define the type of interaction between system elements (e.g., walls, slabs, etc.) and specify the degree of freedom and the domain of changes for the parameters of these elements. The logic behind these rules and constraints stems from existing engineering knowledge in the realm of BIM and building design. For instance, office buildings are typically designed based on the 'Manhattan world' assumption. This defines the perpendicular connection between elements and limits their positional movement within a specific domain and direction. As a general approach, in the designed parametric digital building model, each wall instance can have one of three possible orientations: horizontal, vertical, or inclined. We define the type of interaction between each wall instance and its connected walls. Fig. 6 illustrates the possible direction of movement for each sampled wall instance.

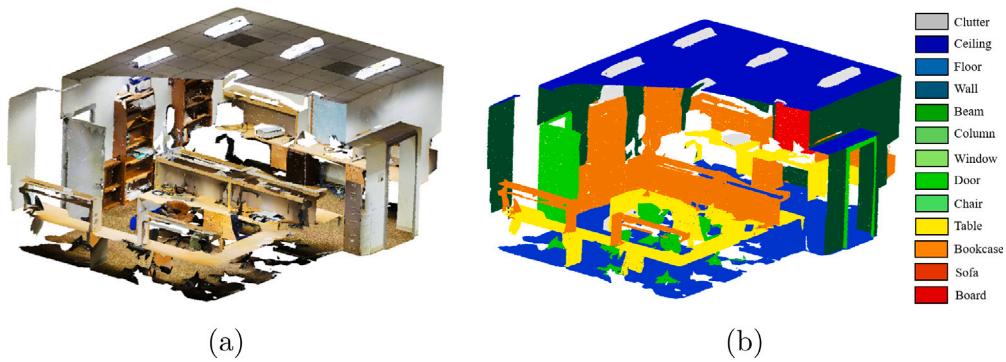


Fig. 2. Semantic segmentation of indoor point cloud: (a) original point cloud of S3DIC Area (5) data, (b) the semantic segmented point cloud.

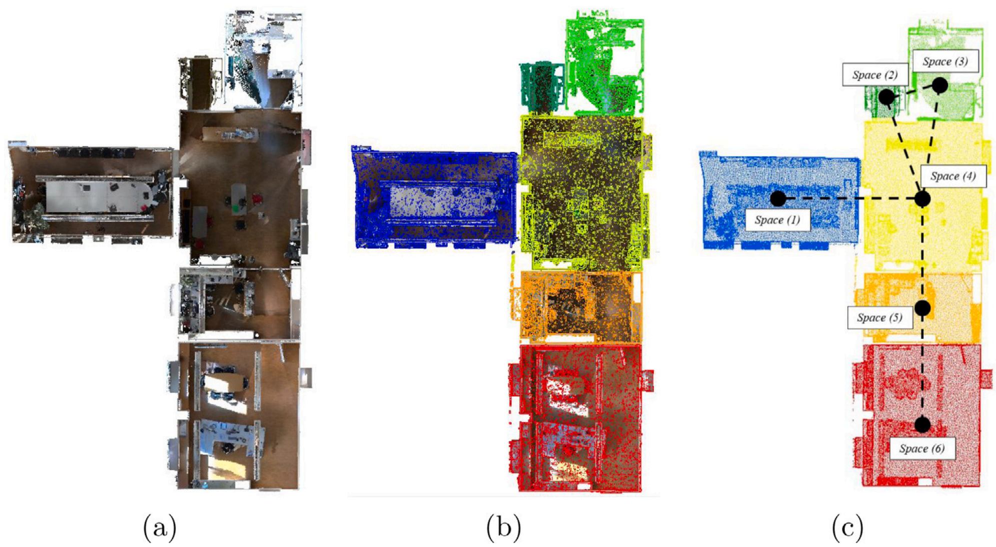


Fig. 3. 3D space parsing of the indoor point cloud using the developed method in [8]: (a) original point cloud, (b) partitioned 3D spaces, (c) the adjacency graph of 3D individual spaces.

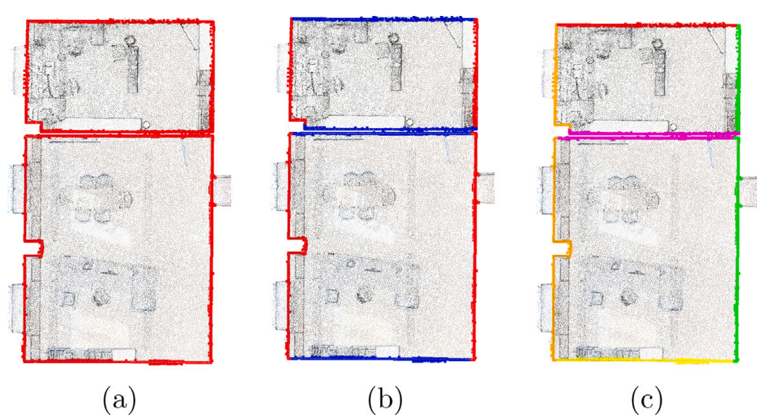


Fig. 4. Wall instances extraction using the developed method [8]: (a) the extracted ceiling boundary points using Mean Shift method, (b) group the boundary points based on their orientation using PCA coefficient, (c) separated wall instances.

The design of the parameterized building models using the proposed method maintains semantic information and enhances topological consistency. This ensures that any change in the internal parameters of a wall, such as length and height, logically affects other related wall elements (Fig. 7).

3.2.2. Optimizing the parameters of digital building model

Despite the consistent semantic topology, the designed parametric model might exhibit low geometric accuracy regarding element property values and their positions within the environment. The initial values of element parameters are extracted from the floor plan mask creation step. Next, the volumetric model is further refined by fitting to

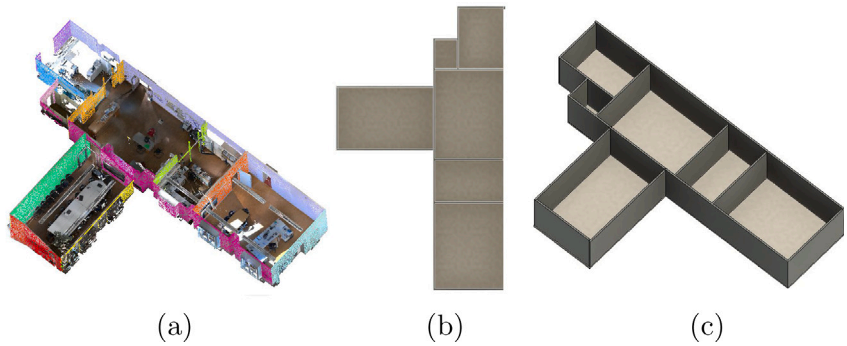


Fig. 5. Design the parameterized building model: (a) separated 3D wall instances, (b) initial floor plan mask, (c) digital building model extruding from floor plan mask (ceiling were removed for better visualization).

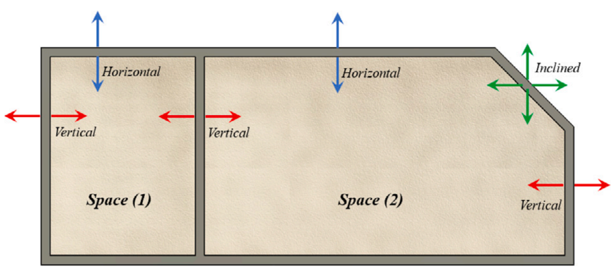


Fig. 6. The possible movement directions for wall instances within the designed parametric model.

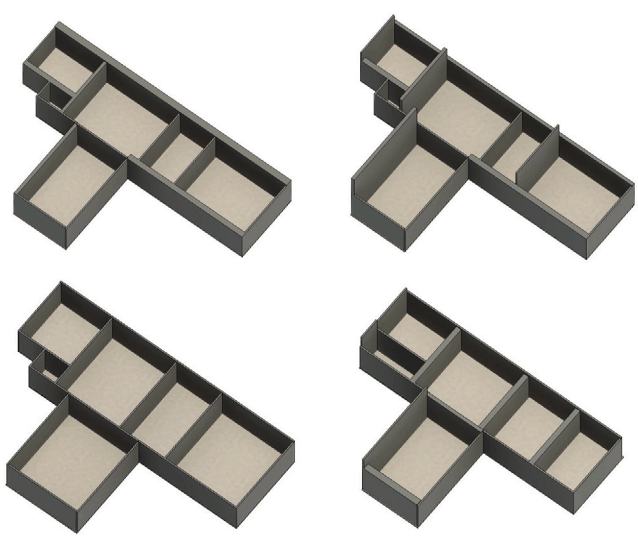


Fig. 7. Designing the parameterized digital model of the building's structure, the process of changing the value of the parameters.

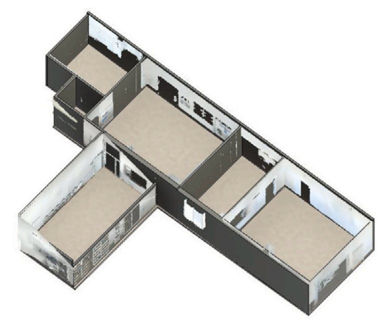


Fig. 8. Fitting the parameterized digital building model to point cloud data using Nelder–Mead optimization.

the point cloud data using the Nelder–Mead optimization method [55] to extract optimal values for the model's parameters (Fig. 8).

Indoor environments are often occluded by furniture and clutter. Consequently, utilizing a data-driven model reconstruction method that optimizes the parameters of each wall individually can result in geometric inconsistencies in the final model, particularly in the case of highly occluded wall instances. In this context, while using global optimization for parametric model fitting may increase computational complexity, it effectively maintains logical consistency throughout the entire model reconstruction process. The algorithm utilizes the type of movement and interaction to adjust the position and length of the wall instances.

