

BIM generation from 3D point clouds by combining 3D deep learning and improved morphological approach



Shengjun Tang ^a, Xiaoming Li ^a, Xianwei Zheng ^b, Bo Wu ^c, Weixi Wang ^{a,*}, Yunjie Zhang ^{d,**}

^a School of Architecture and Urban Planning, Research Institute for Smart Cities & Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Natural Resources, Shenzhen University, PR China

^b The State Key Lab., LIESMARS, Wuhan University, Wuhan, PR China

^c Department of Land Surveying and Geo-Informatics, Hong Kong Polytechnic University, Hong Kong, PR China

^d Department of Architecture and Civil Engineering, City University of Hong Kong, Hong Kong, PR China

ARTICLE INFO

Keywords:

Indoor modeling
Floorplan generation
Markov random field
Parametric modeling
Building information modeling (BIM)

ABSTRACT

The demand for the models of BIM has increased due to the numerous applications in emergency response and location-based services. Creating a semantically rich 3D interior model has traditionally been a very manual process. Although the latest LiDAR scanning and photogrammetry techniques efficiently capture the existing building, the automatic reconstruction of complete volumetric and functional interior models is still an intensive and challenging research topic. This paper presents a novel parametric modeling method for reconstructing semantic volumetric building interiors from the unstructured point cloud of a building. Unlike existing partitioning-based methods, our proposed method overcomes the limitations by providing a flexible framework for combining 3D Deep Learning and an improved morphological approach for inverse BIM modeling. By employing a flexible and robust three-step mechanism, the proposed method first classifies the point cloud into thirteen types using a 3D Deep Learning method, then extracts the surfaces and generates the initial feature space. In regularizing the space and modeling BIM, the space boundaries are optimized by formulating the subordination between the grid cells generated by walls and the segmented boundary shapes generated by the gridded floor plan through energy minimization. Then, a grammar-enhanced point-line polygon and parametric description are designed to generate the final BIM model. By using a robust space partitioning and optimization method, three main goals are achieved: recovery of geometric and semantic information, robustness in different data sources (especially in non-Manhattan scenes), parametric BIM modeling in our proposed method. The paper demonstrates the potential of our algorithms for both RGB-D and LiDAR point clouds acquired from scenes in Manhattan and outside Manhattan. Experimental results show that this method is capable of automatically generating a geometrically and semantically consistent BIM model that is competitive with the existing method in terms of geometric accuracy and model completeness.

1. Introduction

In recent years, the demand for detailed and accurate three-dimensional (3D) models of interiors has increased in areas such as indoor mapping and navigation, simulation, intelligent building management, and virtual reality [17,24,47]. Many researchers recover the interior structure from the architectural floor plans by recognizing the spaces, openings, and object arrangements. Unfortunately, a large number of buildings do not have detailed floorplans. With the significant revolution in 3D data acquisition techniques, a variety of sensors such as

monocular cameras, RGB-D sensors, and Light Detection and Ranging (LiDAR) can be used for 3D mapping of interior spaces with mobile mapping technology. However, the unstructured raw data is difficult to use in practical applications, and the key technical challenge is to reconstruct a geometric and semantically rich building information model (BIM) for an indoor scene [9,13,28]. Many efforts have been made to automatically develop BIM with these immediate survey datasets, such as a 3D point cloud or 3D mesh model with specific algorithms [1,18,33,43,48]. However, the automated generation of the BIM model of indoor environments is challenged by the inherent noise and

* Corresponding author at: School of Architecture and Urban Planning, Research Institute for Smart Cities, Shenzhen University, PR China.

** Corresponding author.

E-mail addresses: weixiwang@szu.edu.cn (W. Wang), yunjzhang4-c@my.cityu.edu.hk (Y. Zhang).

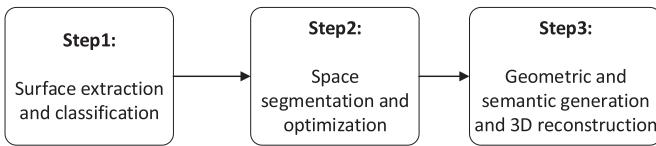


Fig. 1. A general framework for indoor 3D modeling.

incompleteness of the data and requires further investigation [45].

As shown in Fig. 1, 3D reconstruction of indoor scenes based on point cloud data usually consists of three steps. First, most interior structures can be fitted with continuous planes or other regular shapes. These geometric primitives are extracted from 3D point cloud with traditional plane fitting algorithm [12,14,40], or by learning approaches [9,10,28]. Second, these geometric primitives will be assembled to generate watertight boundaries of either the whole scenes or separate rooms [12,45]. The optimization framework [9,28] or 2D space partition approach [12,32,45] are employed in this step, where the former is more sensitive to data missing problems, while the latter is typically more robust. Third, some other indoor components such as doors and windows will be detected with a geometric approach [43], or deep learning method [10,20,21] thus the relations among these components are recovered. The framework considers the task of reconstruction problem by combining deep learning and the shape grammar constraints, which enhance the robustness during the primitive detection and classification process. The proposed method first classifies the whole scene into thirteen types based on 3D deep learning to facilitate the extraction of geometric primitives. Traditionally, the planes that are not part of the walls, floors, and ceilings, such as the furniture, window, and doors, will be filtered out based on specific rules such as the size of the plane, and the distance from the plane to the floor or ceiling, etc. However, it is difficult to apply the same rules to different interior scenarios due to their different interiors. Unlike the surface-based segmentation method, the deep learning approach uses a combination of color, shape, point cloud density, and other features to train a predictive model. The model can effectively distinguish between building structural components and furniture, which will provide pre-processed data for further plane segmentation and significantly enhance the robustness during structure components detection. Then, a hybrid surface extraction method is used for plane detection, and these planes are optimized by normal and distance constraints.

Meanwhile, the raw point cloud and the vertical planes are projected on the ground surface to generate a 3D map, and an improved morphological algorithm is used for space decomposition. Finally, the results of 2D space decomposition would be regularized by solving a Markov Random Field (MRF) problem. It should be noted that three main objectives need to be achieved in our proposed method. First, in terms of geometric accuracy, the reconstructed BIM model should retain as much structural details as possible, such as columns and curved walls. The geometric errors of the final model should be as minor as possible. Second, the semantic information in the output model should be accurate. As both component properties and their related information need to be preserved. Third, the proposed method needs to be robust in different indoor scenarios or data sources, particularly non-Manhattan scenes. In addition, the spatial segmentation algorithm should still work well without deep learning classification.

This paper proceeds with a literature review of BIM reconstruction methods in Section 2, followed by a detailed description of our proposed indoor components extraction and automatic BIM reconstruction method in Section 3. Section 3 also outlines the implementation of our BIM reconstruction based on a well-designed parametric data structure. Section 4 validates the feasibility of the proposed method in real-world indoor environments and public data sets. Finally, prospective future work and the paper's conclusion are presented in Section 5.

2. Related work

Traditionally, creating a semantically rich 3D indoor model is generally a manual process. The manual construction of as-built BIMs is time-consuming, labor-intensive, tedious, and subjective, and requires skilled workers [12,16,42]. To accelerate data processing and improve the modeling accuracy, many attempts have been made to automate 3D model creation from 2D drawings or 3D point clouds in the GIS and architecture, engineering, and construction domains [1,35,43–45]. Recent approaches to indoor reconstruction can be classified into three main categories:(1) surface-based approaches, (2) grammar-based approaches, and (3) learning-based approaches. Our review of previous works will be carried out in the above three areas.

2.1. Surface-based approaches

This approach starts with basic structured elements such as walls, floors, etc., based on geometric features of the indoor scenes, and then divides the interior scene into separate spaces. The local data features such as point density, normal, and dimension among a local neighborhood were often used for primitive extraction. Sanchez and Zakhor [39] presented the indoor scene with an arrangement of planar primitives, which are extracted by using the plane fitting algorithm. Based on the surface segmentation method, Xiong et al. [49] implemented an automatic indoor reconstruction method to transform a point cloud data into a structural model, in which the semantic information of these extracted surfaces is recovered by using their shape-grammar or contextual relationships. They also performed a detailed analysis of the recognized surfaces to locate openings, such as windows and doorways. However, the proposed methodology requires a commitment to the wall class before openings are detected, which may result in errors during doors and windows detection. Macher et al. [30] proposed a semi-automatic indoor reconstruction method in which the recognized components were first exported into an OBJ file and subsequently transformed into an IFC file with a post-processing step. In this approach, a building is described by planar shapes. Thus complex architectures composed of non-planar shapes are set aside. This method cannot integrate information such as materials into the BIM model and is also, for now, not applicable to all point clouds. A similar strategy was proposed by Tang et al. [43], who combined the planar features, contextual constraints, and relations between components to enable the automatic reconstruction of indoor environments. Some researchers have also explored methods for generating expert BIM models based on a Scan-vs- BIM framework. Bosché et al. [6] presents a novel framework that integrates scan-to BIM and Scan-vs- BIM techniques to achieve a robust automated comparison between the as-builds and as-planned cylindrical structures in mechanical engineering, plumbing and electrical (MEP), where the object detection and recognition [5,7,8] are uniquely integrated into a Scan-vs- BIM framework that derives the types of objects from proximity analysis. Subsequently, the similar framework is extended to the overall reconstruction of internal structures and MEP components Valero et al. [46] proposed a semi-automatic scan-to-BIM approach to generate semantically rich BIM modeling in IFC format. In this method, three steps, including structural Scan-to-BIM, MEP Scan-to- BIM, and Scan-to-BIM editor, are involved for different purposes. In particular, the method fully integrates geometric features of point clouds and visual images for extraction and modeling for various interior components. However, there are still two problems that need to be solved. First, the reconstruction of walls is based on the Manhattan hypothesis and planar matching theory, which may result in walls without matching planes that cannot be reconstructed and need to be manually processed. Second, the hole-based method for detecting openings may be compromised by occlusion or inaccurate wall detection. The above methods all perform spatial segmentation based on local features or their relationships. A watertight space geometry can be generated directly from the point cloud and each space can be reconstructed individually. In

contrast, there are methods that focus on extracting wall elements, where a topology rule is established for wall connections [3,30,31]. Bassier and Vergauwen [3] proposed an unsupervised method for BIM wall reconstruction. In this method, the point cloud of the building is first segmented into planar segments by a region-growing method, and then four connection types, including intersecting, orthogonal, blended and direct connections, are used to recover the complete wall structure. The advantage of this method is that most wall elements can be quickly identified and recovered a BIM model. However, this method can only reconstruct the wall components, not the space (important for interior applications), the relationships between the components (essential for topological recovery), and the openings. In addition, this method ignores a large number of details when extracting the planes, so small interior elements such as columns and corners cannot be detected, which can lead to degradation of the final BIM model. Most of the above methods are mainly designed for the Manhattan building. Besides, there are still some problems that should be addressed, such as improving the robustness to occlusions, extracting other interior components, and creating expert models BIM and so on.

2.2. Grammar-based approaches

Some researchers have explored integrating architectural design rules into 3D reconstruction methods. These building rules can be collectively referred to as grammatical information, a powerful constraint during model reconstruction. In general, the primitives obtained by surface detection are integrated into the reconstruction process, and then the grammar information is used to constrain structure generation [4,36]. Khoshelham and Diaz-Vilarino [25] first proposed a shape grammar to reconstruct 3D interior spaces of "Manhattan-World" buildings from point clouds. Two rules are used in the reconstruction process. The first rule is the absolute orientation of the main walls and the floors/ceilings. The second rule connects neighboring cuboids that are not separated by an interior wall. However, the algorithm uses only two simple rules and cannot be used in non-Manhattan scenes. The Manhattan building means that all boundaries of the structural elements, such as walls, columns, and windows, have either vertical or horizontal orientation and are perpendicular to each other. All other types are non-Manhattan buildings, such as the building walls with curved structures, triangular spaces, etc. Tran and Khoshelham [45] extended the method and proposed an automatic indoor reconstruction method by combining a data-driven process and a procedural reconstruction process. In this approach, reversible-jump Markov Chain Monte Carlo (rjMCMC) guides the automated application of grammar rules in deriving a 3D indoor model. Another way of using grammar information for building reconstruction is to build the structure graph of each element and then segment rooms with heuristics [23], in which the grammar rules are applied to recover a structure graph together with the geometries. For each rule application, an individual geometric representation and a reconstruction algorithm are involved and used for the task. However, the main limitation of the method is the poor accuracy of object recognition and that it cannot be used in non-Manhattan scenes. Another intuitive solution is to connect the detected isolated structure elements to form a global optimization function with specific grammars. Reference [34] describes the global optimization of an indoor model structure in which a graph-cut formulation was used to label the cells inside and outside of space and reconstruct a watertight surface model, rather than the utilization of local features and contextual constraints. Ochmann et al. [33] presented an automatic reconstruction of the parametric and volumetric model, in which the globally plausible connectivity of all elements was considered, and the solution was subsequently improved by transforming the indoor reconstruction task to a linear integer programming problem [32]. However, this method cannot recover the non-planar primitives and does not extract the window and door elements. Similarly, Fang et al. [12] proposed a floor plan generation method with a space partition idea. The space is decomposed into

the polygonal partition, and subordinate relationships between their edges and the wall components were determined by an energy function. However, the algorithm still can not detect the higher-order geometric primitives and the openings in the building interior. Most of these methods mentioned above are mainly designed for the Manhattan building. These structure models do not contain the furniture or opening components, which is significant during navigation or emergency evacuation. Besides, the methods mentioned above generally generate a surface model for the building interior instead of a BIM model, which plays a significant role in some expert applications.

2.3. Learning-based approaches

With the development of deep learning, many researchers explore the new indoor reconstruction methods assisted by object segmentation with deep learning. In early research, the DNN (Deep Neural Networks) detectors were used for the detection of the primitives such as corners and then recovered wall or room boundary information by constructing graph models with those low-level primitives [29]. The method proposed by Liu et al. [29] used a 2D image learning-based method to convert a rasterized floorplan image into a vector-graphics representation. The junctions such as doors and corners of the interior building were first detected to represent those low-level geometries. Then a topologically and geometrically consistent floor plan was generated by aggregating the junctions with an integer programming approach (IP). At the same time, the approach is designed with the Manhattan hypothesis of the interior structure. Subsequently, they improved the method by combining DNN and IP methods [9,28]. However, it can also fail to construct a complete indoor model in the conditions of corners missing, false candidate primitives, and still needs to restrict the solution space to Manhattan scenes. Furthermore, they rely on a post-processing step performed on image coordinates, which will likely lead to a miss-alignment between the final floorplan. Meanwhile, since the accuracy of this method is affected by the resolution of the projected image, it is easy to fail to extract the boundaries in the sparse region of the original point cloud. Instead of detecting the corners of the building interior, Lee et al. [27] proposed an end-to-end trainable encoder-decoder network to predict the locations of the room layout, called "RoomNet". 11 room layouts under the "Manhattan-World" assumption were used for the model training. Once the trained model predicts those keypoint locations with an associated room type, those points will be connected to produce a boxy room layout representation.

With the development of computer vision tasks, such as layout estimation and object detection, increasing attention from academia and industry has focused on building indoor semantic 3D models with the end-to-end trainable encoder-decoder network by using monocular images or low-cost RGB-D data. Unlike the traditional indoor structure reconstruction method, this semantic-assisted reconstruction method segmented the image semantically and predicted each element's orientation and bounding box. Then These methods modeled indoor objects with appearance-similar shapes and placed the retrieved models in a layout cuboid with alignment to the scene depicted in the images. Gupta et al. [15] proposed an Align3D method which first detected and segmented object instances in the scene and then used a convolutional neural network (CNN) to predict the pose of the object. Furthermore, the prototypical models will be placed into the scene with the model that fits best. However, this method only models indoor tables and chairs. The final indoor model has some differences from the real environment. Huang et al. [22] combined a Global Geometry Network (GGN) and a local object network (LON) for end-to-end scene recovery, where the GGN estimated a 3D layout cuboid and the related camera pose. At the same time, the LON learned 3D object bounding boxes along with object poses. Instead of using single-view camera data, RevealNet is proposed by Hou et al. [19] to reconstruct an indoor semantic model with scans generated from RGB-D video sequences, in which two sub-tasks including 3D object detection and point-based object mesh prediction

is employed for model generation. Likewise, Huan et al. [21] proposes an end-to-end multi-task neural network for geometry-enhanced semantic 3D reconstruction of RGB-D indoor scenes. A geometric extractor network is used to estimate the room layout, the camera pose, and the 3D object bounding box. However, although the method can identify indoor components well, it still ends up with a triangular mesh model rather than vectorized model such as cityGML or IFC models. Moreover, those learning-based methods could still be used in small rooms with Manhattan structures only and hard to use for large-scale scene reconstruction.