For the optimization process, the overall Points-To-Model distance is the objective function to create accurate geometric models by optimizing the dimensional properties and locations. This involves calculating the distance of points from the surfaces of the digital model. In this context, achieving the lowest value for Points-To-Model distance indicates a superior adaptation of the digital model to point cloud data and greater accuracy in estimating model parameters towards their actual values.

Accurately estimating the width of walls shared between spaces is the main challenging task in creating a volumetric building model. Existing literature has introduced various techniques for estimating wall width, relying solely on different assumptions and thresholds. The proposed methods involve considering tolerance for detecting corresponding parallel wall segments and calculating their average point distance as thickness [24,56]. These methods are heavily influenced by noise and clutter and require tolerance adjustment for different building data. To address the problem, we treat the width values for shared walls as unknown parameters in the optimization problem. Walls are considered as boxes with dimensions (width, length, height) and are allowed to have different values during the model fitting process. Additionally, exterior walls, appearing as single planar surfaces, often yield width values of around 1–3 cm through the optimization process. To tackle this issue, a modification is implemented, adjusting the attributes of these walls. Their width is considered with the minimum value observed in the width of shared walls within the respective models.

In designing and optimizing the parametric digital building model, the degrees of freedom are proportional to geometric complexity, specifically the number of walls. This encompasses properties such as wall length, width, height, and their location in 2D space (Table 5). These unknown parameters are incorporated into the optimization problem as both explicit and implicit mathematical equations with geometric constraints. The type and number of equations, as well as unknown values in the optimization problem, vary according to the case study.

(a) (b)

Fig. 9. Transforming the projected wall points into RGB images: (a) projected wall instance points, (b) resulting RGB image.

Table 5
Encoding unknown parameters of the highly parameterized digital building model.

Building element	X_{corner}	Y_{corner}	Parameters		
			Length	Thickness	Height
Wall (1)	P_1	P_2	P_3	P_4	P_5
Wall (2)			P_6	P_7	P_8
Wall (3)			P_9	P_{10}	P_{11}
...			...	...	...
Building model	R_z (the parameter of rotation model around Z axis) P_1, P_2 serve as the origin for applying partial 2D shift				

Table 6
The hyperparameter values employed in training the object detection network using the YOLOv8 architecture.

Parameter	Value
Image size	640
Batch size	8
Epoch	150
Learning rate	0.001
Solver	Adam

Table 7
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

3.3. Door and window detection

In this research, we propose a novel approach for detecting door and window elements in an indoor environment. First, after fitting the digital model of the building structure to the point cloud, the points with a 1 m distance around each wall instance are extracted. This ensures all points belonging to the wall surfaces and elements around are well considered. Subsequently, the points belonging to one of the two walls' surfaces are projected onto the X–Z and Y–Z planes (Fig. 9a). Then, a method involving griding and sampling with dimensions of d is employed to convert the representation of the point cloud into an RGB image (Fig. 9b). The average spectral values of the points within each grid cell are used to estimate the spectral values of the image pixels. In this context, setting small values for the parameter d as sampling distance enhances image quality but results in an increase in the processing time. The resulting images encompass points representing the wall, doors, and windows, as well as all other elements installed close to the surfaces of the wall, as illustrated in Fig. 9.

The proposed method leverages the advantages of AI techniques by employing the YOLOv8 object detection network to detect door and window instances in open, semi-open, and closed states [57]. YOLOv8 utilizes a single neural network to predict bounding boxes and class probabilities directly. Its single-stage architecture allows for a faster training process compared to other architectures, such as Mask R-CNN [57,58].

To train the network, a comprehensive image dataset from two individual sub-categories was compiled, as shown in Fig. 10. The first dataset consists of 214 normal RGB images taken from various buildings at the Technical University of Munich (TUM), showcasing door and window instances in all three possible states: open, semi-open, and closed, with different materials such as timber, glass, and aluminum. Additionally, another dataset comprising 89 images from projected wall points of TUM point cloud datasets was provided to fulfill the diverse training dataset requirement. This inclusion helps minimize challenges arising from differences in image scale, light conditions, and spectral values between normal RGB images and the resulting images from projected wall points. To facilitate the network training process, bounding boxes corresponding to door and window elements are meticulously annotated within the acquired images. Then, the annotated dataset is partitioned into training and validation subsets, with ratios of 80% and 20%, respectively. This ensures an effective balance between the robustness of model training and the rigor of assessing its generalization performance on an independent subset. Table 6 shows the hyper-parameters used for training the object detection network.

To assess the network's performance, annotated element instances in images are compared with the detected instances using standard metrics: Precision, Recall, and mAP, using Eqs. (2)–(4):

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

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

Where TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively. In this regard, the mean average precision (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. As can be seen in Table 7, these values are all above 86%. Also, the precision–recall curve, presented in Fig. 11, illustrates the model's capabilities in balancing precision and recall, showcasing its ability to accurately detect relevant instances while minimizing false positives.

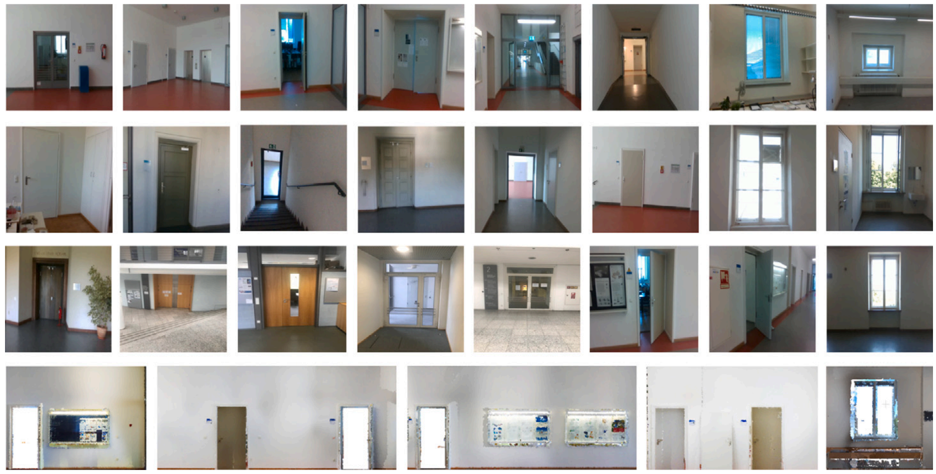


Fig. 10. The TUM image dataset provided to train the door and window detection network.

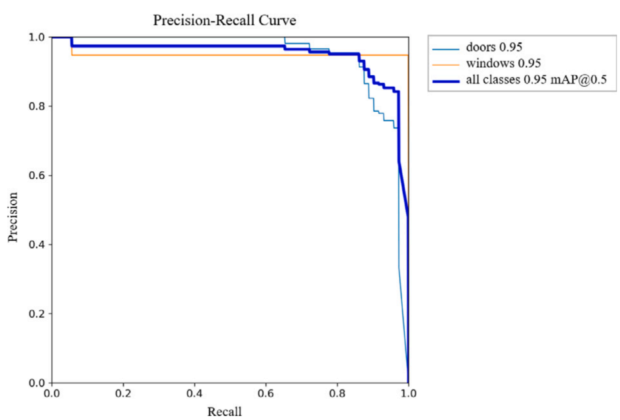


Fig. 11. The Precision–Recall Curve of training the object detection network for door and window detection.

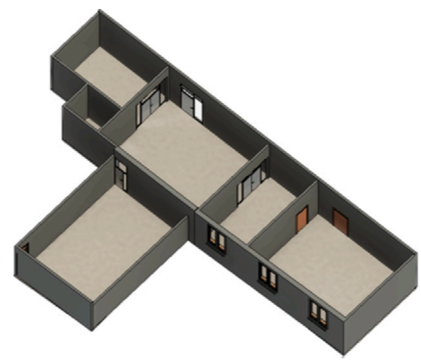


Fig. 12. Parameterized digital building model, with the appropriate representation of doors and windows in the reconstructed digital model.

3.3.1. Representation of doors and windows digital model

After detecting the door and window elements in the images, the coordinates of the detected bounding boxes are projected back from 2D onto the 3D point cloud of the corresponding walls using reverse mapping. In some cases, the presence of gaps and defects in specific parts of the projected wall points may result in mislabeling of detected door and window instances. To address the problem, the heights of the detected bounding boxes are examined. The door elements are consistently connected to the building floor. In this context, considering a confident distance of 25 cm, any door-bounding boxes detected beyond this distance from the building floor height are relabeled as windows. Also, errors in the transformation of projected wall points into images, and the detection of door and window pixels using the trained object detection network can lead to deviations in the dimensions of the detected elements from their actual values. In this regard, a library of door and window elements, including their dimensional characteristics, used in the construction of buildings is compiled, a subset of this library is shown in Fig. 13. To model the door and window elements accurately, the closest model is selected from the library based on dimensional parameters (width and height). Subsequently, the selected model is utilized to replace the primitive dimensions of the detected elements with their actual dimensional parameters. This results in an enhancement of reconstruction accuracy for detected door and window 3D digital models, as well as improved geometric and structural integrity in the reconstructed digital model(Fig. 12).

4. Experimental result

4.1. Case study

In this research study, six distinct indoor point cloud datasets from different buildings of the TUM city campus and NavVis company building office are considered to evaluate the performance of the proposed method for the automatic creation of digital building models with rich semantics and coherent geometry. The building datasets are primarily utilized for educational and research purposes, featuring various areas such as offices, libraries, meeting rooms, hallways, etc. as can be seen in Tables 9–10. Table 8 presents the main characteristics of the datasets. The proposed approach is implemented in Python and MATLAB on a research computer (11th Gen Intel(R) Core(TM) i7-1165G7, with 16.0 GB1053 memory). The considered evaluation metrics encompass various aspects of the proposed method, including accuracy, efficiency, and scalability in crucial terms of geometry and semantics. This comprehensive analysis provides insights into the potential practical implementation of the method for creating digital building models in the real world.