Nevertheless, note that the surface or grammar-based approach has better robustness in the extraction of planar structures and can recover the topology better by combining grammar information. However, this method has some limitations for the extraction of curved structures or other components such as doors, windows, etc. While deep learning-based approaches can achieve semantic segmentation with high accuracy, but have significant drawbacks in geometric reconstruction and can only be used for single room reconstruction with specific structures. Inspired by the surface-based method and 3D deep learning method [12,20,32], our proposed method overcomes the limitations of previous approaches by providing a flexible framework for the combination of DNN and improved morphological approach for inverse BIM modeling. In terms of geometric reconstruction, our proposed method in this paper can better identify small building components, such as columns, which are generally not reconstructed in other methods. Therefore, the model obtained by this method has higher geometric accuracy. In terms of semantic information, the deep learning method is involved for object detection in our method, which makes it possible to reconstruct windows, doors, or furniture. This is rarely addressed in other methods. Besides, the state-of-art indoor 3D reconstruction method generates either a surface model, such as CityGML, Indoor GML, OBJ model, or a volumetric model such as IFC. The generation of an expert BIM model for the as-built building was rarely investigated, which can avoid information loss problems during data exchange and is directly usable in expert BIM applications. Table 1 summarizes and compares key features of the state-of-art approaches and the proposed approach in this paper. The main contributions of the proposed BIM reconstruction method are as follows.

1) A fusion strategy by integrating deep learning method and an improved morphological algorithm is proposed for indoor modeling, which can effectively improve the robustness and accuracy of the detection and reconstruction of the components, especially doors and windows.

2) The space regularization is solved by modeling the intersection problem as a Markov Random Field energy function, which enables a more accurate overlay analysis between regular vector boundaries and irregular raster boundaries.

3) We design a grammar-extended point-line-polygon and parametric description to represent their geometry, types, function, material, and relations, which can be directly imported into expert BIM software and generate a full volumetric and functional BIM model.

3. Methodology

3.1. Overview

The entire pipeline starts from a colorized or gray-level point cloud collected by either RGB-D cameras or LiDAR sensors. The input of our approach is a 3D indoor point cloud with an oriented normal whose “up” direction is assumed to be the z-axis. If the normal of points are not yet available, they will be estimated by a local Principal Component Analysis (PCA) algorithm [37]. This approach is designed for general scenes with arbitrary wall directions instead of “Manhattan-World” assumptions. In summary, the approach proposed in this paper is based on the integration of deep learning on the 3D point cloud and the shape-grammar constraints, which takes the raw point clouds as input and

outputs a full-reconstructed BIM model. Three subsequent steps are involved in our pipeline, as illustrated in Fig. 2. First, as shown in Fig. 2 (b), the entire scene is classified into thirteen types based on the 3D deep learning method proposed by Hu et al. [20], and then the potential structure elements such as the wall, column, ceiling, floor, and the opening elements are extracted. Subsequently, a set of local geometric primitives, e.g., vertical planes, and horizontal planes are detected with a hybrid surface extraction method shown in Fig. 2(c). Then, as shown in Fig. 2(d) and (e), all the wall planes are projected onto the X-Y plane, which will be optimized by normal, parallel constraints. The optimized wall planes are then used to partition the 2D space into a grid map. Meanwhile, the original vertical planes and other components are projected on the ground surface to generate a 2D binary map (Fig. 2 (f)). An improved morphological algorithm is used to generate a 2D space decomposition map shown in Fig. 2(g). This step enables us to divide the whole scene into different functional spaces. It is worth mentioning that functional space means an area, or room that is substantially enclosed by a roof and walls, regardless of whether the roof or walls are permanent or temporary, open or closed. Finally, the 2D space decomposition results are optimized with an effective space regularization method by assigning a label configuration to each grid cell, in which the label configurations are determined by solving a Markov Random Field problem (Fig. 2 (h)). After that, the BIM model shown in Fig. 2 (i) is reconstructed by a parametric modeling process. The final model is the union of the main indoor components consisting of room space, wall volumes, ceiling volumes, floor volumes, and opening components, in which the relations among different components are also maintained. It should be noted that considering the generalization problem of deep learning point cloud classification, the indoor 3D reconstruction method proposed in this paper can directly process the original unclassified point clouds to construct BIM models containing structures such as walls, ceilings, floors, and columns.

3.2. Classification of indoor point cloud with 3D deep learning

Unlike other state-of-art methods, except for the 3D layout of the indoor interior, the main components such as windows and doors are also considered in our method, which is hard to detect based on traditional shape grammar. Thus, the 3D deep learning framework, RandLA-Net[20] is used for point labeling. The segmentation categories mainly include thirteen common types such as the wall, floor, ceiling, door, window, and so on. In our method, the point cloud labeling by wall, beam, ceiling, door, floor, and window is used as the initial data for indoor reconstruction. Fig. 3 shows the sample of point labeling based on 3D deep learning. Based on the 3D point cloud deep learning classification method, the original point cloud data is sliced into three groups shown in Fig. 4. The first group contains walls, columns, ceilings, and floors. The second group contains door components. The third group contains window components. On this basis, the first set of data is mainly used for modeling the interior structure, and the remaining two sets of point cloud data are used for modeling the window and door components. It should be noted that the algorithm can perform semantic recognition of both open and closed doors, as multiple types of features such as color, geometry, and density are involved in training an integrated model. On the other hand, the algorithm may fail to obtain an accurate point cloud of the windows when the windows are covered by curtains. This will require more training data to improve the accuracy of the semantic classification model in the occlusion situation. Fig. 4 shows the results of components used for BIM reconstruction.

3.3. Robust scene decomposition based on the improved morphological algorithm

3.3.1. Surface extraction and refinement

Intuitively, the surfaces of the indoor scene characterize the interior structure across different parts. Even curved structures can be well fitted

Table 1

Overview of reconstructed features by recent indoor modeling approaches.

References	Non-Manhattan	Volumetric walls	Volumetric spaces	Doors and Windows
Xiong et al. [49]	No	No	No	Yes
Oesau et al. [34]	Yes	No	No	No
Macher et al. [30]	No	Yes	No	Yes
Liu et al. [29]	Yes	No	No	No
Ochmann et al. [32]	Yes	Yes	No	No
Tran and Khoshelham [45]	Yes	Yes	Yes	No
Fang et al. [12]	Yes	No	No	No
Our method	Yes	Yes	Yes	Yes

with planes. To enhance the robustness of plane extraction and classification, knowledge of building shapes and grammar is fully utilized. In the process of structure primitive segmentation, to filter the undesired surfaced out, a two-level process to integrate shape and grammar rules, including normal constraints, histogram of the height and areas, is proposed to retain the optimal plane candidates. It should be noted that the algorithm can handle both classified and unclassified original point clouds. Thus, given the input point cloud, this step aims at decomposing the entire scene into planar primitives of structures $\mathcal{P} = \{\mathcal{P}_i\}_{i=1,\dots,n}$, orientation and dimensional parameters will be calculated for each planar primitive. In this section, a consecutive four-step approach to achieve these objectives was proposed as Fig. 5.

3.3.2. Planar primitive segmentation

The algorithm starts by detecting all the potential planar primitives $\mathcal{P} = \{\mathcal{P}_i\}_{i=1,\dots,n}$ from an input raw triangular mesh by a hybrid surface detection method [38]. In this step, a region growing segmentation method is first applied to divide the original point cloud into different clusters. Considering the noise problem of the point cloud data, a pair of

parameters are carefully determined ($N = 50$, $S = 1$) to constrain the region growing process, where N encodes the number of regional growth proximity points and S denotes the regional growth curvature threshold. Since all point clouds were downsampled to 5 cm resolution in our experiments, the N will be fixed to 50, which means 50 neighborhoods will be automatically acquired as candidate seed points during region growing. S is used to determine whether the current point needs to be added to the seeds set. Larger S may result at larger areas of each segmented region in the process of regional growth. However, this has little effect on the accuracy of structure extraction, so fixed value 1 is set for parameter S . After that, the original point cloud data will be sliced into multiple pieces $R = \{R_i\}_{i=1,\dots,m}$ according to the region growth rules. Subsequently, the vertical and horizontal planes of each region R_i were extracted using a random sample consensus (RANSAC)-based plane-fitting algorithm, in which the parameters d_i are used to constraint the distance between the points and the target planar primitives and R_k encodes the number of proximity points in the calculation of the normal vector.

3.3.3. Structure primitive extraction

However, the segmentation process may still produce undesired surfaces with a small number of supporting points when applied to a complex point cloud of a building. To tackle this problem, two-level shape grammar rules are proposed to retain the optimal plane candidates. According to the prior knowledge about the indoor environment, the normal rule is first used to divide all planar primitive $\mathcal{P} = \{\mathcal{P}_i\}_{i=1,\dots,n}$ into three categories, including vertical planes \mathcal{P}_{ver} , horizontal planes \mathcal{P}_{hor} and other planes \mathcal{P}_{oth} [32,34]. Since the entire scene is registered in advance such that its upward direction aligns with the z-axis, a simple but robust way is to analyze the angles difference between the planar primitive and the horizontal plane. Since the average error of the point cloud from the LiDAR or RGB-D sensor is generally 2–4 cm in 10-m lever distance (suitable for most of the rooms). Based on the tangent formula, the maximum angular drift is around 10°. Therefore the

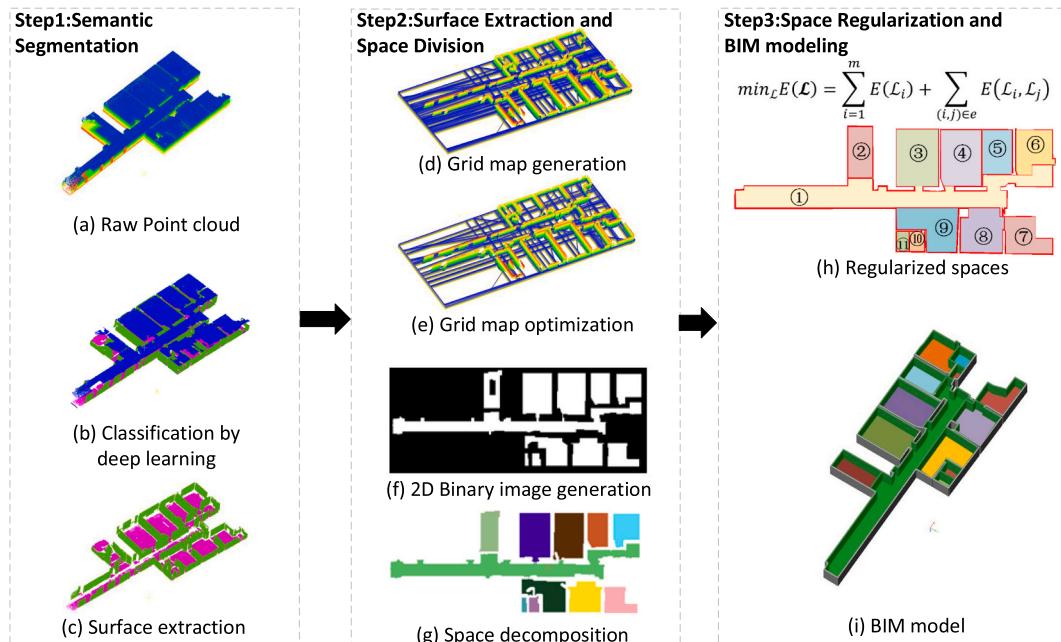


Fig. 2. Overview of the main steps. (a) The input is a registered but otherwise unstructured and unfiltered indoor point cloud. (b) Components are classified by the deep learning process. (c) Vertical and horizontal planes are segmented using a robust surface extraction and global optimization method. (d) Grid maps generated with the fitted vertical planes for space regularization. (e) Grid maps are optimized by normal, parallel constraints. (f) The 2D binary map is generated by projecting vertical planes and the raw point cloud onto the X-Y plane. (g) 2D space decomposition result based on an improved morphological algorithm. (h) A 3D plane arrangement is constructed by a space shape regularization method. (i) The final IFC model consisting of an interrelated room, wall volumes, ceiling volumes, and floor volumes is reconstructed by a parametric modeling process, in which the relation of each component remains.

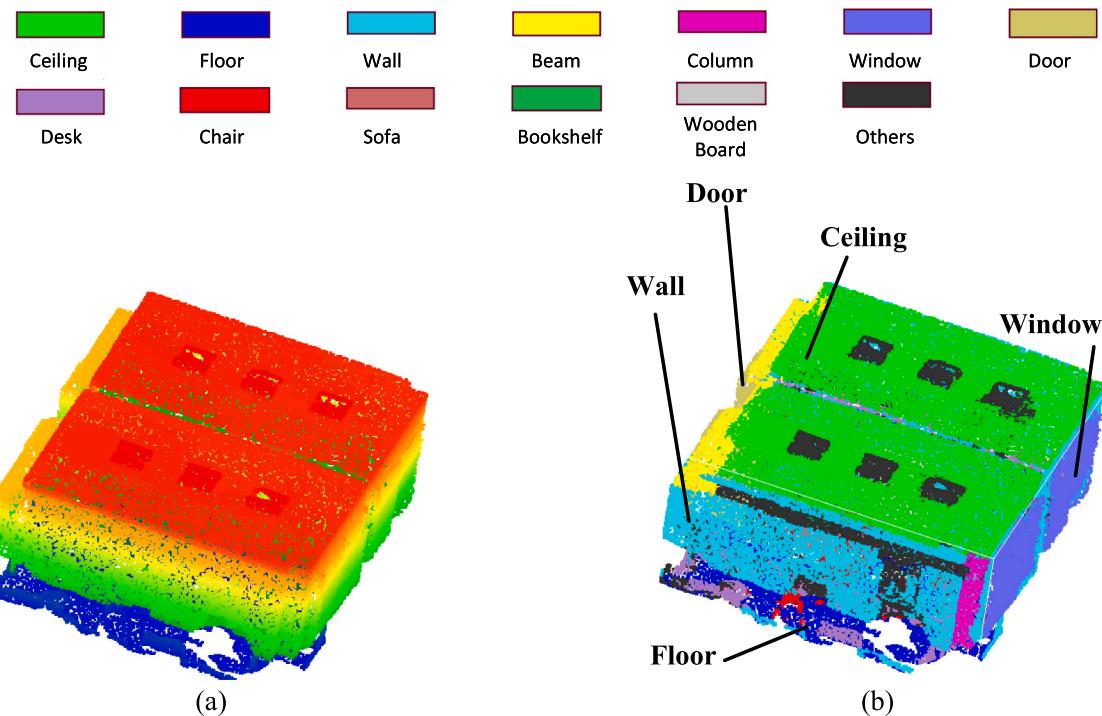


Fig. 3. Point labeling based on 3D deep learning method. (a) original point cloud, (b) labeled point cloud with different component types.