4.2. Experimental results of point cloud semantic enrichment

4.2.1. Semantic segmentation of indoor point cloud

To assess the effectiveness of the employed network for accurately labeling the points, the manually annotated ground truth data is compared with the result of the semantic segmentation network. For each dataset, the standard quality metrics of class-wise intersection over

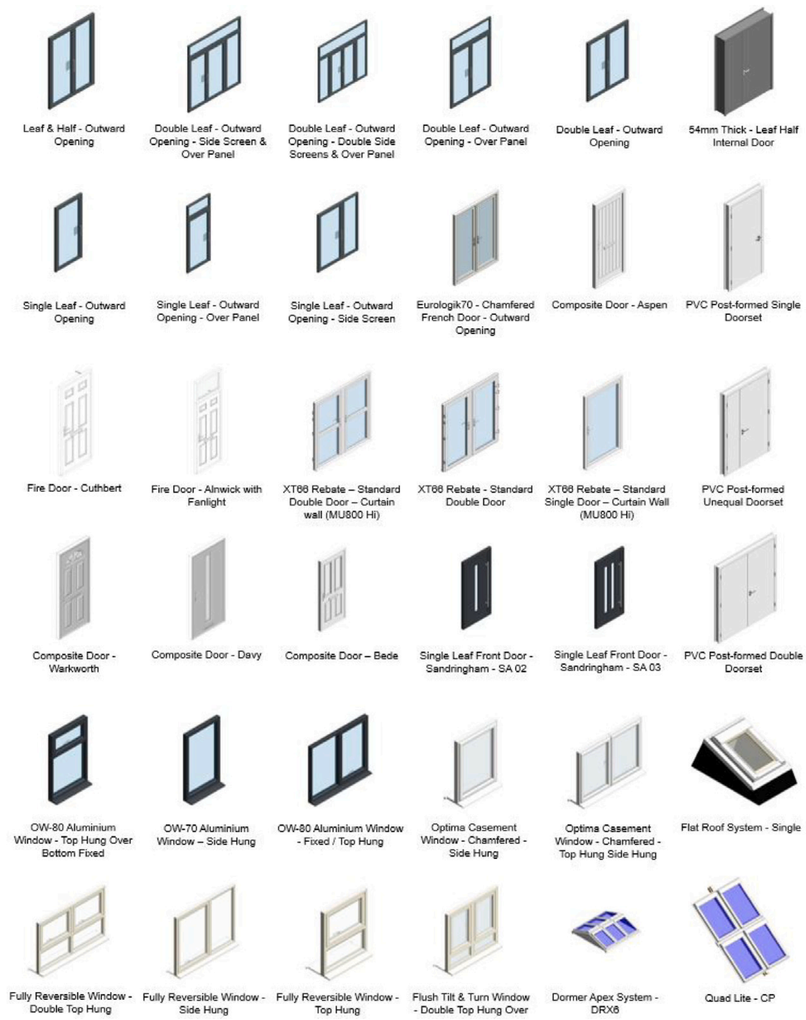


Fig. 13. Subset of the library of parametric door and window elements (https://www.autodesk.de & https://www.bimstore.co).

Table 8
Overview of data used in this research.

Dataset	Length (m)	Width (m)	Number of points	Avg density (r = 1 cm)
NavVis office building	34.03	35.41	56.244.568	9
TUM main entrance	61.61	23.28	37.595.228	6
TUM - Floor (2)	34.42	47.70	11.308.120	2
TUM - Floor (3)	18.26	32.88	13.239.024	2
TUM - Floor (4)	19.58	34.94	10.027.980	2
TUM CMS chair	32.98	59.73	40.506.234	4

union (IoU) and mean of class-wise accuracy (mAcc) for extraction of main structural elements are calculated (Table 11).

$$IoU = \frac{TP}{TP + FP + FN} \tag{5}$$

According to the results, the overall mean IOU for semantic segmentation of main elements across all building datasets is about 89.5%. In this regard, the accuracy of separation of the wall points is lower than that of other elements. Several factors can contribute to this challenge, including the complexity and clutter present in office spaces, the similarity in geometric and spectral features between building elements and furniture, as well as the presence of outlier points (related to elements made with glass and mirrors). These challenges are often inherent in building environments, and addressing them necessitates the preparation of extensive and diverse datasets to achieve optimal performance in training the semantic segmentation network.

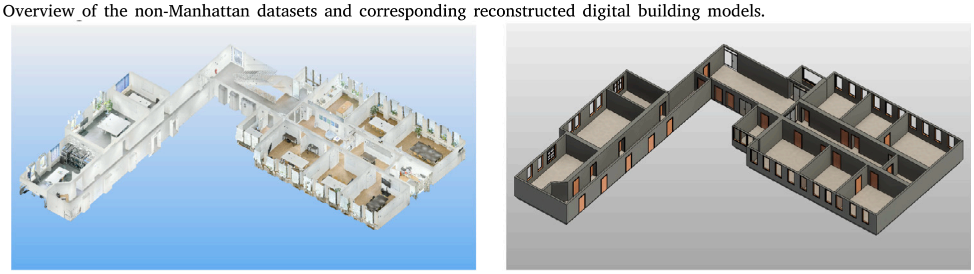
To assess the performance of the employed network for extraction of the main structural elements, a quantitative comparison is conducted between the results of PointTransformer network and other developed architectures, including KPConv and RandLA-Net (Table 12). The models are similarly trained with S3DIC area 1–4 and 6 datasets and tested on building datasets.

According to Table 12, the PointTransformer model yields a higher mean accuracy for segmenting the main structural elements compared to other trained models. Although the KPConv and RandLA-Net networks also exhibit strong performance, there is an average difference of 2.5% in overall mean accuracy compared to the PointTransformer. In contrast to other commonly developed networks that utilize convolutions, the PointTransformer network employs point transformers as the feature aggregation operator in the core of its network. Also, despite the KPConv and RandLA-Net network architectures that use pre-computed kNN indices for considering local neighborhoods, the PointTransformer utilizes a heap sort algorithm, resulting in efficient implementation running time.

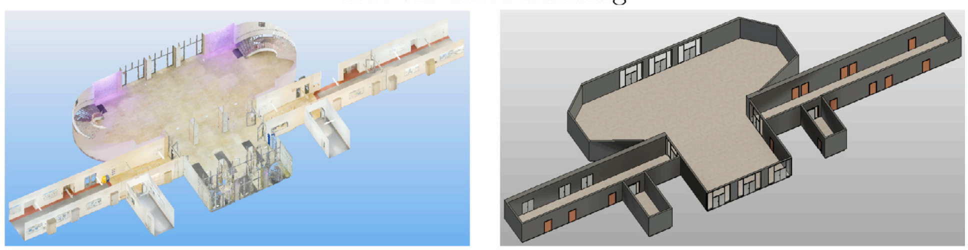
4.2.2. 3D space parsing and wall instances separation

To disjoint 3D spaces, ceiling points located within 30 cm of the wall segment (distance threshold *th*) are initially excluded from the ceiling segment. The DBSCAN clustering method with a 30 cm distance threshold is then employed along with the nearest neighbor algorithm

Table 9
Overview of the non-Manhattan datasets and corresponding reconstructed digital building models.

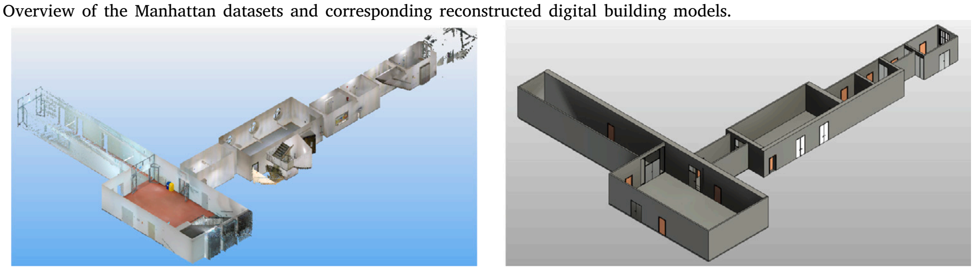


NavVis office building

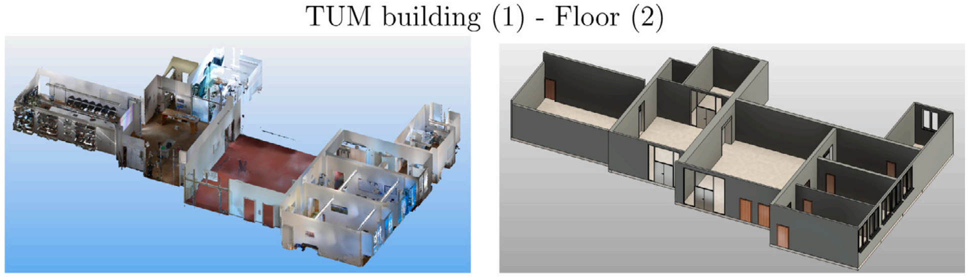


TUM main entrance

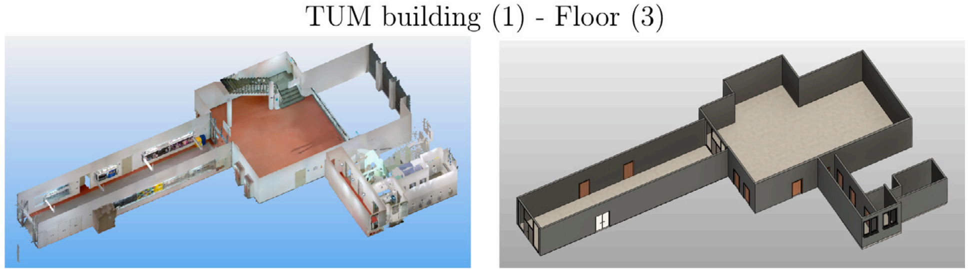
Table 10
Overview of the Manhattan datasets and corresponding reconstructed digital building models.