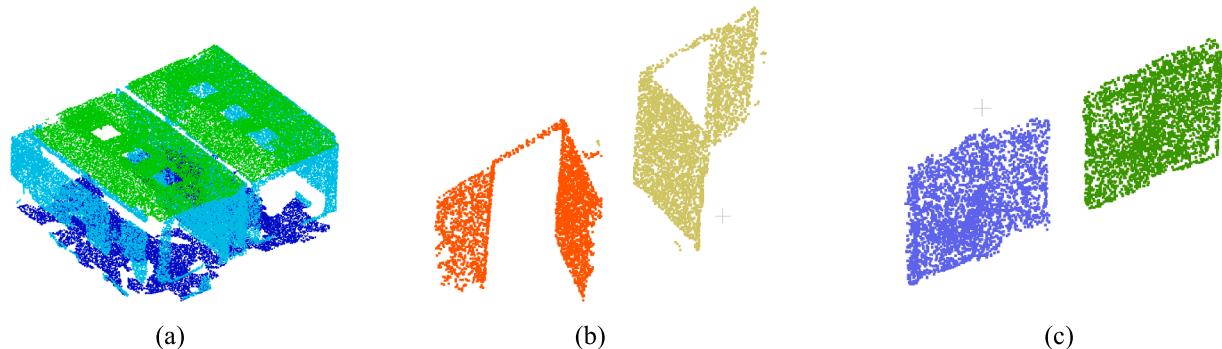


Fig. 4. Extracted components used for BIM reconstruction. (a) point cloud containing wall, beam, ceiling and floor, (b) doors segmented by clustering method, (c) windows segmented by clustering method.

thresholds of 10° and 80° are used for the extraction of the in horizontal and vertical planes. The framework considers a primitive \mathcal{P}_i as a horizontal plane if angle $(\vec{n}_i, \vec{Z}) < 10^\circ$, \vec{n}_i represent the normal vector of \mathcal{P}_i . Similarly, the vertical planes can be selected when angle $(\vec{n}_i, \vec{Z}) > 80^\circ$. The angle θ can be calculated with Eq. 1, in which v_1 is the normal of \mathcal{P}_i , v_2 is the direction of the horizontal plane. Our second level of hierarchy lies in selecting planar primitives which present the structural characteristics of the interior model and entitling semantic information for them. The floors \mathcal{P}_{floor} and ceilings $\mathcal{P}_{ceiling}$ will be first extracted from the group of horizontal planes. The height of the planes in \mathcal{P}_{hor} are calculated and sorted. Then the algorithm divides the z-axis into bins of 5 cm and projects all the large horizontal planes in the histogram. The primitives are ceiling planes $\mathcal{P}_{ceiling}$ if the height of the primitive located in the right of the histogram and the area is larger than $4m^2$. Likewise, when the height of the primitive located at the bottom of the histogram and the area is greater than $4m^2$, the primitive is considered to be the floor surface \mathcal{P}_{floor} . Simultaneously, wall planes \mathcal{P}_{wall} and column planes \mathcal{P}_{column} would also be selected from \mathcal{P}_{ver} . Empirically, walls are relatively long vertical parts that are close to the corresponding ceilings

(see Fig. 5(a) and (b)). In this method, the primitive in \mathcal{P}_{ver} is denoted as wall plane if (i) $height(\mathcal{P}_{ver}) > 1.5m$, (ii) $min(\mathcal{P}_{ver}, \mathcal{P}_{ceiling}) < 0.5m$, (iii) $area(\mathcal{P}_{ver}) > 3m^2$. And the primitives are considered as column planes if the first two conditions for wall planes are satisfied and the area is large than $1m^2$ and less than $3m^2$. Finally, all the extracted primitives throughout the above method make up the structure primitive set $P_{sp} = P_{ceiling} \cup P_{floor} \cup P_{wall} \cup P_{column}$.

$$\theta = \cos^{-1}(v_1 \cdot v_2 / (\|v_1\| \times \|v_2\|)) \quad (1)$$

3.3.4. Robust space decomposition by improved morphological approach

Given an arrangement of the structure primitives and the original point cloud, this step makes an effort toward generating a set of functional spaces by using an improved morphological segmentation process.

3.3.4.1. Primitive optimization. The rigorous parameters for plane segmentation may lead to an over-segmentation of planar primitives (see Fig. 5(c)). To recover the correct affiliation of the primitives, two constraints, including distance constraints in normal, parallel constraints,

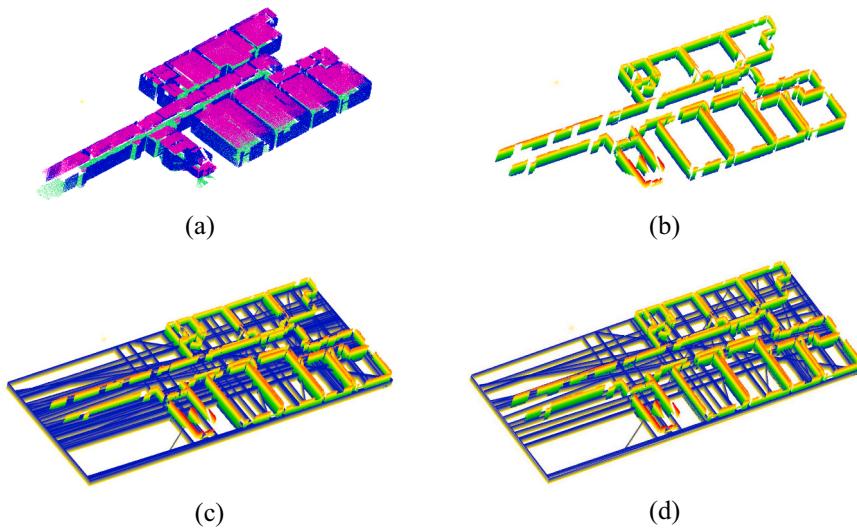


Fig. 5. Surface extraction and refinement, (a) segmented vertical and horizontal planes; (b) extracted vertical surfaces; (c) fitted planes of raw vertical surfaces; (d) optimized planes by the global refining process.

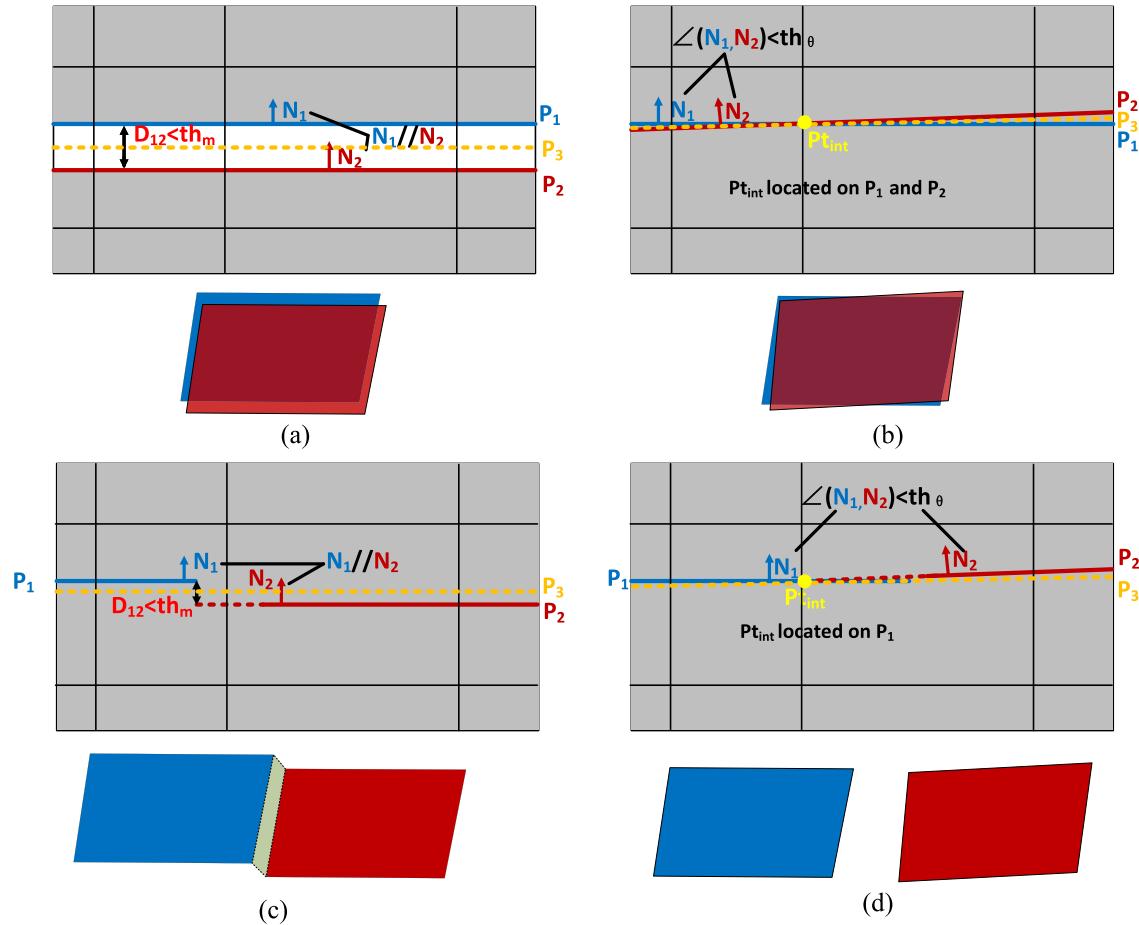


Fig. 6. Explanation of notation and constraints for plane refinement, (a) two projected planes are with small distance and parallel. (b) the angle between the two post-projection planes is small and the intersection point is in the middle of the line segment. (c) two projected planes are also with small distance and parallel. (d) the same wall is divided into multiple planes due to the noise of the point cloud.

are used to guide the plane optimization, which iteratively merges the pair of adjacent primitives that are most likely beyond the same components. The primitive merging procedure terminates when no more pairs of primitives $(\widehat{p}_{\text{spi}}, \widehat{p}_{\text{spj}})$ full fill the following hypothesis: (i) angle

$\left(\widehat{p}_{\text{spi}}, \widehat{p}_{\text{spj}}\right) < \theta$, (ii) distance in normal direction $\left(\widehat{p}_{\text{spi}}, \widehat{p}_{\text{spj}}\right) < \tau$. In our experiments, θ and τ is recommended to be set to 10° and 0.2 m. As shown in Fig. 6, four cases satisfying the merging condition are plotted, in which the planar primitives are projected on the horizontal or vertical



Fig. 7. 2D binary image generated by projected point cloud with different expansion factor, (a) $E_{factor} = 0$, (b) $E_{factor} = 1$, (c) $E_{factor} = 2$, (d) $E_{factor} = 3$.

surfaces and fitted with a straight line. Case 1 (see Fig. 6(a)) presents the situation in which two projected planes are small distance and parallel. This usually occurs in planes extracted from both sides of a single wall. Case 2 (see Fig. 6(b)) shows the situation where the angle between the two post-projection planes is small, and the intersection point is in the middle of the line segment. This usually occurs in planes extracted from the floor, ceiling, or one side of a wall. In Case 3 (Fig. 6(c)), two projected planes are also small distances and parallel. Not quite the same as Case 1, the pendant point from the endpoint of a line segment l_1 to another line segment l_2 is outside l_2 . This situation usually occurs when two parallel walls are joined by a narrow wall. Case 4 (see Fig. 6(d)) also usually occurs in planes extracted from the same wall, in which the same wall is divided into multiple planes due to the noise of the point cloud. The large distance between these two projected line segments may be due to the fact that the entire wall is separated by the column. Fig. 5(c) and (d) shows the wall P_{wall} and column P_{column} surfaces before global refinement and after refinement. Finally, the surface set P_{sp} is iteratively optimized and the final optimized surface set \widehat{P}_{sp} is represented with $\widehat{P}_{ceiling} \cup \widehat{P}_{floor} \cup \widehat{P}_{wall} \cup \widehat{P}_{column}$.

3.3.4.2. 2D binary image generation. Intuitively, the functional spaces in the building interior are divided through a series of permanent vertical planes detected from walls and columns. In this step, the algorithm first project the original point cloud and those vertical structure primitives onto the plane separately and generate the corresponding binary 2D image, in which the occupied pixels by the raw point cloud are marked in white and the pixels occupied by are marked in black. However, due to inevitable occlusion during data collection or incomplete primitives detection, it is easy to cause unclear boundaries and result in inaccurate segmentation during watershed segmentation. Therefore, an expansion factor was introduced to strengthen the boundaries of the adjacent spaces. Fig. 7 shows the binary images acquired with different expansion parameters E_{factor} . It can be seen that the larger the expansion coefficient E_{factor} result in, the more obvious partition boundary. However, the space size shrinks with increasing expansion coefficient, which may lead to inaccurate spatial regularization described in Section 3.3.3. Therefore, in this method, the optimal expansion factor value can be set to either 1 or 2.

3.3.4.3. Noise rejection. Due to the presence of noise in the original



Fig. 8. Noise rejection results by contour filtering, (a) the contours generated from the original binary image, (b) the remained contours after noise rejection, (c) the binary image generated from the original binary image, (d) the binary image after noise rejection.



Fig. 9. space decomposition based on the morphological algorithm.

point cloud, these noises appear as scattered white pixels in the binary image, which may have a negative impact during the morphological space division process. Therefore, to remove the pixel noises in the binary image, a noise rejection method is proposed by retrieving contours from the binary image using the Suzuki85 algorithm [41]. As shown in Fig. 8, the binary image before and after noise rejection are presented. The algorithm calculates the number of pixels occupied by each contour and removes the contours that occupy pixel numbers less than a certain threshold.

3.3.4.4. Space decomposition through the morphological method. This step aims to divide the indoor scene into different functional spaces, which can best represent the layout of the indoor space. Given the optimized 2D binary image, the framework performs a mean shift segmentation on the image, and the output of the function is the filtered “posteriorized” image with color gradients and fine-grain texture flattened. After that, a distance transformation was applied to the image, which calculates the approximate or precise distance from every binary image pixel to the nearest zero pixels, shown in Fig. 9(a). For zero image pixels, the distance will obviously be zero. Finally, watershed transformation is employed for space decomposition. The watershed transformation treats the image it operates upon like a topographic map, with the brightness of each point representing its height, and finds the lines that run along the tops of ridges. Fig. 9(b) shows the space decomposition results based on the proposed morphological method, in which different functional spaces are distinguished by different colors. Likewise, for each region, their corresponding contours from the segmented image are retrieved using the algorithm proposed by Suzuki and Be [41] and a set of boundary shapes is obtained.

Based on the proposed space decomposition method, the building interior will be divided into different functional regions, representing a basic floor plan of the building structure. With the advantage of the adaptability of data occlusion and noise, this method can produce an accurate indoor structure without regularization.

3.3.5. Space shape regularization

The objective of this step is to extract an accurate boundary shape of each space. The space is divided into a grid map by the wall, while the layout of the indoor scenes recovered by space decomposition presented an irregular boundary shape. The former has better regularity, but each “grid cells” have no spatial semantic information. The latter is with spatial semantics in each “functional space” but irregular boundaries. Different from the previous method, which mainly uses the MRF model to determine the inclusion relationship between walls and rooms, the MRF-based space shape regularization method proposed in this paper is used to model the intersection problem of “grid cells” and irregular boundaries of “functional space”. Therefore, the assumptions for the space shape regularization is as follows. 1) The larger area of intersection between a grid cell and a functional space, the higher probability that the grid cell belongs to that functional space; 2) All grid cells that fall into the same functional space can be combined into one regularized “functional space”. It should be noted that the boundary of the space should also conform to the manifoldness assumption, in which the vertex is only connected to two adjacent edges. The core of the method is

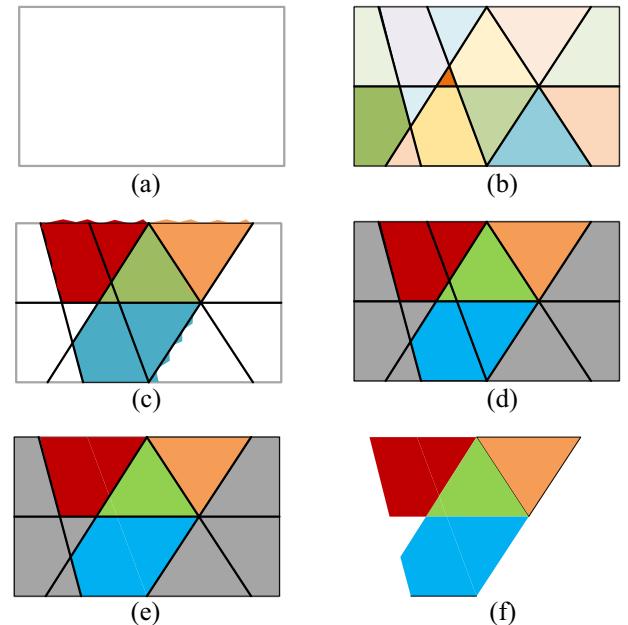


Fig. 10. Space regularization using raster and vector map, (a) projected vertical surfaces, (b) grid map generated by line intersections of vertical surfaces, (c) overlay of the map of space decomposition and the grid map, (d) intersection and cell labeling, (e) cell merging and (f) Interior structure after space regularization.

to organize the scene by grid map generated by line intersections of vertical planes and recover the subordination between grid cells and the segmented boundary shapes $\mathcal{B} = \{\mathcal{B}_i | 1 \leq i \leq n\}$ obtain in Section 3.3.2.

Given an arrangement of the structure primitives $\widehat{\mathcal{P}}_{\text{wall}}$ and $\widehat{\mathcal{P}}_{\text{column}}$ in Fig. 10(a), the algorithm calculates an oriented bounding box(OBB) based on the projected point cloud, which aligns with the principal axis of the projected point cloud and the grid map of the building interior is generated through iteratively intersections between the fitted lines of $\widehat{\mathcal{P}}_{\text{wall}}$ and $\widehat{\mathcal{P}}_{\text{column}}$ and the polygons. Fig. 10 (b) shows the grid cells $\mathcal{G} = \{\mathcal{G}_i | 1 \leq i \leq m\}$ of the building interior, in which the counterclockwise cyclic sequence of the alternating polygon edges and polygon vertices is referred to as the polygon of each grid cell. Then, an overlay analysis (Fig. 10(c)) between the segmented boundary shapes $\mathcal{B} = \{\mathcal{B}_i | 1 \leq i \leq n\}$ and the grid map is used to assign a room instance (Fig. 10(d)) labeling $\mathcal{L} = \{\mathcal{L}_i | 1 \leq i \leq m\}$ to each grid cells $\mathcal{G} = \{\mathcal{G}_i | 1 \leq i \leq m\}$. Finally, the grid cells with the same labeling are merged (Fig. 10(e)), and the regularized room boundaries are obtained (Fig. 10(f)). In particular, the intersection problem is modeled as a Markov Random Field approach via the energy of the standard form in Eq. 2 as follows.

$$\min_L E(L) = \sum_{i=1}^m E(L_i) + \sum_{(i,j) \in e} E(L_i, L_j) \quad (2)$$

where L is a labeling configuration and for each $\mathcal{L} \in \mathcal{L}$, the value space is $\mathcal{B} = \{\mathcal{B}_i | 1 \leq i \leq n\}$. $\mathcal{E}(\mathcal{L}_i)$ encodes unary term and encodes pairwise

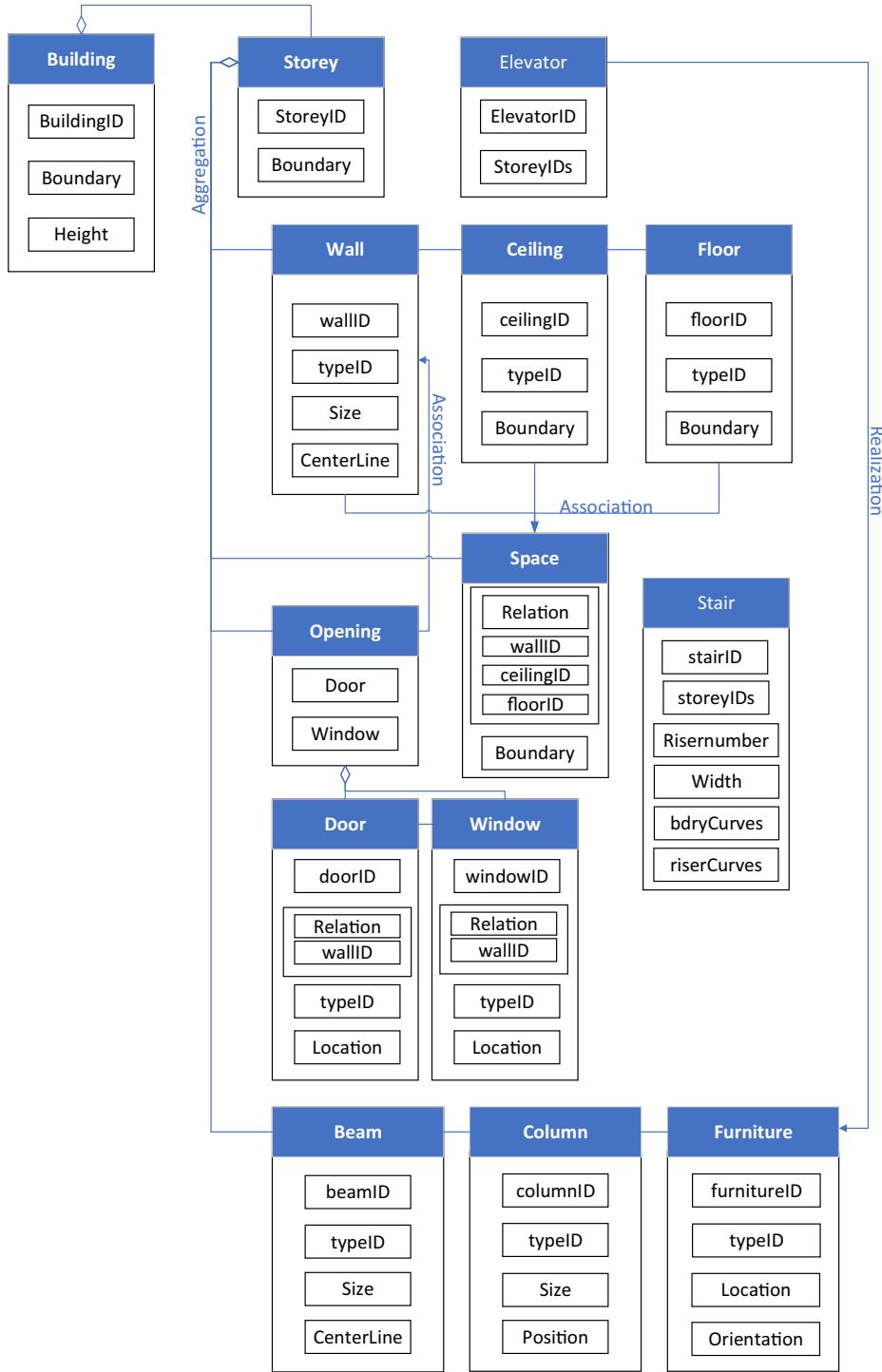


Fig. 11. Parametric representation for building model.

term. Denotes all pairs of adjacent grid cells.

3.3.5.1. Unary energy. $\mathcal{E}(\mathcal{L}_i)$ is designed to encourage assigning each grid cell a label that is with adequate overlapping relations with the segmented boundary shapes $\mathcal{B} = \{\mathcal{B}_i | 1 \leq i \leq n\}$. The term $\mathcal{E}(\mathcal{L}_i)$ are represented as following:

$$\mathcal{E}(\mathcal{L}_i) = -\sqrt{O_i} \log P(\mathcal{L}_i), \quad (3)$$

where O_i is number of the occupied pixels to measure the weight of each grid cell. $P(\mathcal{L}_i)$ is the probability of \mathcal{L}_i to be labeled as \mathcal{L}_i , which is

computed as the ratio of the number of the occupied pixels with label \mathcal{L}_i in the segmented boundary shapes \mathcal{B} to the number of total pixels of the grid cell.

3.3.5.2. Pairwise energy. $E(\mathcal{L}_i, \mathcal{L}_j)$ presented the adjacent constraints between grid cells. The penalties of adjacent relations make the labeling results separated by vertical planes. To achieve it, the edges are penalized whose adjacent cells preserve the same room instance labels that contain wall segment:

Table 2
Samples of constructed BIM components.

Type	Sample 1	Sample 2
Wall	<p>Parameters:</p> <ul style="list-style-type: none"> • wallID:0 • typeID:001001001001 • Size:Height2700mm • CenterLine:Location1(3,2.7,0), Location2(10,2.7,0) 	<p>Parameters:</p> <ul style="list-style-type: none"> • wallID:1 • typeID:001002001001 • Size:Height2700mm • CenterLine:Location1(5,5,0), Location2(12,12,0)
Door	<p>Parameters:</p> <ul style="list-style-type: none"> • doorID:0 • Relation:Embing • wallID:0 • typeID:004001001001 • Location:(3,2.7,0) 	<p>Parameters:</p> <ul style="list-style-type: none"> • doorID:1 • Relation:Embing • wallID:0 • typeID:004002001001 • Location:(8,6,0)
Window	<p>Parameters:</p> <ul style="list-style-type: none"> • windowID:0 • Relation:Embing • wallID:0 • typeID:005001005001 • Location:(5,3,8.3) 	<p>Parameters:</p> <ul style="list-style-type: none"> • windowID:1 • Relation:Embing • wallID:0 • typeID:005001004002 • Location:(5,3,3.4)
Ceiling	<p>Parameters:</p> <ul style="list-style-type: none"> • ceilingID:0 • typeID:003001001001 • Boundary:Location1(2,0,0), Location2(7,0,0), Location3(7,4.5,0), Location4(2,4.5,0) 	<p>Parameters:</p> <ul style="list-style-type: none"> • ceilingID:1 • typeID:003001002001 • Boundary:Location1(1.8,2,0), Location2(6.8,2,0), Location3(6.8,7,0), Location4(1.8,7,0)
Floor	<p>Parameters:</p> <ul style="list-style-type: none"> • floorID:0 • typeID:002001001001 • Boundary:Location1(0,0,0), Location2(5,0,0), Location3(5,5,0), Location4(0,5,0) 	<p>Parameters:</p> <ul style="list-style-type: none"> • floorID:1 • typeID:002001006001 • Boundary:Location1(8,3,0), Location2(13,3,0), Location3(13,8,0), Location4(8,8,0)

$$E(\mathcal{L}_i, \mathcal{L}_j) = \begin{cases} 0 & \text{if } \mathcal{L}_i \neq \mathcal{L}_j \\ -\sqrt{(l_i)} \log \left(1 - \frac{|\tilde{l}_i|}{|l_i|} \right) & \text{if } \mathcal{L}_i = \mathcal{L}_j \end{cases} \quad (4)$$

where l_i is the common edge between grid cells \mathcal{G}_i and \mathcal{G}_j . $|l_i|$ is the length of the common edge and $|\tilde{l}_i|$ is the length of the edge l_i overlapping with corresponding wall segment.

The MRF model is solved by a standard graph-cut optimization algorithm with the above space regularization terms. Furthermore, all grid cells with the same label are merged and generate a regularized space. To simplify the polygon of each space, the redundant vertices are removed by the angle between adjacent edges.

3.4. Parametric BIM generation

In this framework, the BIM reconstruction is based on a parametric strategy. In contrast to previous approaches, the proposed parametric representation framework provides a minimalist abstraction of the BIM model, allowing the user to generate geometrically and semantically BIM model with a minimum number of parameters. Based on this conceptual model, A GML data description is used to store component parameters, and an automated BIM generation toolkit is provided to facilitate BIM construction from expert BIM software. The new proposed GML data structure can be used as a procedural data during BIM reverse modeling either from point clouds or 2D drawings, which can

Table 3

The real data sets including point cloud and ground truth model constructed by BIM software.

ID	Original Pointcloud	Ground truth model	Size of the region(m ³)
1			9.81×38.81×2.96
2			7.43×24.25×2.89
3			7.56×44.73×2.98

significantly reduce the problem of data degradation for the traditional indoor reconstruction methods.

In details, parameterization for each component differs from the creation model and the inherent relations among them, which results in a different combination of the parameters. As shown in Fig. 11, we provided a detailed parametric definition for the building and parts. Those parameters facilitate the creation of building components and support the recovery of the spatial and semantic relationships of them. Typically, all components contain an orphan ID, which provides an independent identifier in the building model. "TypeID" is used to guarantee a correct style for those components. Encoding rules for "TypeID",

which consists of "Component type ID", "Structure Type ID", "Feature Type ID" and "Size ID" are used. For example, the "component type ID" of a wall component is defined as "001", and three kinds of structure, basic wall, curtain wall, and exterior wall are separately encoded with "001" to "003". For each kind of structure, the specific feature of a wall is defined by "Feature Type ID". It should be noted that the size of the wall, ceiling, and floor is also determined by "Feature Type ID". Therefore, the "Size ID" is always "001". Besides, in this model, all relations are associated with ID information of the components, and the geometric information is abstracted into basic geometric primitive, such as width, position, and so on. The building and parts, its indexes, features, construction modes, and relations are presented in the following.

Table 4

Description of the statistics of the components and the parameters for BIM reconstruction.

Ground truth	Case study1	Case study2	Case study3
Rooms	13	10	13
Doors	13	8	15
Windows	12	0	15
Parameters	Case study1	Case study2	Case study3
d_t	0.03	0.04	0.02
E_{factor}	1	2	1
r_k	25	25	25
T_{occupy}	0.4	0.5	0.5

3.4.1.1. Building. A building can be represented by a 2.5D polygon with footprint level height and partitioned into different stories, which contains the parameters of "BuildingID", "Boundary" and "Height".

Storey: The space of one story can be represented with a "Boundary", which is the aggregation of all components in the story. Each story is associated by the components of the Elevator and Stair.

Elevator: In this model, Elevator is inherit from the "Furniture". The stories that the Elevator passed are defined by the parameter "StoryIDs".

Stair: Similar with Elevator, "StoryIDs" is also defined. While different, parameters like "stairID", "storyIDs", "Risenumbers", "Width", "CenterLine", "bdryCurves" and "RiserCurves" are used to supports the

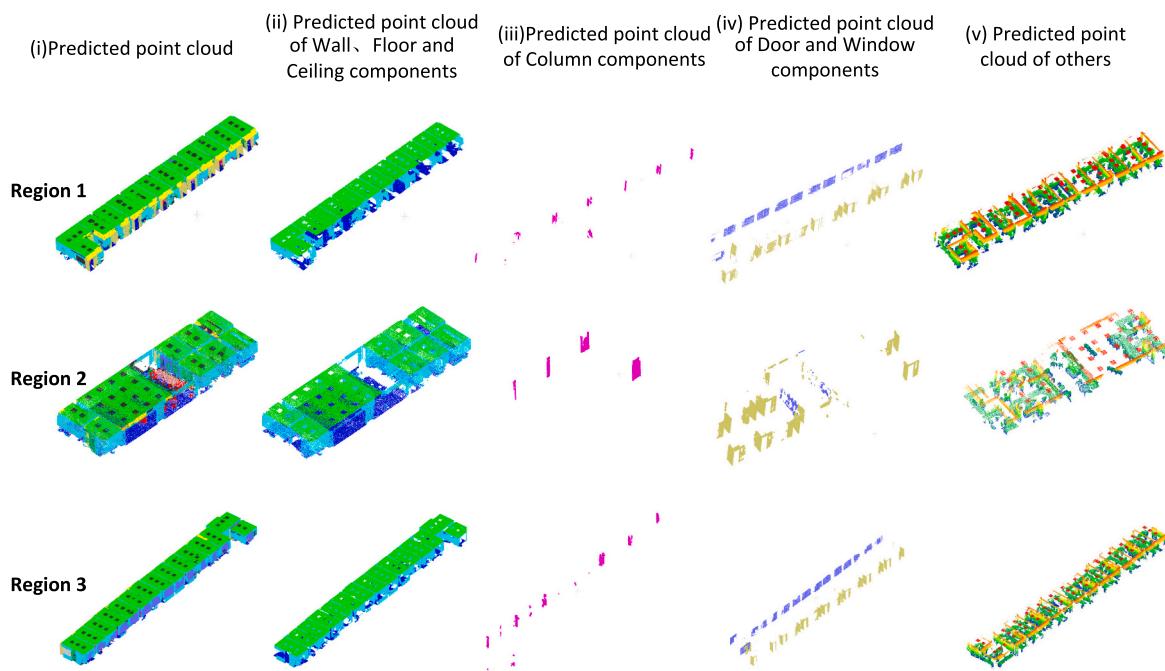


Fig. 12. The classification results of 2D-3D-S data sets.

Table 5

Precision Rate (PREC) of the point cloud classification results in 2D-3D-S datasets.