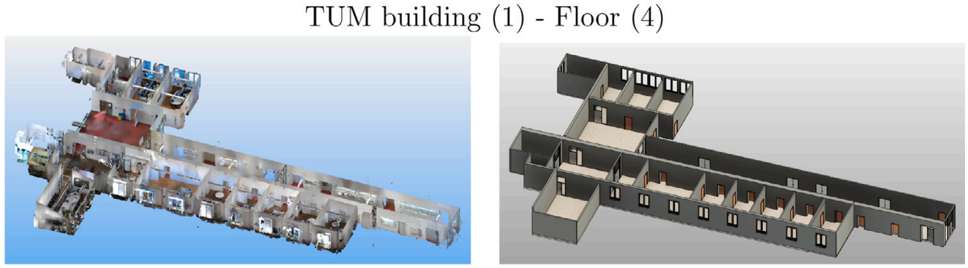
TUM building (1) - Floor (2)



TUM building (1) - Floor (3)



TUM building (1) - Floor (4)



TUM building (1) - CMS chair

Table 11
The results of semantic segmentation on building datasets.

Dataset	mAcc	mIoU	Ceiling	Floor	Wall
NavVis office building	92.6	89.1	91.1	95.5	80.8
TUM main entrance	93.5	90.8	92.4	95.9	84.1
TUM - Floor (2)	92.1	88.4	89.1	94.5	81.7
TUM - Floor (3)	93.3	91.9	93.5	96.6	85.7
TUM - Floor (4)	91.1	87.1	85.2	92.6	83.5
TUM CMS chair	92.7	89.9	91.3	94.5	83.9
Overall	92.5	89.5			

Table 12
Quantitative comparison of the mean accuracy among various network architectures used for extracting the main structural elements within the point cloud datasets.

Network	PointTransformer	KPConv [42]	RandLA-Net [41]
mAcc (%)	92.5	89.26	90.73

Table 13
Accuracy evaluation of 3D space parsing.

Dataset	Rand index(%)
NavVis office building	0.93
TUM main entrance	0.90
TUM - Floor (2)	0.91
TUM - Floor (3)	0.96
TUM - Floor (4)	0.89
TUM CMS chair	0.94
Overall	0.92

to group points in 3D space into distinct clusters. To create the adjacency graph of spaces, a tolerance threshold of 1 meter is considered as neighborhood distance.

Next, the ceiling boundary points are extracted and the PCA coefficients are then calculated for each point considering the 50 nearest neighbor points. The wall instances are subsequently separated by corresponding PCA coefficients and then grouped using the DBSCAN clustering algorithm with a distance threshold of 1 m. Finally, a 10 cm buffer is applied around each separated wall instance to extract the corresponding 3D wall points within the point cloud space.

To assess the performance of the employed algorithm for 3D space parsing, a quantitative comparison is conducted between the results of the employed algorithm and manually partitioned spaces (Table 13). For each dataset, the clustering similarity metric Rand Index (RI) is calculated [59]. In this context, the overall accuracy for 3D space parsing is 92.1%, signifying the effectiveness and performance of the employed algorithm for partitioning 3D spaces in indoor environments. Among all building datasets, the TUM main entrance and building (1) - Floor (4) data have achieved lower RI similarity values in partitioning individual 3D spaces compared to the others. In this context, the use of glass and mirror materials in the environment results in an increase in noise and outlier points as well as distinct void areas within the corresponding walls. This complicates the process of extracting wall instances and leads to errors in disjointing the corresponding space from other spaces.

4.3. Experimental results of digital model representation

After partitioning 3D spaces and extracting the corresponding wall instances, the initial floor plan masks are generated. Subsequently, these masks are extended into a 3D volumetric model. The proposed method employs the PCA technique and analyzes the 50 nearest neighbors around each query point in the X–Y plane. This results in grouping the points belonging to the closest wall surfaces with the same orientation. For each building model, geometrical and mathematical rules and constraints are considered to define the parametric relations and interaction among structural elements such as walls, slabs, and ceilings.

Table 14
The values of parameters used for optimization processes.

Parameters			
Problem	Tolerance-X	Tolerance-Obj	Iterations
Volumetric digital model fitting	0.0001	0.0001	100

Table 15
Accuracy evaluation of digital model reconstruction (the values for the reported parameters in the table are all in cm).

Parameters						
Dataset	Precision	Recall	δ location	δ height	δ length	δ width
NavVis office building	0.96	0.85	4.9	3.2	6.5	4.8
TUM main entrance	0.68	0.81	6.2	4.2	7.3	5.7
Floor (2)	0.95	0.86	5.3	3.4	9.4	4.5
Floor (3)	0.95	1.00	4.4	6.3	6.9	4.6
Floor (4)	0.94	0.81	4.5	2.6	8.3	5.2
CMS chair	0.96	0.96	6.8	5.1	7.7	7.2
Overall	0.90	0.88	5.5	4.1	7.7	5.3

Finally, the parameterized building models are fitted to the point cloud through the optimization process, and the optimal values for the parameters of the building elements are extracted. Table 14 shows the values of optimization parameters used for creating highly parameterized building models.

To evaluate the performance of the proposed approach for the creation of digital building models, a quantitative comparison is conducted between the parameters of the elements in the reference models and corresponding parameters in the reconstructed digital building models (Table 15). In this regard, the corresponding wall elements in both the reference and reconstructed models are identified by utilizing the coordinates of their endpoints, considering a buffer with specific dimensions of 10 cm. For each model, the accuracy of the reconstruction is measured by calculating the corresponding Precision and Recall values using Eqs. (2)–(3). The precision quantifies the proportion of reconstructed actual wall elements relative to all elements reconstructed as wall by the algorithm while recall assesses the effectiveness of the proposed approach in the creation of all actual wall elements present in the reference model.

The overall recall value of about 0.88% in creating all corresponding wall instances, coupled with a mean accuracy of approximately 6 cm in estimating models' parameters, highlights the effectiveness of the proposed method in creating volumetric–parametric digital building models with diverse designs and layouts. According to Table 15, the highest error is associated with estimating the length parameter for wall elements. The office dataset includes furniture attached to the walls, such as bookcases, cabinets, and glass boards, contributing to the presence of clutter and noise. These factors introduce challenges in the separation of 3D spaces and wall instances and subsequently estimating the element parameters, leading to a decrease in overall model reconstruction accuracy.

In addition, the TUM main entrance dataset has achieved the lowest precision and recall values in the creation of the digital model. The dataset includes two curved wall instances. Accurately separating and modeling curved walls necessitates considering small values for the NN value when calculating PCA coefficients and separating entire wall points by grouping segments with different partial curvature angles. In this regard, setting the NN value to 50 separates the entire curved wall into individual wall patches with inaccurate orientation and location, resulting in the creation of extra wall instances. This shows the limitation of the proposed approach in separating and modeling curved wall instances.

4.4. Experimental results of door and window detection

After fitting the structural model to the point cloud data, the projected wall points are converted into raster images using a gridding and

Table 16
Accuracy evaluation of door and window detection (the values for the reported parameters in the table are all in cm).

Dataset	Doors		Windows		Parameters	
	Precision	Recall	Precision	Recall	δ location	δ dimension
NavVis office building	1.00	1.00	0.94	0.97	8.7	9.1
TUM main entrance	1.00	0.84	–	–	7.1	8.7
Floor (2)	0.90	0.87	1.00	0.25	6.4	9.3
Floor (3)	1.00	0.88	1.00	0.85	8.3	6.1
Floor (4)	1.00	0.70	1.00	0.50	4.7	8.2
CMS chair	1.00	0.94	1.00	0.93	6.6	9.8
Overall	0.98	0.87	0.99	0.69	6.9	8.5

Table 17
The result of door and window detection on the projected wall points images.

sampling method with a grid dimension of 5 cm. Next, the trained object detection network is utilized to detect door and window elements in the images (Table 17), and the detected boxes are subsequently projected back into point cloud space. To assess the accuracy of door and window detection, precision and recall metrics are computed by comparing reference models with reconstructed digital building models using a 10 cm buffer. In this context, the parameters of detected doors and windows, including their location and dimensions before adjustment, are compared with the corresponding elements' parameters in the reference building models.

As presented in Table 16, the overall recall values for detecting door and window elements across all datasets are approximately 0.87 and 0.69, respectively. Additionally, the mean accuracy in estimating the corresponding element parameters is about 8 cm. These findings underscore the effectiveness of the proposed approach in accurately detecting and modeling doors and windows, including various states, such as open, semi-open, and closed, within indoor environments. Across all the datasets, the TUM - Floor (4) data has achieved lower recall values than the others. Using various mirrors, glass doors, and windows, along with the presence of different lamps and lighting, creates diverse lighting conditions in the point cloud space. This leads to differences in the spectral features of the projected wall point images compared to real RGB images. These lighting conditions pose difficulties for the trained network in accurately detecting the elements in image pixels and extracting their dimensional parameters. Additionally, the TUM - Floor (2) dataset has the lowest recall value for detecting window elements. This data includes several round window instances, presenting challenges for the employed object detection network specifically designed to detect elements with box-shaped geometry.