Components	Region1	Case study2	Case study3
Wall(%)	82.5	85.6	84.5
Ceiling(%)	97.2	97.5	96.4
Floor(%)	99.3	97.4	98.1
Floor(%)	86.2	90.2	82.1
Door(%)	68.5	59.2	69.5
Window(%)	59.4	45.1	65.4
Overall(%)	70.5	73.3	71.8

Table 6

Precision Rate (PREC) of the BIM reconstruction for different components.

	Space	Door	Window	Ceiling	Floor
Case studyID	TP/FP/GP/REC				
1	11/0/11	13/0/13	11/1/12	11/0/11	11/0/11
	/100%/	/100%/	/91.7%/ 91.7%	/100%/	/100%/
	100%	100%	91.7%	100%	100%
2	10/0/10	8/1/8	10/0/10	10/0/10	10/0/10
	/100%/	/88.9%/ 88.9%	0/2/0	/100%/	/100%/
	100%	100%	0%/-	100%	100%
3	13/0/13	14/1/15	15/0/15	13/0/13	13/0/13
	/100%/	/93.3%/ 93.3%	/100%/ 100%	/100%/ 100%	/100%/ 100%
100%	93.3% / 100%	100%	100%	100%	100%

Bold text indicates that there is an error detection for this component type.

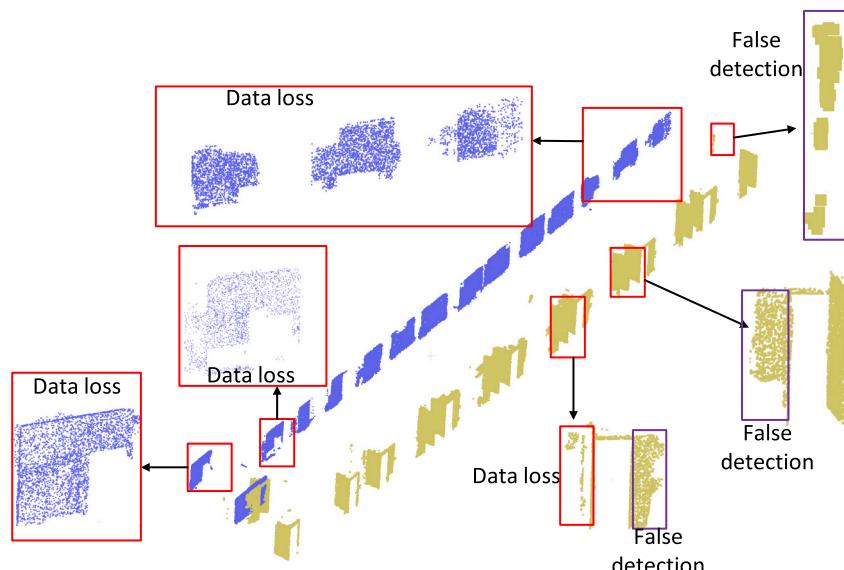


Fig. 13. Example of the errors during opening detection in Case study3.

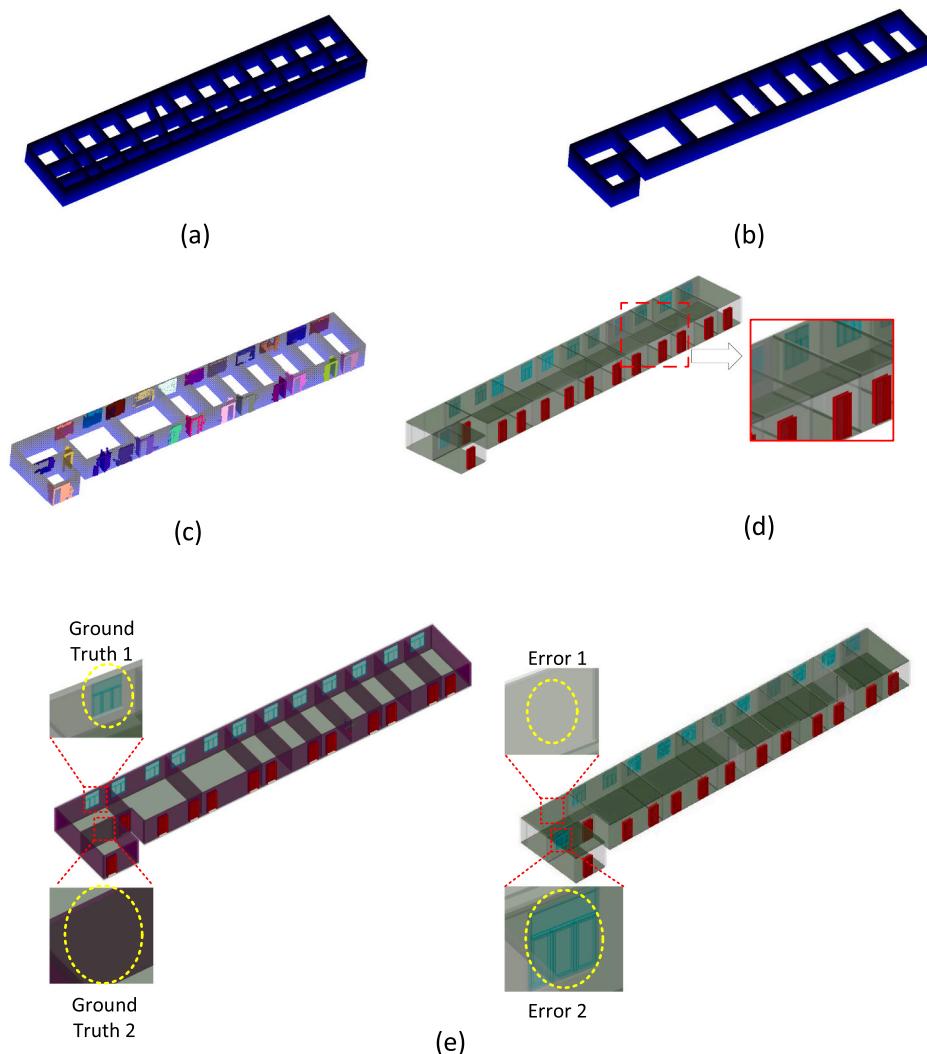


Fig. 14. Reconstruction results of Case study1, (a) the cell decompositions, (b) the reconstructed spaces, (c) the segmented opening elements, (d) the reconstructed IFC models, (e) the errors of BIM reconstruction compared with the ground truth.

automatic reconstruction of a Stair component.

Wall: Geometrically, Wall is constructed from "height" and "Centerline". The thickness and the material of the wall are defined by "TypeID". Ceiling and Floor: Ceiling and Floor are described by a 3D "Boundary", which is to be reconstructed as a solid. Similarly, the thickness and the appearance are defined by "TypeID" respectively.

Space: The Space is bounded by Wall, Ceiling and Floor components. The corresponding IDs are used to present the "Surrounding" relation between them. 3D "Boundary" is also used for Space reconstruction, which provides a constraint as geometrically closed volume object.

Opening: In each Opening (Door and Window), "WallID" is used to represent the "Embed" relation between them. Once "Location" and the associated Wall are defined, the Opening is to be reconstructed automatically. "TypeID" of Opening are presented in Table 1.

Beam: Similar with Wall, Beam can be generated with "height" and "CenterLine". Thickness and appearance are defined by "TypeID" respectively.

Column: The parameters "Location" and "Size" are used for the reconstruction of Column. The style of Column is given according to "TypeID".

Furniture: In this model, Furniture is placed according to the "Location" and "Orientation" parameters. The type of Furniture (desk, chair, cabinet, Elevator, etc.) is defined by "FurnitureID". The specific appearance is also given by "TypeID".

The results of modeling with two sets of parameter combinations are listed in Table 2. For example, the wall is reconstructed according to the defined "TypeID" and the geometric parameters (Centerline and height). The style and thickness of the wall are defined using "TypeID", which is "001001001001" and "001002001001" in Sample 1 and Sample 2, respectively. The location of the wall is parameterized with a 3D line comprising two points, and the height of the wall is determined by the parameter of Size. Similarly, the locations of the Door, Window, Ceiling, Floor, Beam, and Column are defined by single points or sequences of points. Importantly, the correlation between the Opening and the wall is given by the parameters of Relation and "WallID". As shown in Table 2, different types of components with different styles can be reconstructed based on the extracted parameters.

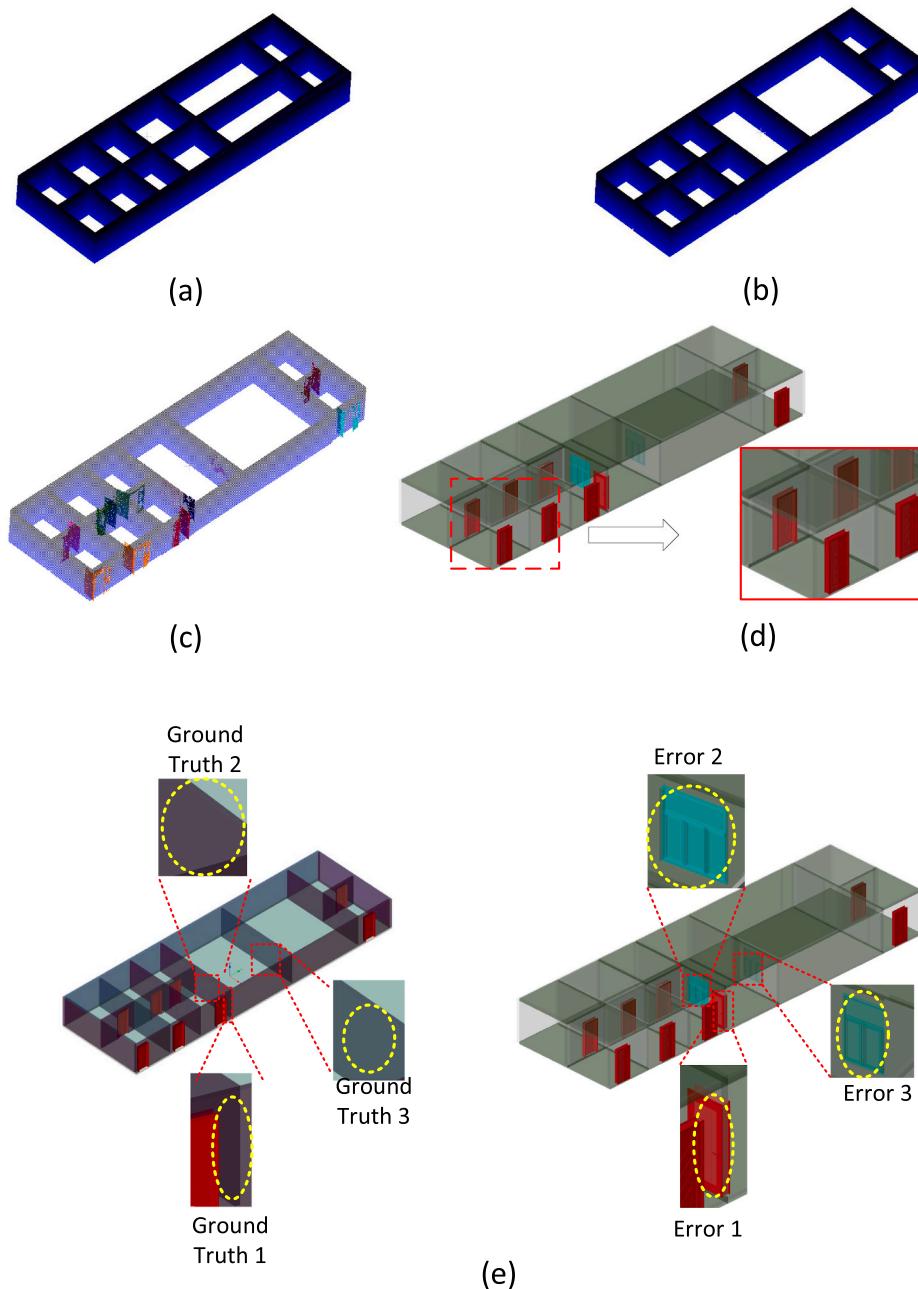


Fig. 15. Reconstruction results of Case study2, (a) the cell decompositions, (b) the reconstructed spaces, (c) the segmented opening elements, (d) the reconstructed IFC models, (e) the errors of BIM reconstruction compared with the ground truth.

4. Experimental analysis

4.1. Data preparation

The proposed approach was implemented in C++ on a personal computer (i7-7700 CPU @ 3.60GHz, with 16 GB memory) and relies upon point cloud library(PCL) [38] the computational geometry algorithms library(CGAL) [11] for basic geometric data structure and computational tools. Three sets of datasets containing the 2D-3D-S dataset from Stanford university [2], ISPRS benchmark [26] and collected by RGB-D devices are collected and used to evaluate our method for different purposes throughout the experiments.

Experiments on the 2D-3D-S dataset [2] are first conducted to test the effectiveness of our proposed methods. The 2D-3D-S dataset provides a variety of mutually registered modalities from 2D, 2.5D, and 3D domains, with instance-level semantic and geometric annotations. The dataset covers over 6000 square meters and contains over 70,000 RGB images, along with the corresponding depths, surface normals, and semantic annotations. In this experiment, the effectiveness of the proposed method on three different regions of the 2D-3D-S dataset are tested, and the reasons for parameter selection based on the experiments are fully discussed. The accuracy of both the classification results and the BIM reconstruction results are also evaluated. It should be noted that each Case study has a ground truth point cloud with semantic annotations and

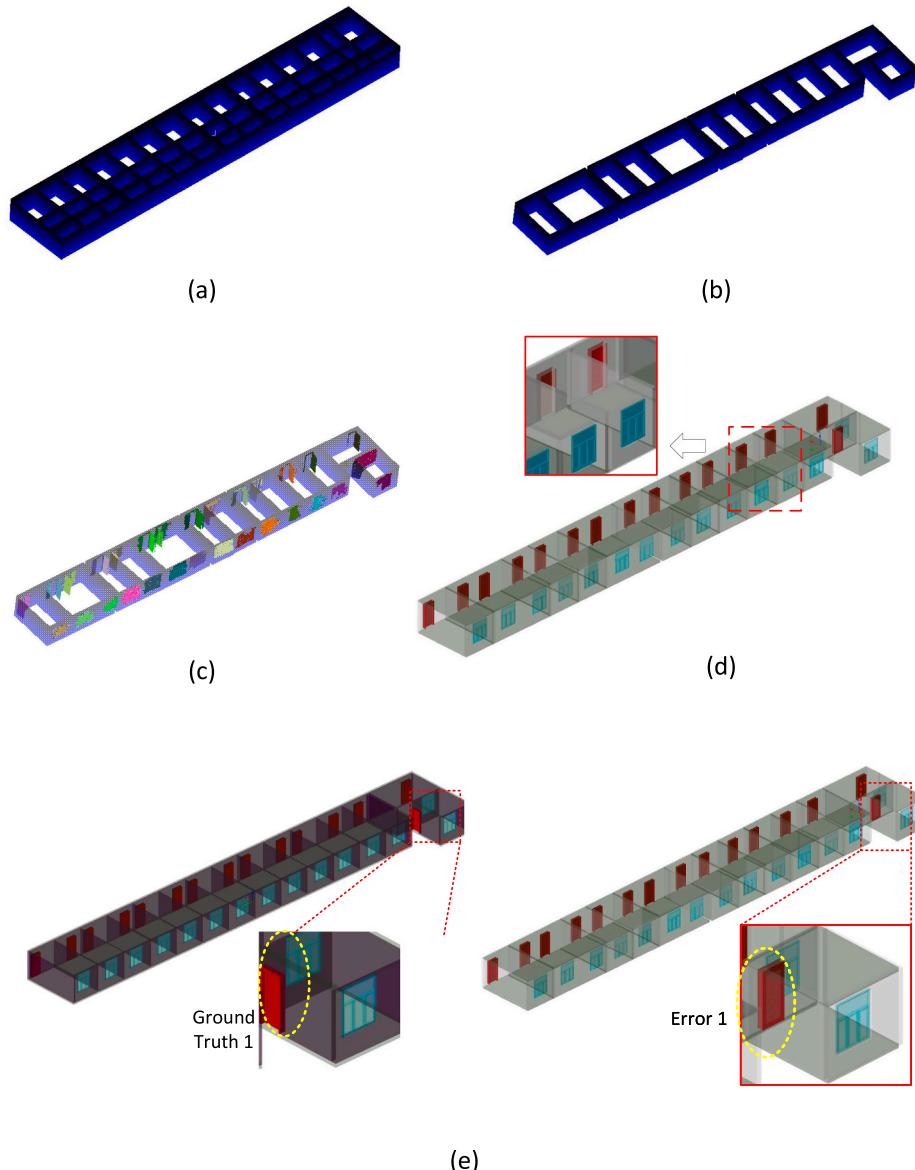


Fig. 16. Reconstruction results of Case study3, (a) the cell decomposition, (b) the reconstructed spaces, (c) the segmented opening elements, (d) the reconstructed IFC models, (e) the errors of BIM reconstruction compared with the ground truth.

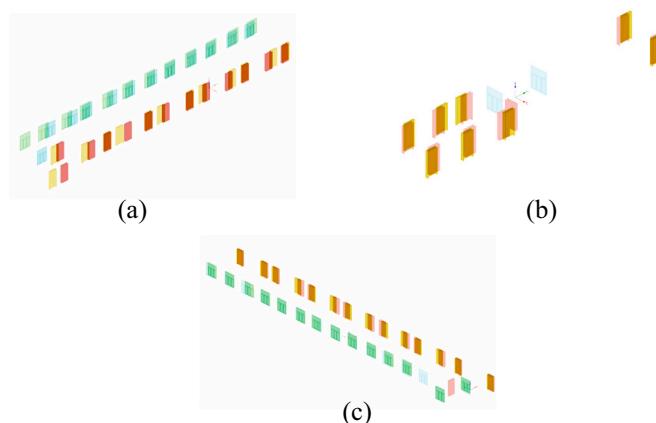


Fig. 17. Comparison the location of the windows and doors between obtained BIM model and ground truth, (a) The result of Case study 1, average error of the location is 0.21 m, (b) The result of Case study 2, average error of the location is 0.12 m, (c) The result of Case study 3, average error of the location is 0.14 m.

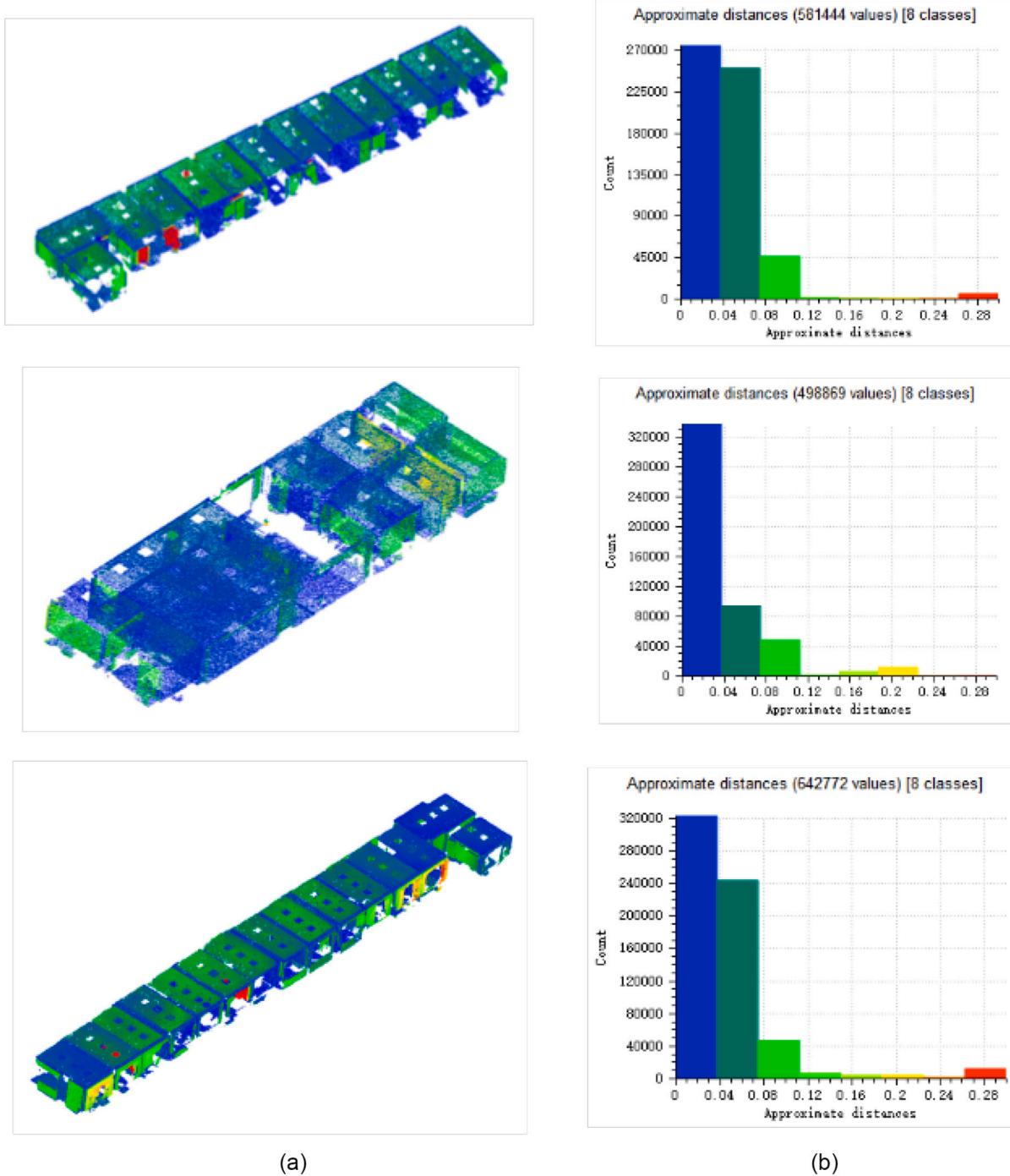


Fig. 18. Quantitative evaluation of the BIM model on three data sets of 2D-3D-S data (a) Geometric distance between the model and raw point cloud, (b) Histogram of geometric metrics on three regions.

a ground truth BIM model, which is used for the accuracy evaluation of the classification results and the reconstructed model. The ISPRS benchmark is also used for comparison purposes, which consists of five-point clouds captured by different sensors in indoor environments of various complexities. Several experiments were carried out on four data sets of ISPRS benchmark, including TUB1, UoM, the 2nd floor of TUB2, and GM, to compare our framework against the state-of-art indoor reconstruction method. In addition, four data sets collected by RGB-D devices in real scenes are also used to evaluate the feasibility of the proposed method. These data sets contain scans from simple to complex non-Manhattan indoor environments with a high level of clutters. It should be noted that the deep learning model will be trained with five

regions of the 2D-3D-S dataset and then used in the rest of the case studies. And the computational cost of the training process is about 5 h, and the testing time is about 3 min for a single room ($24 \times 7 \times 3$ m), which is implemented in C++ on a personal computer (i7-7700 CPU @ 3.60GHz, with 16 GB memory).

4.2. Experimental results of 2D-3D-S data sets

Table 3 shows three regions of 2D-3D-S data sets, which contain the point clouds and the ground truth model of the real environments. To get faithful models, four parameters d_t, r_k, E_{factor} and T_{occupy} denoting the maximum distance threshold in RANSAC, the maximum point number

Table 7

Quantitative results on 2D-3D-S data and RGB-D scenes data, all the reported errors are measured in meter.

2D-3D-S data	RMS	Avg dist
Case study1	0.06	0.007
Case study2	0.087	0.054
Case study3	0.1	0.019

RGB-D scenes data	RMS	Avg dist
Scene 1	0.09	0.02
Scene 2	0.222	0.086
Scene 3	0.238	0.074
Scene 4	0.127	0.018

of normal estimation of the expansion, the expansion coefficient for binary image generation and the occupying ratio of intersection between raster and grid map respectively need to be tuned. The specific values of these three parameters and the statistics of the components in three regions are also listed in Table 4. For the distance threshold in RANSAC, a higher value is more appropriate for data with lower quality so that it can avoid generating multiple planes from one wall in the plane fitting process. The point number for normal estimation is set equal to indicate an equal consideration of the density of the point clouds. The occupying ratio is set according to the level of completeness of the point cloud, and a higher value indicates better completeness of the original point cloud.

4.2.1. Results of the classification of point cloud

The paper first performs a thorough evaluation of the segmentation method for point cloud labeling. In this work, the RandLA-Net framework[20] generally works well on the three test data sets. Fig. 12 shows the classification results of the data sets, in which the main types of components are segmented and presented. As the aim of our experiments is to reconstruct the whole layout of the building interior, the point clouds after classification are partitioned into parts, including the predicted point cloud of the wall, floor, and ceiling components, the predicted point cloud of door components, the predicted point cloud of window components, and the predicted point cloud of others. The predicted point cloud containing wall, floor, ceiling, and column components is used for plane extraction. The predicted point clouds of door and window are used for wall and window reconstruction, respectively. The precision rate (PREC) of the point cloud labeling is calculated as follows:

$$PREC = \frac{TP}{(TP) + (FP)} \times 100 \quad (5)$$

where TP is the sum of true positives for each component, FP is the sum of false positives for each component. From the results listed in Table 5, the results show that the $PREC$ of the wall, ceiling, floor, and column consistently outperforms the results of the door and window and presents a high classification accuracy of over 80%. This shows that geometry features domains the segmentation process and thus provides better results on the planar elements. Meanwhile, Fig. 12(ii) shows the predicted wall, floor, ceiling, and column components labeled by a different color, which also indicates the efficiency of the method in these three kinds of components. In these three experiments, the $PREC$ of the door and window are both lower than 70%. As shown in Fig. 13, some examples of data loss and false detection are presented. For the window component, several windows have large data missing problems, which results in low classification accuracy for windows. There are two main problems with the classification of door parts, false detection, and data loss. In these cases, the wall can easily be detected as a door frame, and data loss problems tend to occur during door panel detection. This may result from two reasons. First, there are inadequate points located at these two kinds of objects, and therefore discriminative features cannot

be derived from the neighborhood points. Second, this is likely due to the similar structure among the wall, window, and door, which results in misclassification of them. In our experiment, the proposed reconstruction method can overcome the problem of misclassification to a certain extent.

4.2.2. Qualitative and quantitative analysis of BIM model

Our approach's BIM reconstruction quality and completeness are evaluated on these three sets of data and show comparisons with the ground truth IFC model. The experiment also evaluated the quality and completeness of the reconstructed BIM model with the value of recall rate (REC) and $PREC$ on five main component types, including space, door, window, ceiling, and floor. (REC) and $PREC$ are calculated in Eq. 5 and Eq. 6, where the definition of TP and FP is presented in Section 4.2.1 and GP is the total number of the components in the ground truth.

$$REC = \frac{TP}{(GP)} \times 100 \quad (6)$$

Table 6 shows the reconstruction performance for different components in the three sets of datasets. As shown in Fig. 14, Fig. 15 and Fig. 16, the cell decompositions, the reconstructed spaces, the segmented doors and windows, the reconstructed IFC models of the three regions, and the errors of the reconstructed model compared with the ground truth model are demonstrated respectively. The final model contains not only the final interior spaces (room and corridor) and volumetric structural elements (wall, ceiling, floor, door, and window) but also the “IfcRelVoidsElements” relations between opening and walls which describes an objectified relationship between a building element and one opening element that creates a void in the element.

The experimental results show that our reconstruction approach generally worked well on the test data sets without any data-specific tuning. The automatic point cloud labeling process reliably ignored even large furniture scanned in indoor interiors. In the three data sets, the proposed method achieved a 100% PREC rate and REC rate in the reconstruction of space, ceiling, and floor components. According to the results of Fig. 14 and Table 6, except for the window element, all components in the Case study1 are fully reconstructed and achieve 100% PREC and 100% REC. Same with the segmentation results for opening elements shown in Fig. 14(c), the window in the far left room is not detected, which is presented with “error 1” shown in Fig. 14(e). Fig. 12 shows the original classification result of this area. The main reason is that the size of these labeled points of this window is too small to be detected as a window component, which is filtered out with a size threshold in our approach. “Error 2” in Fig. 14(e) shows a false positive result of the window component, which is also caused by misclassification during point cloud labeling. The algorithm achieves a 91.7% PREC and 91.7% REC in window reconstruction of Case study1. There have three errors in the reconstructed result of the Case study2 shown in Fig. 15(e). The algorithm achieves 88.9% PREC and 100% REC on wall reconstruction, respectively. “Error 1” presented in Fig. 15(e) shows a false positive result of the wall reconstruction, which is caused by the misclassification of the point cloud. Similar problems can be found in the result of window components. In the reconstruction results of Case study2, two nonexistent windows are detected and reconstructed due to misclassification of the original point cloud, which results in 0% PREC

Table 8

Precision Rate (PREC) of the point cloud classification results in ISPRS benchmark.

Components	TUB1	UoM	TUB2	GM
Wall(%)	76.2	72.8	71.4	73.5
Ceiling(%)	81.2	82.7	84.3	79.2
Floor(%)	80.2	83.9	80.1	79.6
Column(%)	78.5	69.7	68.7	70.2
Overall(%)	55.5	53.2	54.3	55.2

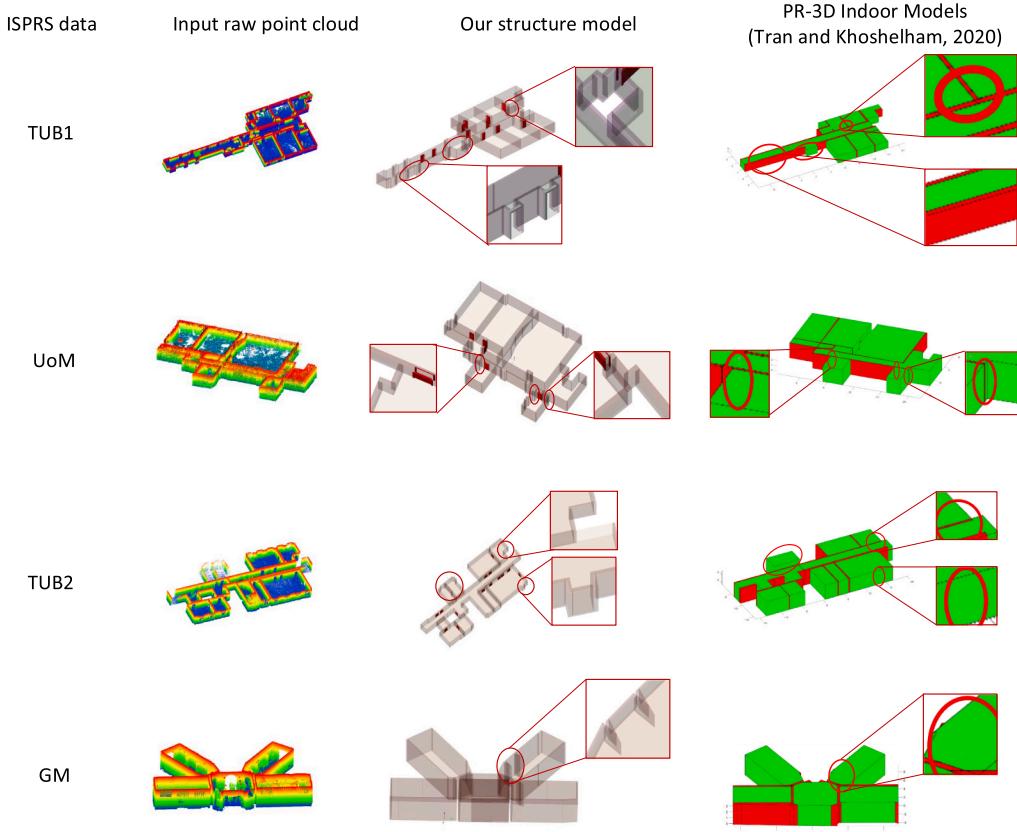


Fig. 19. Comparisons against ISPRS benchmark data sets. Our method produced 3D models that are closer to the input point cloud than the models generated by PR-3D Indoor models.

on window reconstruction of Case study2. Fig. 16 shows the reconstruction results of the Case study3. In this experiment, the proposed algorithm achieves 100% PREC and 100% REC on all components. All walls, floors, ceilings, windows, and doors are successfully reconstructed. The only problem is the position error of the door shown in Fig. 16(e) due to an inaccurate estimation of the center point of the door coordinate. The problem can be corrected by applying collision detection to the components. The comparison of the location of the windows and doors between the obtained BIM model and the ground truth is also presented in Fig. 17. Most of the reconstructed doors and windows are very close to the ground truth, and the proposed method achieves an average distance error of 0.21 m, 0.12 m, and 0.14 m, respectively, in Case study 1, Case study 2, and Case study 3.