5. Discussion

5.1. Data requirements

This research paper presents an automated pipeline for generating parameterized digital building models from raw laser scanner point clouds. The proposed method does not require the intensity feature as input. Still, it utilizes RGB values for semantic segmentation of indoor point clouds and creating images from the projected wall points for door and window detection. According to the information reported in Table 8, the test data utilized for validating our method exhibit average density ranging from 2 to 9 points in a sphere of a radius of one centimeter. Consequently, algorithm parameters were configured accordingly. Thus, the proposed method is well-suited for effectively processing data with similar point densities. Additionally, since the proposed method relies on ceiling information to extract the footprint of the walls and create the digital building model, input data must include only points from a single floor and contain corresponding ceiling points for each space. This method cannot handle scenarios where points from multiple floors are mixed.

5.2. Comparison with other methods

In this section, a quantitative comparison is conducted to evaluate the performance of the proposed method in digital building model creation from point cloud data compared to other developed algorithms. This involves a comprehensive examination of our proposed parametric modeling approach compared to the data-driven method presented in [35]. The developed method initially utilizes the eigenvalues of the covariance matrix of points to calculate geometrical features, such as Planarity, Linearity, Surface variation, etc. These features are then incorporated into the feature-based PointTransformer semantic segmentation network to group the points into classes such as doors, door leaves, windows, walls, ceilings, floors, and clutter. As access to the utilized training data is not possible, the PointTransformer semantic segmentation network, previously trained on S3DIC data areas 1–4 and 6, is again employed. The segmented wall points are then employed in a 2D projection method combined with a neighborhood graph structure to detect individual spaces within indoor environments. Subsequently, the iterative RANSAC plane fitting algorithm is employed to reconstruct the geometric planar 3D model for each detected space. Finally, a bounding box fitting algorithm is utilized to model individual door and window instances in the open state. The implemented algorithm is tested on the building point clouds, and the resulting digital models are compared with the reference digital model. This involves the comparison of the parameters of corresponding reconstructed elements present in both digital building models.

According to Table 18, thanks to the applying parametric modeling process through the optimization, our proposed method has achieved a 1.5 cm mean accuracy superiority in estimating the dimensional parameters of the wall elements. Specifically, the substantial 9 cm difference in estimating the parameters of doors and windows in an open state highlights the capabilities of the proposed AI-based method for detecting doors and windows. The proposed method in [35] utilized the results of point cloud semantic segmentation and subsequent bounding box fitting to model doors and windows. Due to the complexity of indoor point clouds and the similarities between the geometric characteristics of doors and windows with walls and other elements, the results of detecting doors and windows from point clouds often include noise and outliers. This, in turn, leads to substantial errors in estimating the dimensional parameters for the detected doors and windows during the bounding box fitting steps.

Table 18
Quantitative comparison of the results between proposed parametric modeling approach and the reconstruction algorithm proposed by [35] (the values for the reported parameters in the table are all in cm).

Walls			Doors and windows	
Method	δ location	δ dimension	δ location	δ dimension
[35]	4.6	7.2	14.7	17.5
Ours	5.5	5.7	6.9	8.5

Table 19
Comparison of key features of the proposed method with six state-of-the-art methods.

Method	Volumetric walls	Volumetric spaces	Topological relation	Parametric modeling	Door window
Ochmann et al. [24]	✓	×	×	×	×
Nikoohemat et al. [19]	✓	✓	×	×	✓
Tran and Khoshelham [25]	✓	✓	✓	×	×
Wu et al. [56]	✓	×	×	×	✓
Bassier and Vergauwen [30]	✓	×	✓	×	×
Pan et al. [34]	✓	✓	×	×	✓
Ours	✓	✓	✓	✓	✓

Table 19 compares the key features of the proposed method with recent methods in creating digital building models. This involves investigating various aspects and the potential contributions of the proposed methods in creating semantic digital models for the real world.

In this regard, most of the developed methods are capable of creating volumetric digital models of indoor spaces. However, the majority of these methods rely only on data-driven modeling approaches. Due to the complexity of indoor scenes and noise and outliers, data-driven methods encounter challenges in accurately representing geometric models and inferring and simulating topological relationships between elements. In addition, most developed methods for detecting and modeling door and window elements from point cloud data often rely on setting various threshold values (e.g., height, density, angle) and assumptions to identify potential door and window candidates. This approach necessitates manual calibration and adjustment, hindering the desired level of automation. Also, some techniques require supplementary data like camera sensor images or intensity values to accurately identify doors and windows with any state (open, semi-open, closed) within the scanned environment. Integrating these additional data streams complicates the computational process and adds to the overall cost of data acquisition.

Unlike purely data-driven approaches, our proposed hybrid bottom-up, top-down approach aligns the capabilities of AI methods in scene understanding along with the existing knowledge in the design and construction of buildings to create high-quality parametric digital building models with correct semantics and proper relationships between components. Thanks to the utilization of parametric modeling along with the optimization process, the proposed method enables overcoming obstructions and accurately estimating the model's parameters. Additionally, the proposed AI-based approach for detecting doors and windows from projected wall points' images can detect door and window elements in all three possible states: open, semi-open, and closed, without specific threshold values.

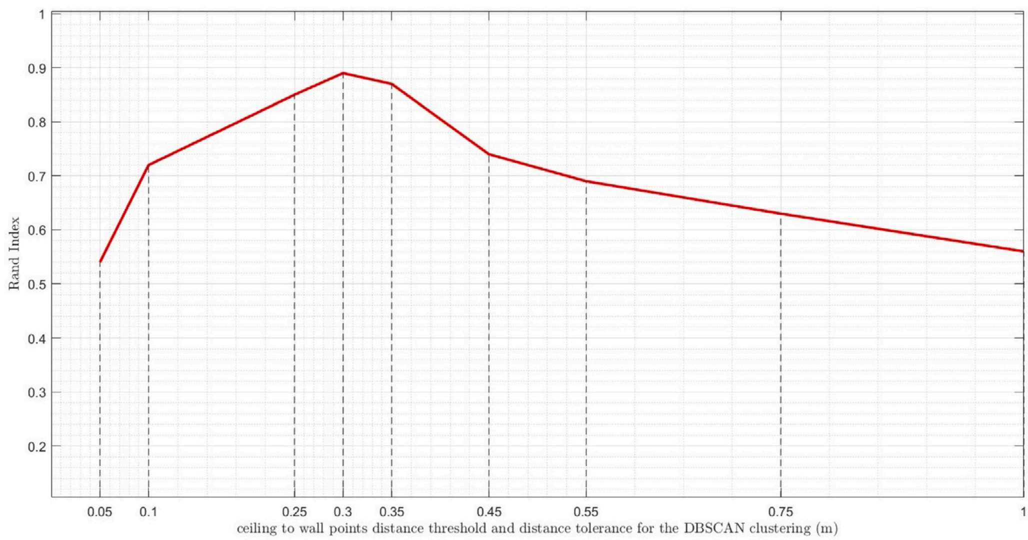


Fig. 14. Assessment of the impact of the different values for ceiling to wall points distance threshold and distance tolerance for the DBSCAN clustering parameters on the overall Rand Index similarity value in 3D space parsing task.

Table 20
Overview of the essential parameters employed during the creation of digital model using the proposed method.

Parameter	Value
Semantic enrichment:	
1. Ceiling to Wall points distance threshold (space parsing)	0.30 m
2. Distance tolerance for the DBSCAN clustering	0.30 m
3. Neighborhood distance for space adjacency graph	1 m
4. The number of NN points used to calculate the PCA coefficients	50
Door and window detection:	
5. Grid size for converting projected wall points into images	0.05 m

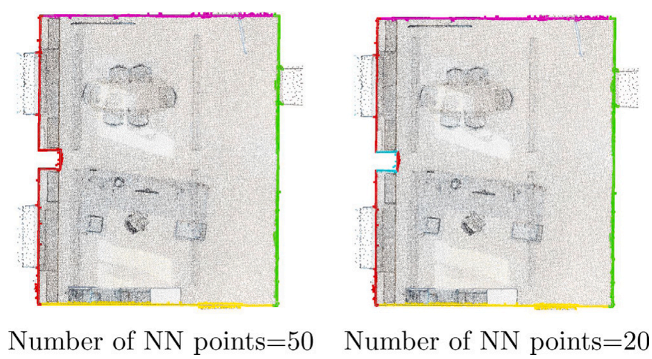


Fig. 15. Assessment of the impact of different values on the number of NN points on calculating the PCA coefficient to separate the wall footprints.

5.3. Sensitivity to the parameters

Table 20 reports the crucial parameters employed in creating the digital models using the proposed algorithm.

The ceiling-to-wall points distance and maximum distance tolerance for the DBSCAN clustering are crucial parameters utilized in the 3D space parsing algorithm. Setting these parameters to a default value of 30 cm for implementation was based on the average width of walls across all data. According to Fig. 14, deviating from this value by selecting a distance parameter less than or greater than 30 cm reduces RI accuracy due to issues like over-segmentation or merging of individual spaces. Additionally, the neighborhood distance for measuring the proximity of spaces was set to 1 m. Due to the minimum and maximum widths of 0.1 m and 0.7 m for internal dividing walls, the assigned value produces the best result for creating the adjacency graph of spaces.

As mentioned in Section 3.1.3., the proposed method employs the PCA technique to group the points of the closest wall surfaces with the same orientation. In this regard, the number of NN points for calculating the PCA coefficient was experimentally set to 50. Considering different values for the number of NNs can affect the level of geometric detail. Structural or architectural elements, such as columns and shaft panels, are installed along the walls in many buildings. Although these elements may resemble walls in appearance, their dimensional and geometrical sizes are significantly different. In this regard, considering only a range of about 20–30 neighboring points to calculate the PCA coefficients can result in separating these elements from the points on the wall surface (Fig. 15). As the main aim is to create the digital model of the building's structural elements and recognizing the geometric and architectural nature of these elements using point cloud data proves challenging, a high number of neighboring points is considered. This effectively integrates the small segments specific to these small elements with the points on the wall instances.

Despite the effectiveness of the developed method for modeling inclined wall instances, the proposed pipeline still faces challenges in accurately separating and modeling curved wall instances. In this regard, the proposed method for separating wall instances using PCA coefficients leads to the separation of the entire curved wall into individual wall patches with inaccurate orientation and location, resulting in the creation of extra wall instances (Fig. 16). Addressing these challenges requires a novel approach to separating the points belonging to curved wall surfaces and formulating the geometry of curved structures into the parametric modeling process.

To assess the impact of the grid size parameter on the overall accuracy in detecting and modeling door and window elements, a statistical analysis is conducted, testing different grid sizes of 5 cm, 7 cm, 10 cm, and 15 cm (Fig. 17).

According to Table 21, a grid size of 5 cm yielded the highest accuracy, with overall recall values of approximately 87% and 69% for detecting the door and window, respectively. In alternative scenarios with grid dimensions of 7 cm, 10 cm, and 15 cm, an increase in the grid size parameter leads to a decrease in the overall recall values and a subsequent increase in the mean error in estimating element parameters, reaching 20 cm (Fig. 18). Selecting a lower grid size value can result in the creation of more detailed images and subsequently

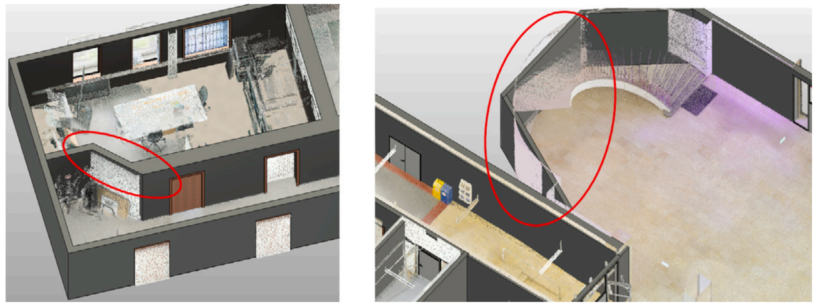


Fig. 16. Assessment of the effectiveness of the proposed method for separation of wall instances using the PCA coefficient on inclined and curved wall samples.

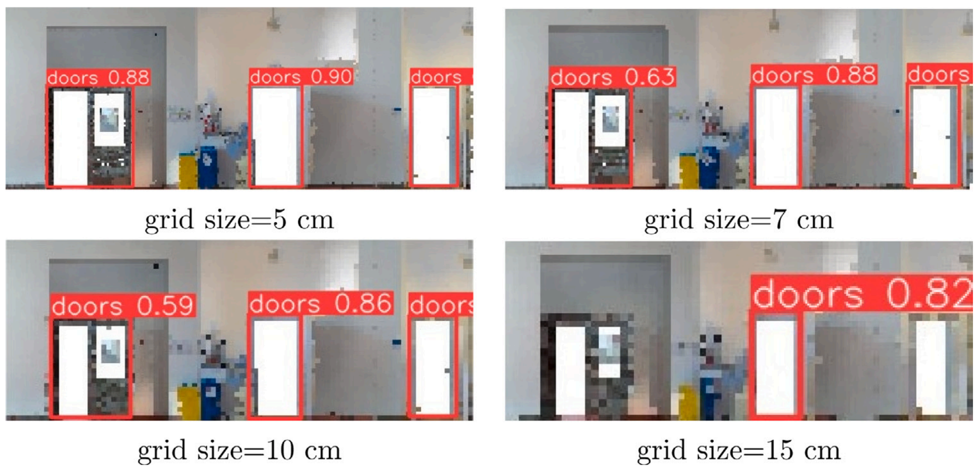


Fig. 17. The comparison between the results of door and window detection using projected wall point images with different grid sizes.

Table 21
Assessment of the impact of different grid size values used to convert the projected wall points to images on the accuracy of detection and modeling of door and window instances.

Grid size (cm)	5	7	10	15
Doors:				
precision (%)	0.98	0.96	0.96	0.96
recall (%)	0.87	0.81	0.77	0.65
Windows:				
precision (%)	0.99	0.99	0.99	0.99
recall (%)	0.69	0.56	0.50	0.47
Parameters:				
δ location (cm)	6.9	10.8	15.4	18.7
δ dimension (cm)	8.5	14.1	18.7	20.6

increase the accuracy of door and window detection. However, the grid size value is significantly related to the point density. Since the point density ranges from 2 to 9 in a sphere with a radius of one centimeter, selecting a 5 cm grid dimension improves the computational time cost and ensures that each grid cell contains at least two points for the sampling process.

5.4. Limitations

Despite diligent considerations, the proposed method's inherent limitations impact its efficacy in creating digital building models with coherent geometry. This section provides insights into the approach's limitations, highlighting specific areas where the proposed methodology may face challenges.

The developed algorithm utilizes the pre-trained PointTransformer semantic segmentation network to separate the main structural elements within the building point cloud. In this regard, the accuracy and efficiency of the labeling process are contingent upon the performance of the trained model, demanding a substantial volume of diverse annotated data and significant computational power for optimal training. Furthermore, a primary challenge in employing AI networks for the semantic segmentation of indoor point clouds involves effectively distinguishing surfaces such as walls, doors, and other elements constructed with glass or mirrors. In this case, the laser scanner beam does not accurately reflect off these surfaces, generating clutter. This, in turn, poses difficulties for the semantic segmentation network in accurately labeling points, particularly those corresponding to walls. These challenges subsequently impact the 3D space parsing step results and the subsequent separation of wall instances.

The proposed method leverages information from 3D space parsing and the corresponding ceiling boundary points to separate the footprints of wall instances. The process specifically concentrates on modeling external walls for individual spaces and modeling corresponding floor elements. This introduces challenges in accurately representing the digital model for buildings that incorporate openings or staircases within specific parts of their floors (Fig. 19).

The object detection network employed for door and window detection can only detect box-shaped elements. This limitation poses challenges in accurately detecting and parametric modeling of doors and window elements with different geometric appearances, such as circles or ellipses (Fig. 20). Addressing this issue necessitates the collection of numerous annotated image data containing elements with round shapes and the utilization of semantic segmentation networks for extracting their detailed geometry. Also, due to the similar appearance characteristics between door elements in the open state and inherent openings in wall elements, the network cannot distinguish between them, categorizing both as the door component. Furthermore, the collected library of parametric door and window elements encompasses instances with identical dimensional parameters but varying materials

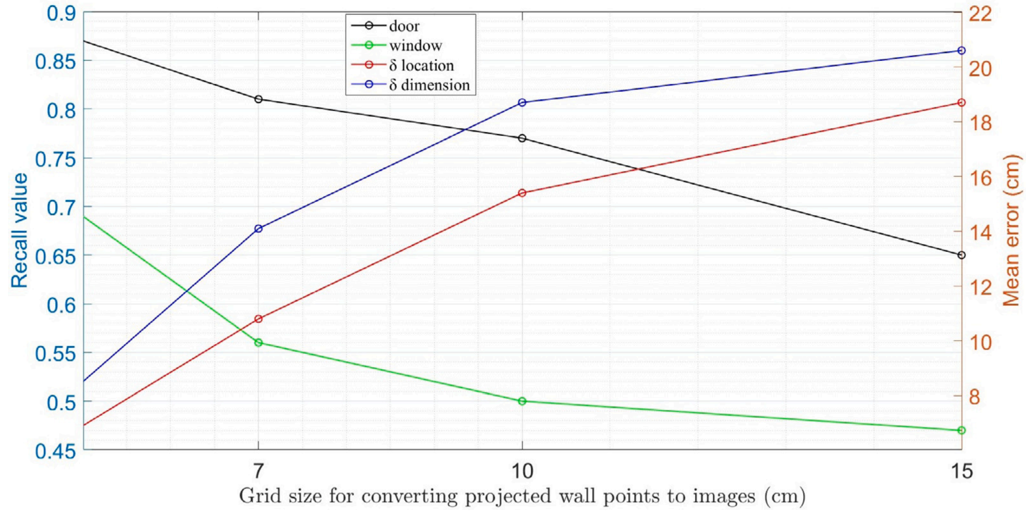


Fig. 18. The impact of different grid size values on the results of door and window detection. The door and window recall decreases with the increase of the grid size and the error in estimating location and dimension increases. This makes clear that the best value for the grid size parameter should be the lowest, in our case 5 cm.

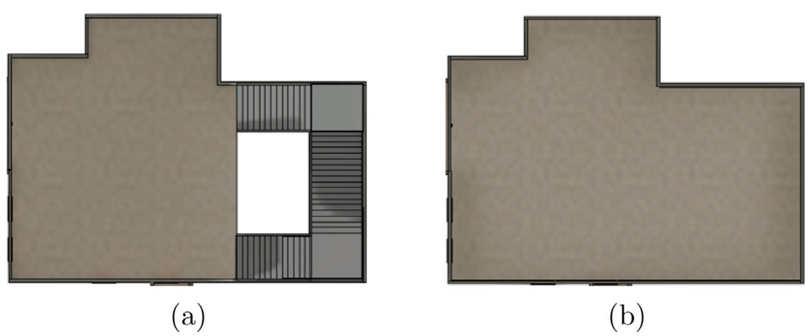


Fig. 19. Error in modeling the floor footprint for individual spaces using the proposed method: (a) reference digital building model, (b) the result of the proposed method for digital building model creation.