In terms of geometric accuracy, the Root Mean Squared distance (RMS) and the corresponding histogram of approximate distances from vertical points to output IFC models are computed. Fig. 18 exhibits the distribution of the approximate distances from the point to the nearest triangulated surface on the reconstructed model, and Table 7 reports the evaluation metrics on three scenes. The different colors in Fig. 18 indicate the fitting error levels. For each reconstructed model, the histogram of geometric metrics is also presented in Fig. 18(b), in which the x-axis indicates geometric error ranging from 0 to 0.3 m, and the y-axis is the number of scenes contained in each bin. In the three experiments, most of the point clouds are located in the first three bins, which means our method produces 3D models that are close enough to input wall, ceiling, and floor points. According to the quantitative results shown in Table 7, our approach achieves the RMS under 0.1 m in regions 1 and 2, respectively. The average distance errors are 0.007 m, 0.054, and 0.019 m, respectively.

4.3. Experimental results of ISPRS data sets

The experimental results are also compared between our framework and the state-of-art indoor reconstruction method. Four ISPRS Benchmark data sets, including TUB1, UoM, the 2nd floor of TUB2, and GM, are used for comparison. In this experiment, for comparison purposes, the space decomposition on the raw point cloud was conducted instead of the classified indoor point cloud, and the final model contains the wall, beam, ceiling, floor, and space components. The Precision Rate (PREC) of the point cloud classification results in the ISPRS benchmark is shown in Table 8. Due to the lack of color information in the laser point cloud, the method achieves lower overall accuracy 55.5%, 53.2%, 54.3%, and 55.2%, respectively, in the ISPRS benchmark. However, the labeling for the structure elements is still with high accuracy. Fig. 19 illustrates the qualitative comparison results of the fours scenes. Visual examination of the reconstructed model by comparing it with the input raw point cloud indicates that most of the elements of the indoor environments are correctly reconstructed on both the PR-3D indoor modeling method and our method.

However, at first glance, it is obvious that the reconstructed structure models generated by the proposed method present more details of the scenes than PR-3D indoor models. For the TUB1 data set, the main space structure of the interior scene is recovered successfully by both our method and the PR-3D indoor modeling method. While for small components, like pillars and small corners, the PR-3D modeling method fails in keeping their original shapes. The failure mainly arises from the incorrect global plane refinement during surface extraction. In contrast, our structure model still preserves most of the small but important structure details marked by the red circle in Fig. 19. In addition, the

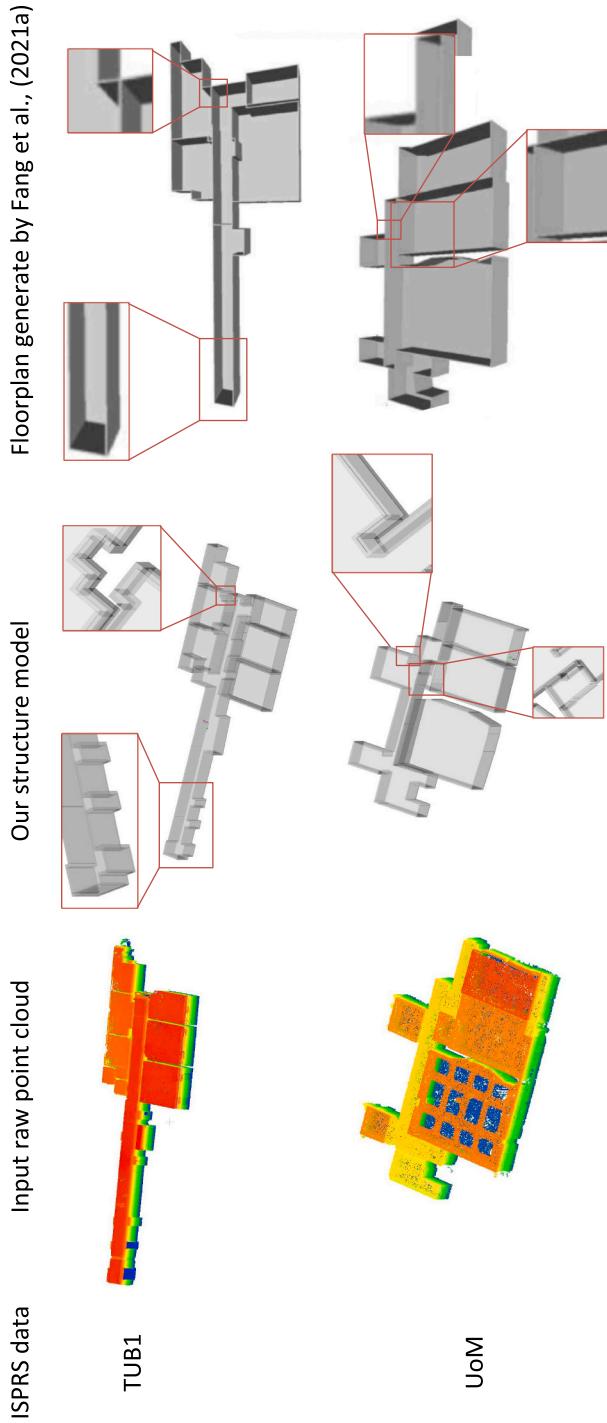


Fig. 20. Comparisons against on ISPRS benchmark data sets. Our method produced 3D models that are closer to the input point cloud than the models generated by Fang et al., (2021a).

Table 9

Precision Rate (PREC) of the point cloud classification results in RGB-D scenes.

Components	Scene 1	Scene 2	Scene 3	Scene 4
Wall(%)	79.3	78.2	80.5	79.8
Ceiling(%)	89.3	88.1	89.3	84.6
Floor(%)	87.7	90.1	88.9	87.7
Column(%)	69.3	64.9	65.3	71.2
Overall(%)	62.2	61.9	62.4	60.9

method proposed in this paper can still construct some openings in the case of laser point cloud data without color information, even though the segmentation accuracy of deep learning for colorless point clouds is reduced. However, the reconstructed opening components still suffer from the problems of false detection and data loss. The experiment on UoM and TUB2 data set exhibits a similar result to TUB1. The visual quality of our result is better since our method is robust enough handing missing data, while the PR-3D modeling method is not able to obtain correct space division results due to incomplete data. Moreover, the reconstructed structure model from our method contains more details, while the PR-3D modeling method preserves a lower geometric error on a small corner or small components. The experiments on the GM data set exhibit similar results generated by our approach and PR-3D indoor modeling method.

To further verify the effectiveness of the proposed method, we compare the experimental results with the newest floorplan generation algorithm proposed by Fang et al. [12]. As shown in Fig. 20, two scenes of TUB1 and UoM are used for comparison and the visual examination of the reconstructed model generated from two method respectively are also conducted. As expected, the reconstructed 3D models generated the proposed method present higher accuracy than from the method proposed by Fang et al. [12]. Similar to the PR-3D modeling method, the floorplan generation algorithm fails in modeling tiny concave and convex structures such as columns, U-shaped corner. Furthermore, for the UoM scene, There is one wall in this model obtained by floorplan generation algorithm that is not detected, which leads to a false merge operation of the two individual rooms.

It can be concluded that this proposed method out-performance comes from two aspects. First, our scene decomposition succeeds in separating the whole scene into the structure and non-structure parts, where even small vertical structures are also captured. Second, our space regularization approach is robust enough to assemble all structure primitives together and form a watertight polygon even in case of noisy data.

4.4. Experiments of RGB-D scenes data sets

To evaluate the feasibility of our approach in the reconstruction of non-Manhattan indoor environments and large-scale scenes, four experiments were conducted on the data sets collected by RGB-D devices. Those data sets have a lower accuracy compared with that from the lidar capturing system and contain lots of irregular shapes such as curved walls or irregular bending walls. The Precision Rate (PREC) of the point cloud classification results in RGB-D scenes is shown in Table 9. The method achieves the overall accuracy of 62.2%, 61.9%, 62.4%, and 60.9%, respectively, in four data sets.

Fig. 21 illustrates the reconstructed BIM models on these non-Manhattan scenes. It should be noted that the datasets are collected in Museums, most of which have no windows, and the doors only have a doorway without a door frame. Therefore openings have not been reconstructed in these cases. For better analysis of the reconstruction results, the details, especially in the curved region for four scenes, have been enlarged and highlighted. As demonstrated, our proposed method achieved accurate reconstruction of that basic interior structure, such as floors and large walls and spaces. Especially the curved walls in scene 1 and scene 2 are also maintained and constructed with quasi-planar

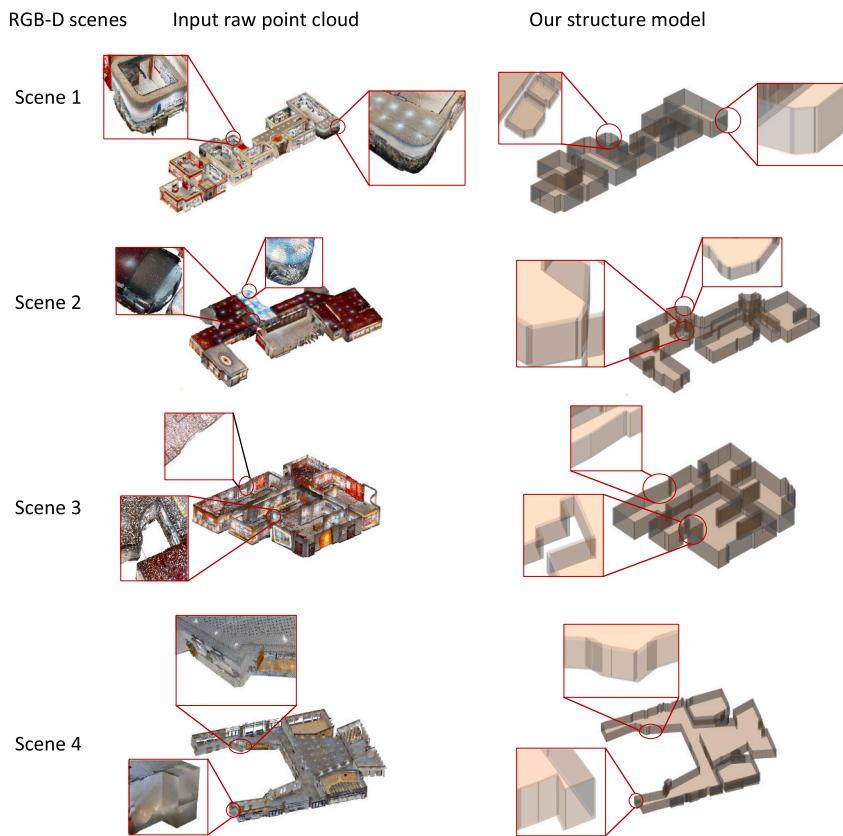


Fig. 21. Reconstruction results of the RGB-D scenes datasets.

shapes with a low number of facets. While for small structures such as beams and protrude structures, the proposed algorithm is able to preserve most of the small but important structure details, which can be easily found in scene 4. However, the algorithm fails in keeping the original curved shapes located in the left part of scene 3. Similar problems can also be found in the reconstruction results of scene 4. This failure mainly arises from the detection method for those curved components. In our approaches, those curved components, such as curved walls, cylinders columns, etc., are still reconstructed using planar fitting. Theoretically, a curved surface can also be approximated with continuous planes. However, the obtained 3D model will inevitably have some loss in geometric accuracy, which may be caused by inaccurate plane detection or incomplete data. Shape fitting algorithms, such as surfaces, ellipses, cylinders, etc., will be integrated to enhance the robustness of the reconstruction of curved structures in future work.

Quantitative results of the four data sets are also presented in Fig. 22 and Table 7. The algorithm achieves 0.02 m, 0.086 m, 0.074 m, and 0.018 average accuracy on the RGB-D scenes. For each reconstructed model, the x-axis indicated a geometric error in the histogram of geometric metrics ranging from 0 to 0.3 m. The y-axis represents the number of scenes contained in each bin. In the four experiments, the algorithm achieves 0–0.1 m reconstruction accuracy on most of the point clouds. Although our BIM model preserves a fine visual quality in all scenes, the geometric accuracy of the RGB-D scenes is lower than that of 2D-3D-S data sets. This phenomenon is reasonable because the structures of the RGB-D scenes are more complex than that of 2D-3D-S data, which contains lots of curved walls. The algorithm approximates those irregular walls by planar primitives, which causes a tiny gap in input data and results in lower geometric accuracy in those areas. Even though there have been some errors in our reconstruction results, the comparison results demonstrated that our proposed indoor BIM reconstruction method is robust enough to assemble all structure primitives

and form a complete BIM model for the non-Manhattan scene even in the case of noisy and incomplete data.

5. Conclusions

The paper proposed an automatic algorithm to reconstruct an indoor BIM model from RGB-D or LiDAR point cloud. The whole scene is first processed with a deep learning point cloud classification method. The potential structure elements such as the wall, column, ceiling, floor, and opening elements are extracted. Then, geometric primitives, e.g., vertical planes and horizontal planes, are detected with a hybrid surface extraction method and optimized by normal, parallel constraints. Those structure primitives are then used to generate the 2D binary map and partition the 2D space into a grid map. Finally, the 2D space is divided into different functional spaces and regularized with a grid map by solving a Markov Random Field problem. It should be noted that the proposed indoor 3D reconstruction method provides a flexible framework for the processing of both the original point clouds and the classified point clouds. Experimental results demonstrate the ability of this method to automatically generate a geometric and semantic consistent BIM model, which is competitive with respect to the existing method in terms of geometric accuracy and robustness.