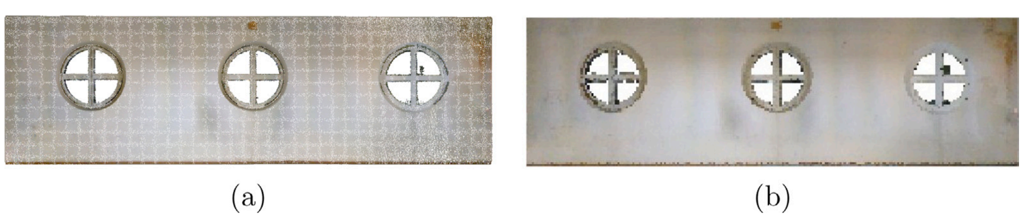


Fig. 20. The inability of door and window detection network to detect round-shaped windows: (a) projected wall points, (b) the result of door and window detection.

and frame designs. This challenges the proposed method in selecting appropriate instances that accurately represent both correct dimensional parameters and appearance properties.

6. Conclusions

This paper presents a novel hybrid bottom-up, top-down approach for the automatic creation of digital building models with rich semantics and coherent geometry from a dense laser scanner point cloud.

Unlike pure data-driven approaches, the proposed method leverages the advantages of parametric modeling processes to consider semantic relationships between components and formulate their interactions. The designed parameterized digital model is then fitted to the observed point cloud to estimate the best values for the parameters of elements. These not only enhance the geometric consistency of the digital model but also enable the overcoming of prevalent obstacles and challenges in complex building point clouds, such as noise and clutter.

The proposed method for the detection of doors and windows in different states and the subsequent model creation step integrates the capabilities of AI methods in object detection with domain engineering knowledge in design and construction. This eliminates the need for additional processes to combine data from different sensors and address their complex linkage issues, enhancing the integrity of element parameters in the resulting models.

The results of testing the proposed algorithm on six distinct indoor point cloud datasets demonstrate that the proposed approach can automatically generate digital building models with a mean absolute error of 7 cm in estimating the model parameters. These parameterized digital building models are editable, allowing for further refinement or enrichment to meet requirements and enhance decision-making for facility management, space management, and refurbishment purposes.

Despite the promising results that could signify significant progress in the field of 'Scan-to-BIM', the proposed method is not able to cover buildings constructed entirely using glass and mirror materials. Furthermore, the developed method for detecting doors and windows is only capable of detecting box-shaped elements, posing real challenges in the detection of doors and windows with round geometries.

In future works, the viability of the proposed method for creating digital building models will be examined across various real-world datasets. Also, the potential to enhance the level of development and semantic information of the resulting models will be explored by incorporating digital representations of other structural–architectural elements, such as staircases, columns, etc.

CRediT authorship contribution statement

Mansour Mehranfar: Writing – original draft, Visualization, Validation, Project administration, Methodology, Conceptualization. **Miguel A. Vega-Torres:** Writing – original draft, Visualization, Validation, Methodology, Data curation. **Alexander Braun:** Writing – review & editing, Validation, Supervision, Resources, Methodology. **André Borrmann:** Writing – review & editing, Validation, Supervision, Resources, Funding acquisition, Conceptualization.

Declaration of competing interest

Mansour Mehranfar reports financial support was provided by TUM Georg Nemetschek Institute Artificial Intelligence for the Built World (GNI). If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The point cloud data from the Technical University of Munich used in this research is publicly available through the link below: https://doi.org/10.14459/2024mp1742891.

Acknowledgments

The research presented has been performed in the frame of the AI4TWINNING project ("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), which is thankfully acknowledged.

References

[1] V. Stojanovic, M. Trapp, R. Richter, B. Hagedorn, J. Döllner, Towards the generation of digital twins for facility management based on 3D point clouds, in: 34th Annual Association of Researchers in Construction Management Conference, ARCOM 2018, Belfast, UK, 2018, pp. 270–279, URL https://www.arcom.ac.uk/-docs/proceedings/b65e593d342a8de045cf05698677e600.pdf. (last access 9 March 2024).

[2] N. Bortolini, M. Forcada, M. Macarulla, BIM for the integration of building maintenance management: A case study of a university campus, in: EWork and EBusiness in Architecture, Engineering and Construction: ECPPM 2016: Proceedings of the 11th European Conference on Product and Process Modelling, ECPPM, CRC Press, Limassol, Cyprus, 2016, pp. 427–434, http://dx.doi.org/10.1201/9781315386904.

[3] M. Kassem, G. Kelly, N. Dawood, M. Serginson, S. Lockley, BIM in facilities management applications: a case study of a large university complex, Built Environ. Proj. Asset Manag. 5 (3) (2015) 261–277, http://dx.doi.org/10.1108/BEPAM-02-2014-0011.

[4] O.C. Madubuike, C.J. Anumba, Digital twin–based health care facilities management, J. Comput. Civ. Eng. 37 (2) (2022) 04022057, http://dx.doi.org/10.1061/jccee5.cpeng-4842.

[5] Q. Wang, M.-K. Kim, Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018, Adv. Eng. Inform. 39 (2019) 306–319, http://dx.doi.org/10.1016/j.aei.2019.02.007.

[6] F. Xue, W. Lu, K. Chen, A. Zetkulic, From semantic segmentation to semantic registration: Derivative-free optimization-based approach for automatic generation of semantically rich as-built building information models from 3D point clouds, J. Comput. Civ. Eng. 33 (4) (2019) 04019024, http://dx.doi.org/10.1061/(ASCE)CP.1943-5487.0000839.

[7] J. Abualdenien, A. Borrmann, Levels of detail, development, definition, and information need: a critical literature review, J. Inf. Technol. Constr. 27 (2022) 363–392, http://dx.doi.org/10.36680/j.itcon.2022.018.

[8] M. Mehranfar, A. Braun, A. Borrmann, A hybrid top-down, bottom-up approach for 3D space parsing using dense RGB point clouds, in: Proceedings of 14th European Conference on Product & Process Modelling, 2022, pp. 551–558, URL https://mediatum.ub.tum.de/download/1688614/1688614.pdf.

[9] M. Mehranfar, A. Braun, A. Borrmann, Automatic creation of digital building twins with rich semantics from dense RGB point clouds through semantic segmentation and model fitting, in: Proceedings of the 30th International Conference on Intelligent Computing in Engineering, EG-ICE, 2023, URL https://mediatum.ub.tum.de/download/1712296/1712296.pdf.

[10] Z.C. Marton, R.B. Rusu, M. Beetz, On fast surface reconstruction methods for large and noisy point clouds, in: Proceedings of IEEE International Conference on Robotics and Automation, ICRA, IEEE, Kobe, Japan, 2009, pp. 3218–3223, http://dx.doi.org/10.1109/ROBOT.2009.5152628.

[11] A.-V. Vo, L. Truong-Hong, D.F. Laefer, M. Bertolotto, Octree-based region growing for point cloud segmentation, ISPRS J. Photogramm. Remote Sens. 104 (2015) 88–100, http://dx.doi.org/10.1016/j.isprsjprs.2015.01.011.

[12] F. Poux, C. Mattes, Z. Selman, L. Kobbelt, Automatic region-growing system for the segmentation of large point clouds, Autom. Constr. 138 (2022) 104250, http://dx.doi.org/10.1016/j.autcon.2022.104250.

[13] X. Xiong, A. Adan, B. Akinci, D. Huber, Automatic creation of semantically rich 3D building models from laser scanner data, Autom. Constr. 31 (2013) 325–337, http://dx.doi.org/10.1016/j.autcon.2012.10.006.

[14] A. Dimitrov, M. Golparvar-Fard, Segmentation of building point cloud models including detailed architectural/structural features and MEP systems, Autom. Constr. 51 (2015) 32–45, http://dx.doi.org/10.1016/j.autcon.2014.12.015.

[15] B. Xiong, Y. Jin, F. Li, Y. Chen, Y. Zou, Z. Zhou, Knowledge-driven inference for automatic reconstruction of indoor detailed as-built BIMs from laser scanning data, Autom. Constr. 156 (2023) 105097, http://dx.doi.org/10.1016/j.autcon.2023.105097.

[16] M.A. Vega Torres, A. Braun, F. Noichl, A. Borrmann, H. Bauer, D. Wohlfeld, Recognition of temporary vertical objects in large point clouds of construction sites, Smart Infrastruct. Construct. 174 (4) (2021) 134–149, http://dx.doi.org/10.1680/jsmic.21.00033.

[17] Q. Qiu, M. Wang, J. Guo, Z. Liu, Q. Wang, An adaptive down-sampling method of laser scan data for scan-to-BIM, Autom. Constr. 135 (2022) 104135, http://dx.doi.org/10.1016/j.autcon.2022.104135.

[18] H. Macher, T. Landes, P. Grussenmeyer, Point clouds segmentation as base for as-built BIM creation, ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. II-5/W3 (2015) 191–197, http://dx.doi.org/10.5194/isprsannals-II-5-W3-191-2015.

[19] S. Nikoohemat, A.A. Diakité, S. Zlatanova, G. Vosselman, Indoor 3D reconstruction from point clouds for optimal routing in complex buildings to support disaster management, Autom. Constr. 113 (2020) 103109, http://dx.doi.org/10.1016/j.autcon.2020.103109.

[20] M. Arikan, M. Schwärzler, S. Flöry, M. Wimmer, S. Maierhofer, O-snap: Optimization-based snapping for modeling architecture, ACM Trans. Graph. 32 (1) (2013) 1–15, http://dx.doi.org/10.1145/2421636.2421642.

[21] R.-C. Dumitru, D. Borrmann, A. Nüchter, Interior reconstruction using the 3D hough transform, Int. Arch. Photogramm. Rem. Sens. Spatial Inf. Sci. XL-5/W1 (2013) 65–72, http://dx.doi.org/10.5194/isprsarchives-XL-5-W1-65-2013.

[22] C. Mura, O. Mattausch, A. Jaspe-Villanueva, E. Gobbetti, R. Pajarola, Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts, Comput. Graph. (2014).

[23] S. Ochmann, R. Vock, R. Wessel, R. Klein, Automatic reconstruction of parametric building models from indoor point clouds, Comput. Graph. (2016).