However, The proposed method in this paper still has some limitations. The obtained 3D model will inevitably have some loss in geometric accuracy, especially for the curved structure, which may be caused by an inaccurate plane fitting process. The method is not yet able to reconstruct furniture, such as tables and chairs, even though they can be labeled by deep learning classification. Therefore, there are several aspects to improve in future work. The first one is to integrate the shape fitting algorithms, such as surfaces, ellipses, cylinders, etc., to enhance the robustness of the reconstruction of the curved structure. Secondly, higher-level structure constraints should be embedded inside the

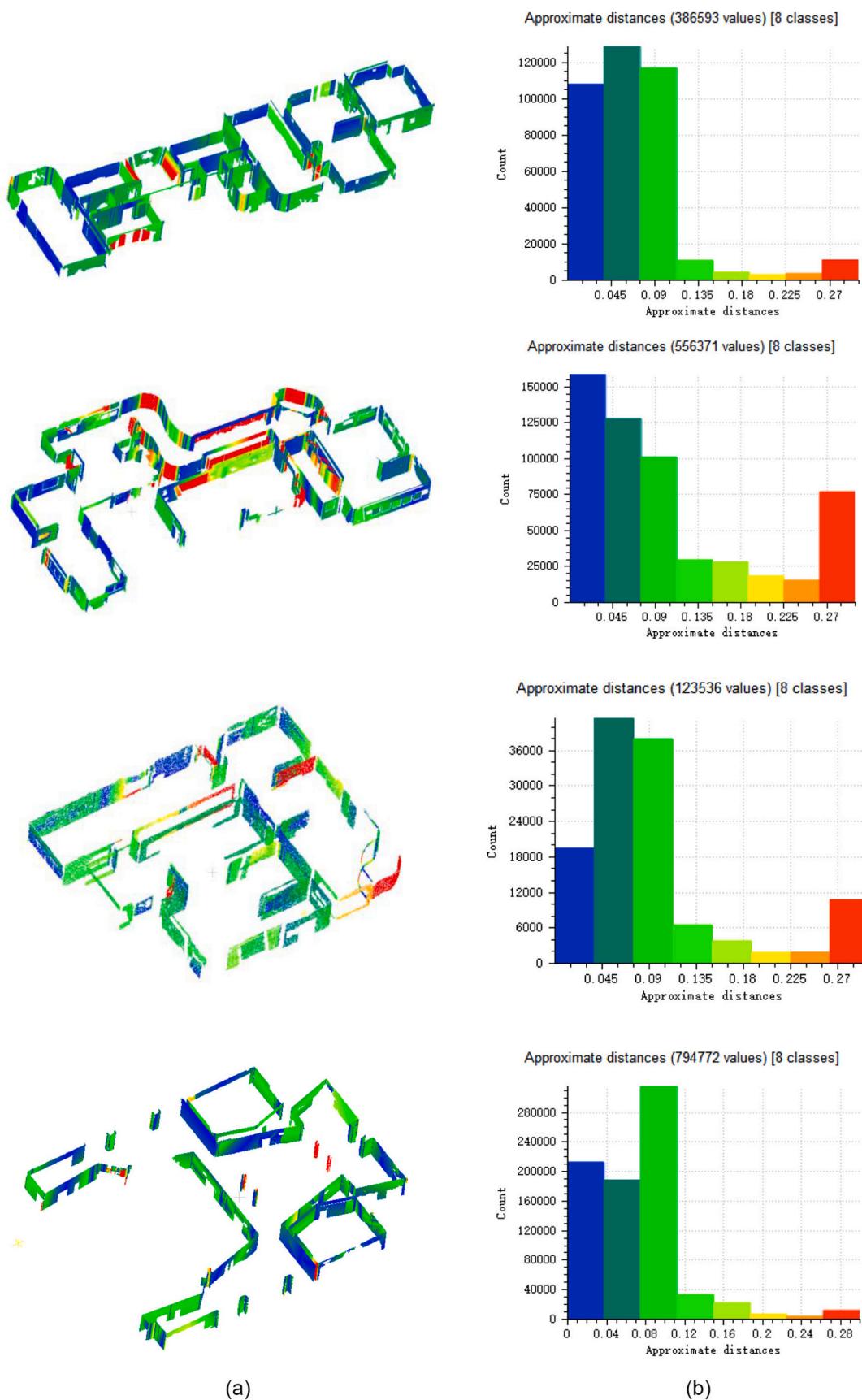


Fig. 22. Quantitative evaluation of the BIM model on RGB-D scenes data (a) Geometric distance between the model and raw point cloud, (b) Histogram of geometric metrics on three regions.

primitive detection processing, and space partition as the structure knowledge is believed to be able to reduce the impact of missing data on space segmentation. Thirdly, the accuracy of the deep learning classification should also be improved, especially for the point cloud without color information. Therefore the third future work is to improve the generalization ability of deep learning semantic segmentation models through more scans or algorithm optimization.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China (Projects Nos. 2019YFB210310, 2019YFB2103104) and in part by a Research Program of Shenzhen S and T Innovation Committee grant (Projects Nos. JCYJ20210324093012033, JCYJ20210324093600002), the Natural Science Foundation of Guangdong Province grant (Projects No. 2121A1515012574), the Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, MNR(Nos. KF-2021-06-125, KF-2019-04-014), the National Natural Science Foundation of China grant (Projects Nos. 71901147, 41901329, 41971354, 41971341) and the Foshan City to promote scientific and technological achievements of universities to serve industrial development support projects (Projects No. 2020DZXX04).

References

- [1] D. Acharya, K. Khoshelham, S. Winter, Bim-posenet: indoor camera localisation using a 3d indoor model and deep learning from synthetic images, *ISPRS J. Photogramm. Remote Sens.* 150 (2019) 245–258, <https://doi.org/10.1016/j.isprsjprs.2019.02.020>.
- [2] I. Armeni, S. Sax, A.R. Zamir, S. Savarese, Joint 2d-3d-semantic data for indoor scene understanding, *arXiv:1702.01105* (2022), <https://doi.org/10.48550/ARXIV.1702.01105>.
- [3] M. Bassier, M. Vergauwen, Unsupervised reconstruction of building information modeling wall objects from point cloud data, *Autom. Constr.* 120 (2020) 103338, <https://doi.org/10.1016/j.autcon.2020.103338>.
- [4] S. Becker, Generation and application of rules for quality dependent façade reconstruction, *ISPRS J. Photogramm. Remote Sens.* 64 (2009) 640–653, <https://doi.org/10.1016/j.isprsjprs.2009.06.002>.
- [5] F. Bosché, Automated recognition of 3d cad model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction, *Adv. Eng. Inform.* 24 (2010) 107–118, <https://doi.org/10.1016/j.aei.2009.08.006>.
- [6] F. Bosché, M. Ahmed, Y. Turkan, C.T. Haas, R. Haas, The value of integrating scan-to-bim and scan-vs-bim techniques for construction monitoring using laser scanning and bim: the case of cylindrical mep components, *Autom. Constr.* 49 (2015) 201–213, <https://doi.org/10.1016/j.autcon.2014.05.014>.
- [7] F. Bosché, A. Guilletmet, Y. Turkan, C.T. Haas, R. Haas, Tracking the built status of mep works: assessing the value of a scan-vs-bim system, *J. Comput. Civ. Eng.* 28 (2014) 05014004, [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000343](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000343).
- [8] F. Bosche, C. Haas, Automated retrieval of 3d cad model objects in construction range images, *Autom. Constr.* 17 (2008) 499–512, <https://doi.org/10.1016/j.autcon.2007.09.001>.
- [9] J. Chen, C. Liu, J. Wu, Y. Furukawa, Floor-sp: Inverse cad for floorplans by sequential room-wise shortest path, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 2661–2670, <https://doi.org/10.1109/ICCV.2019.000275>.
- [10] A. Dai, A.X. Chang, M. Savva, M. Halber, T. Funkhouser, M. Nießner, ScanNet: Richly-annotated 3d reconstructions of indoor scenes, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2432–2443, <https://doi.org/10.1109/CVPR.2017.261>.
- [11] A. Fabri, S. Pion, Cgal: The computational geometry algorithms library, in: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Association for Computing Machinery, New York, NY, USA, 2009, pp. 538–539, <https://doi.org/10.1145/1653771.1653865>.
- [12] H. Fang, F. Lafarge, C. Pan, H. Huang, Floorplan generation from 3d point clouds: a space partitioning approach, *ISPRS J. Photogramm. Remote Sens.* 175 (2021) 44–55, <https://doi.org/10.1016/j.isprsjprs.2021.02.012>.
- [13] H. Fang, C. Pan, H. Huang, Structure-aware indoor scene reconstruction via two levels of abstraction, *ISPRS J. Photogramm. Remote Sens.* 178 (2021) 155–170, <https://doi.org/10.1016/j.isprsjprs.2021.06.007>.
- [14] C. Feng, Y. Taguchi, V.R. Kamat, Fast plane extraction in organized point clouds using agglomerative hierarchical clustering, in: 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 6218–6225, <https://doi.org/10.1109/ICRA.2014.6907776>.
- [15] S. Gupta, P. Arbelaez, R. Girshick, J. Malik, Aligning 3d Models to Rgb-d Images of Cluttered Scenes, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2015, pp. 4731–4740, <https://doi.org/10.1109/CVPR.2015.7299105>.
- [16] P. Hübner, M. Weinmann, S. Wursthorn, S. Hinz, Automatic voxel-based 3d indoor reconstruction and room partitioning from triangle meshes, *ISPRS J. Photogramm. Remote Sens.* 181 (2021) 254–278, <https://doi.org/10.1016/j.isprsjprs.2021.07.002>.
- [17] N. Hichri, C. Stefani, L. De Luca, P. Veron, G. Hamon, From point cloud to Bim: a survey of existing approaches, in: ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XL5, 2013, pp. 343–348, <https://doi.org/10.5194/isprarchives-XL-5-W2-343-2013>.
- [18] S. Hong, J. Jung, S. Kim, H. Cho, J. Lee, J. Heo, Semi-automated approach to indoor mapping for 3d as-built building information modeling, *Comput. Environ. Urban. Syst.* 51 (2015) 34–46, <https://doi.org/10.1016/j.compenvurbusys.2015.01.005>.
- [19] J. Hou, A. Dai, M. Nießner, Revealnet: seeing behind objects in rgb-d scans, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2098–2107, <https://doi.org/10.48550/ARXIV.1904.12012>, arXiv.
- [20] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, A. Markham, RandlaNet: Efficient Semantic Segmentation of Large-Scale Point Clouds, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2020, pp. 11105–11114, <https://doi.org/10.1109/CVPR42600.2020.01112>.
- [21] L. Huan, X. Zheng, J. Gong, Georec: geometry-enhanced semantic 3d reconstruction of rgb-d indoor scenes, *ISPRS J. Photogramm. Remote Sens.* 186 (2022) 301–314, <https://doi.org/10.1016/j.isprsjprs.2022.02.014>.
- [22] Huang, S., Qi, S., Xiao, Y., Zhu, Y., Wu, Y.N., Zhu, S.C., 2018. Cooperative holistic scene understanding: unifying 3d object, layout, and camera pose estimation, in: Advances in Neural Information Processing Systems, arXiv. Doi:[10.48550/ARXIV.1810.13049](https://doi.org/10.48550/ARXIV.1810.13049).
- [23] S. Ikehata, H. Yang, Y. Furukawa, Structured indoor modeling, in: Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), IEEE Computer Society, USA, 2015, pp. 1323–1331, <https://doi.org/10.1109/ICCV.2015.156>.
- [24] Z. Kang, J. Yang, Z. Yang, S. Cheng, A review of techniques for 3d reconstruction of indoor environments, *ISPRS Int. J. Geo Inf.* 9 (2020), <https://doi.org/10.3390/ijgi9050330>.
- [25] K. Khoshelham, L. Diaz-Vilarino, 3d modelling of interior spaces: Learning the language of indoor architecture, in: F. Remondino, F. Menna (Eds.), Proceedings of ISPRS Technical Commission V Symposium, 23–25 June 2014, Riva del Garda, International Society for Photogrammetry and Remote Sensing, Italy, 2014, pp. 321–326, <https://doi.org/10.5194/isprarchives-XL-5-321-2014>.
- [26] K. Khoshelham, H. Tran, D. Acharya, L.D. Vilariño, Z. Kang, S. Dalyot, Results of the isprs benchmark on indoor modelling, *ISPRS Open Journal of Photogrammetry and Remote Sensing* 2 (2021), 100008, <https://doi.org/10.1016/j.photo.2021.100008>.
- [27] C.Y. Lee, V. Badrinarayanan, T. Malisiewicz, A. Rabinovich, Roomnet: End-to-end room layout estimation, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, p. arXiv, <https://doi.org/10.48550/ARXIV.1703.06241>.
- [28] C. Liu, J. Wu, Y. Furukawa, Floornet: a unified framework for floorplan reconstruction from 3d scans, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 201–217, <https://doi.org/10.48550/ARXIV.1804.00090>.
- [29] C. Liu, J. Wu, P. Kohli, Y. Furukawa, Raster-to-vector: revisiting floorplan transformation, in: 2017 IEEE International Conference on Computer Vision (ICCV), IEEE Computer Society, Los Alamitos, CA, USA, 2017, pp. 2214–2222, URL.
- [30] H. Macher, T. Landes, P. Grussenmeyer, From point clouds to building information models: 3d semi-automatic reconstruction of indoors of existing buildings, *Appl. Sci.* 7 (2017), <https://doi.org/10.3390/app7101030>.
- [31] S. Nikoohemmat, 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, <https://doi.org/10.1016/j.autcon.2020.103109>.
- [32] 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, <https://doi.org/10.1016/j.isprsjprs.2019.03.017>.
- [33] S. Ochmann, R. Vock, R. Wessel, R. Klein, Automatic reconstruction of parametric building models from indoor point clouds, *Comput. Graph.* 54 (2016) 94–103, <https://doi.org/10.1016/j.cag.2015.07.008>.
- [34] S. Oesau, F. Lafarge, P. Alliez, Indoor scene reconstruction using feature sensitive primitive extraction and graph-cut, *ISPRS J. Photogramm. Remote Sens.* 90 (2014) 68–82, <https://doi.org/10.1016/j.isprsjprs.2014.02.004>.
- [35] A.K. Patil, P. Holí, S.K. Lee, Y.H. Chai, An adaptive approach for the reconstruction and modeling of as-built 3d pipelines from point clouds, *Autom. Constr.* 75 (2017) 65–78, <https://doi.org/10.1016/j.autcon.2016.12.002>.

- [36] N. Ripperda, C. Brenner, Application of a formal grammar to facade reconstruction in semiautomatic and automatic environments, in: Proc. of the 12th AGILE Conference on GIScience, Citeseer, 2009, pp. 1–12. URL, <https://citeseervx.ist.psu.edu/viewdoc/download?doi=10.1.1.601.9129&rep=rep1&type=pdf>.
- [37] R.B. Rusu, Semantic 3d object maps for everyday manipulation in human living environments, KI - Künstliche Intelligenz 24 (2010) 345–348, <https://doi.org/10.1007/s13218-010-0059-6>.
- [38] R.B. Rusu, S. Cousins, 3d is here: Point cloud library (pcl), in: 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 1–4, <https://doi.org/10.1109/ICRA.2011.5980567>.
- [39] V. Sanchez, A. Zakhor, Planar 3d modeling of building interiors from point cloud data, in: 2012 19th IEEE International Conference on Image Processing, 2012, pp. 1777–1780, <https://doi.org/10.1109/ICIP.2012.6467225>.
- [40] R. Schnabel, R. Wahl, R. Klein, Efficient ransac for point-cloud shape detection, Computer Graphics Forum 26 (2007) 214–226, <https://doi.org/10.1111/j.1467-8659.2007.01016.x>.
- [41] S. Suzuki, K. Be, Topological structural analysis of digitized binary images by border following, Computer Vision, Graphics, and Image Processing 30 (1985) 32–46, [https://doi.org/10.1016/0734-189X\(85\)90016-7](https://doi.org/10.1016/0734-189X(85)90016-7).
- [42] P. Tang, D. Huber, B. Akinci, R. Lipman, A. Lytle, Automatic reconstruction of as-built building information models from laser-scanned point clouds: a review of related techniques, Autom. Constr. 19 (2010) 829–843, <https://doi.org/10.1016/j.autcon.2010.06.007>.
- [43] S. Tang, Y. Zhang, Y. Li, Z. Yuan, Y. Wang, X. Zhang, X. Li, Y. Zhang, R. Guo, W. Wang, Fast and automatic reconstruction of semantically rich 3d indoor maps from low-quality rgb-d sequences, Sensors 19 (2019), <https://doi.org/10.3390/s19030533>.
- [44] C. Thomson, J. Boehm, Automatic geometry generation from point clouds for bim, Remote Sens. 7 (2015) 11753–11775, <https://doi.org/10.3390/rs70911753>.
- [45] H. Tran, K. Khoshelham, Procedural reconstruction of 3d indoor models from lidar data using reversible jump markov chain Monte Carlo, Remote Sens. 12 (2020), <https://doi.org/10.3390/rs12050838>.
- [46] E. Valero, D.D. Mohanty, M. Ceklarz, B. Tao, F. Bosché, G.I. Giannakis, S. Fenz, K. Katsigarakis, G.N. Liliis, D. Rovas, et al., An integrated scan-to-bim approach for buildings energy performance evaluation and retrofitting, in: Proceedings of the International Symposium on Automation and Robotics in Construction, 2021, pp. 204–211, <https://doi.org/10.22260/ISARC2021/0030>.
- [47] R. Volk, J. Stengel, F. Schultmann, Building information modeling (bim) for existing buildings — literature review and future needs, Autom. Constr. 38 (2014) 109–127, <https://doi.org/10.1016/j.autcon.2013.10.023>.
- [48] C. Wang, Y.K. Cho, C. Kim, Automatic bim component extraction from point clouds of existing buildings for sustainability applications, Autom. Constr. 56 (2015) 1–13, <https://doi.org/10.1016/j.autcon.2015.04.001>.
- [49] 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, <https://doi.org/10.1016/j.autcon.2012.10.006>.