[24] S. Ochmann, R. Vock, R. Klein, Automatic reconstruction of fully volumetric 3D building models from oriented point clouds, ISPRS J. Photogramm. Remote Sens. 151 (2019) 251–262, http://dx.doi.org/10.1016/j.isprsjprs.2019.03.017.

[25] H. Tran, K. Khoshelham, Procedural reconstruction of 3D indoor models from Lidar data using reversible jump Markov chain Monte Carlo, Remote Sens. 12 (5) (2020) 838, http://dx.doi.org/10.3390/rs12050838.

[26] A. Abdollahi, H. Arefi, S. Malihi, M. Maboudi, Progressive Model-Driven Approach for 3D Modeling of Indoor Spaces, Sensors 23 (13) (2023) 5934, http://dx.doi.org/10.3390/s23135934.

[27] J. Park, Y. Cho, Point Cloud Information Modeling: Deep Learning–Based Automated Information Modeling Framework for Point Cloud Data, J. Construct. Eng. Manag. 148 (2) (2021) http://dx.doi.org/10.1061/(ASCE)CO.1943-7862.0002227.

[28] Y. Perez-Perez, M. Golparvar-Fard, K. El-Rayes, Scan2BIM-NET: Deep learning method for segmentation of point clouds for scan-to-BIM, J. Construct. Eng. Manag. 147 (9) (2021) 04021107, http://dx.doi.org/10.1061/(ASCE)CO.1943-7862.0002132.

[29] Y. Pan, A. Braun, A. Borrmann, I. Brilakis, Enriching geometric digital twins of buildings with small objects by fusing laser scanning and AI-based image recognition, Autom. Constr. 140 (2022) 104375, http://dx.doi.org/10.1016/j.autcon.2022.104375.

[30] M. Bassier, M. Vergauwen, Topology reconstruction of BIM wall objects from point cloud data, Remote Sens. 12 (11) (2022) 1800, http://dx.doi.org/10.3390/rs12111800.

[31] J. Park, J. Kim, D. Lee, K. Jeong, J. Lee, H. Kim, T. Hong, Deep learning–based automation of scan-to-BIM with modeling objects from occluded point clouds, J. Manage. Eng. 38 (4) (2022) 04022025, http://dx.doi.org/10.1061/(ASCE)ME.1943-5479.0001055.

[32] S. Tang, X. Li, X. Zheng, B. Wu, W. Wang, Y. Zhang, BIM generation from 3D point clouds by combining 3D deep learning and improved morphological approach, Autom. Constr. 141 (2022) 104422, http://dx.doi.org/10.1016/j.autcon.2022.104422.

[33] Z. Xiang, A. Rashidi, G. Ou, Integrating inverse photogrammetry and a deep learning–based point cloud segmentation approach for automated generation of BIM models, J. Construct. Eng. Manag. 149 (9) (2023) 04023074, http://dx.doi.org/10.1061/JCEMD4.COENG-13020.

[34] Y. Pan, A. Braun, A. Borrmann, I. Brilakis, 3D deep-learning-enhanced void-growing approach in creating geometric digital twins of buildings, Smart Infrastruct. Construct. 176 (1) (2023) 24–40, http://dx.doi.org/10.1680/jsmic.21.00035.

[35] K. Maximilian, B. Stahl, R. Alexander, Reconstructing geometrical models of indoor environments based on point clouds, Remote Sens. 15 (18) (2023) 4421, http://dx.doi.org/10.3390/rs15184421.

[36] S. Pu, G. Vosselman, Knowledge based reconstruction of building models from terrestrial laser scanning data, ISPRS J. Photogramm. Remote Sens. 64 (6) (2009) 575–584, http://dx.doi.org/10.1016/j.isprsjprs.2009.04.001.

[37] B. Cheng, S. Chen, L. Fan, Y. Li, Y. Cai, Z. Liu, Windows and doors extraction from point cloud data combining semantic features and material characteristics, Buildings 13 (2) (2023) http://dx.doi.org/10.3390/buildings13020507.

[38] B. Quintana Galera, S. Prieto, A. Adan, F. Bosché, Door detection in 3D coloured point clouds of indoor environments, Autom. Constr. 85 (2018) 146–166, http://dx.doi.org/10.1016/j.autcon.2017.10.016.

[39] T. Vu, K. Kim, T.M. Luu, T. Nguyen, C.D. Yoo, SoftGroup for 3D instance segmentation on point clouds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2022, pp. 2708–2717, http://dx.doi.org/10.48550/arXiv.2203.01509.

[40] M. Zhong, X. Chen, C. Xiaokang, G. Zeng, Y. Wang, MaskGroup: Hierarchical Point Grouping and Masking for 3D Instance Segmentation, in: Proceedings of IEEE International Conference on Multimedia and Expo, ICME, 2022, pp. 1–6, http://dx.doi.org/10.1109/ICME52920.2022.9859996.

[41] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, A. Markham, Randla-net: Efficient semantic segmentation of large-scale point clouds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, Seattle, WA, USA, 2020, pp. 11108–11117, http://dx.doi.org/10.1109/CVPR42600.2020.01112.

[42] H. Thomas, C.R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, L.J. Guibas, KPConv: Flexible and deformable convolution for point clouds, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, IEEE, Seoul, Korea (South), 2019, pp. 6411–6420, http://dx.doi.org/10.1109/ICCV.2019.00651.

[43] V.L. Rvachev, An analytic description of certain geometric objects, Dokl. Akad. Nauk 153 (4) (1963) 765–767, http://mi.mathnet.ru/dan29454.

[44] K.J. Weiler, Topological Structures For Geometric Modeling (Boundary Representation, Manifold, Radial Edge Structure) (Ph.D. thesis), Rensselaer Polytechnic Institute, 1986, URL https://api.semanticscholar.org/CorpusID:115556230.

[45] A.K. Patil, P. Holi, S.K. Lee, Y.H. Chai, An adaptive approach for the reconstruction and modeling of as-built 3D pipelines from point clouds, Autom. Construct. 75 (2017) 65–78, http://dx.doi.org/10.1016/j.autcon.2016.12.002.

[46] R. Sacks, C. Eastman, G. Lee, Parametric 3D modeling in building construction with examples from precast concrete, Autom. Constr. 13 (2004) 291–312, http://dx.doi.org/10.1016/S0926-5805(03)00043-8.

[47] M. Mafipour, S. Vilgertshofer, A. Borrmann, Automated geometric digital twinning of bridges from segmented point clouds by parametric prototype models, Autom. Construct. 156 (2023) 105101, http://dx.doi.org/10.1016/j.autcon.2023.105101.

[48] X. Liu, X. Wang, G. Wright, J. Cheng, X. Li, R. Liu, A state-of-the-art review on the integration of building information modeling (BIM) and geographic information system (GIS), ISPRS Int. J. Geo-Inf. 6 (2) (2017) 53, http://dx.doi.org/10.3390/ijgi6020053.

[49] M. Elsayed, O. Tolba, A. El Antably, Architectural space planning using parametric modeling, in: Proceedings of the 8th Arab Society for Computer Aided Architectural Design Conference, ASCAAD, London, UK, 2016, p. 45, URL https://papers.cumincad.org/data/works/att/ascaad2016_007.pdf. (last access 9 March 2024).

[50] M. Keshavarzi, M. Rahmani-Asl, GenFloor: Interactive generative space layout system via encoded tree graphs, Front. Archit. Res. 10 (4) (2021) 771–786, http://dx.doi.org/10.1016/j.foar.2021.07.003.

[51] H. Zhao, L. Jiang, J. Jia, P.H. Torr, V. Koltun, Point transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, IEEE, Montreal, QC, Canada, 2021, pp. 16259–16268, http://dx.doi.org/10.1109/ICCV48922.2021.01595.

[52] I. Armeni, O. Sener, A.R. Zamir, H. Jiang, I. Brilakis, M. Fischer, S. Savarese, 3D Semantic Parsing of Large-Scale Indoor Spaces, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, IEEE, Las Vegas, NV, USA, 2016, pp. 1534–1543, http://dx.doi.org/10.1109/CVPR.2016.170.

[53] T. Czerniawski, B. Sankaran, M. Nahangi, C. Haas, F. Leite, 6D DBSCAN-based segmentation of building point clouds for planar object classification, Autom. Constr. 88 (2018) 44–58, http://dx.doi.org/10.1016/j.autcon.2017.12.029.

[54] X. Cao, B.-Z. Qiu, G. Xu, BorderShift: toward optimal MeanShift vector for cluster boundary detection in high-dimensional data, Pattern Anal. Appl. 22 (2019) http://dx.doi.org/10.1007/s10044-018-0709-0.

[55] J. Nelder, R. Mead, A simplex method for function minimization, Comput. J. 7 (4) (1965) 308–313, http://dx.doi.org/10.1093/comjnl/7.4.308.

[56] Y. Wu, L. Maosu, F. Xue, Towards fully automatic Scan-to-BIM: A prototype method integrating deep neural networks and architectonic grammar, in: Proceedings of European Conference on Computing in Construction (EC3) and the 40th International CIB W78 Conference, Vol. 4, European Council on Computing in Construction, Heraklion, Greece, 2023, http://dx.doi.org/10.35490/EC3.2023.257.

[57] G. Jocher, A. Chaurasia, J. Qiu, Ultralytics YOLO, 2023, URL https://github.com/ultralytics/ultralytics.

[58] J. Solawetz, Francesco, What is YOLOv8? The ultimate guide, 2023, https://blog.roboflow.com/whats-new-in-yolov8/. (Accessed 11 January 2023).

[59] W.M. Rand, Objective criteria for the evaluation of clustering methods, J. Amer. Statist. Assoc. 66 (336) (1971) 846–850, http://dx.doi.org/10.2307/2284239.